Research Proposal: Exceptional Model Mining in Dynamic and Complex Domains

Aniket Mishra*
2079259

Cristiana Cărbunaru†
2235773

Group 15

September 29, 2025

1 Overview

In this paper, the proposal for two potential research directions for **Exceptional Model Mining (EMM)** and **Subgroup Discovery (SD)** (Duivesteijn, Feelders, & Knobee, 2015; Herrera, Carmona, González, & Del Jesus, 2010) is presented. Both aim to develop interpretable data mining techniques for energy systems, environmental monitoring, or healthcare. In our opinion, healthcare is tougher, though.

Two options are considered:

- **Option A:** Incremental / Anytime Exceptional Model Mining for Stream Data and Handling Concept Drift.(Bifet & Gavalda, 2007; Giobergia, Pastor, de Alfaro, & Baralis, 2025; A. Liu, Song, Zhang, & Lu, 2017; Pesaranghader, Viktor, & Paquet, 2018; Vadlamudi, Chakrabarti, & Sarkar, 2012)
- **Option B:** Rich Subgroup Descriptions for Explaining Deep/Black-Box Models and Residuals.(Bach, 2024; Kearns, Neel, Roth, & Wu, 2018; Pimentel, Azevedo, & Torgo, 2022; Remil, Bendimerad, Plantevit, Robardet, & Kaytoue, 2021)

We need to check the feasibility of both for the possible publishing of the paper we write within the next 3-4 weeks. Option A is on algorithmic novelty for streaming data EMM, and Option B is on interpretability and model behavior analysis of deep learning models.

2 A: Streaming Drift + Incremental Exceptional Model Mining

2.1 Motivation

Most existing EMM and SD methods assume a static dataset (Duivesteijn et al., 2015; Herrera et al., 2010). In sensor and IoT devices, data arrives as a stream and sometimes undergoes concept drift(Bifet & Gavalda, 2007; A. Liu et al., 2017; Pesaranghader et al., 2018). It is

*e-mail: a.mishra3@student.tue.nl
†e-mail: c.carbunaru@student.tue.nl

computationally infeasible to rerun batch subgroup mining at every update. So, an incremental, drift-aware EMM method (Acheli, Grigori, & Weidlich, 2024; Ahmed, Tanbeer, Jeong, & Lee, 2009; Giobergia et al., 2025; C. Lee et al., 2022; C.-H. Lee, Lin, & Chen, 2001) with bounded resources would be nice, especially considering the ever-increasing cloud costs.

Insights: Aniket's personal motivation that led us in choosing these research directions Aniket explains: "In one of my earlier projects on a solar power plant, all our models suddenly started flagging every single device for having a "critical health score", meaning major failure within the next 3–6 months. After investigating it for a while, we realised that it was happening because the average ambient temperature increased from 50 to 60 +, triggering our bad data filter, causing the model to misbehave.

If we had a subgroup-based, drift-sensitive monitoring system, it could have flagged something like: "Subgroup of devices with ambient temperature $> 60°$C shows exceptional deviation in predicted healthscore." "

2.2 Research Questions

RQ1: *Incremental Maintenance of Subgroups:* Can we design an algorithm $\mathcal{A}$ that maintains the top-k or top-1 exceptional subgroups $\{S_1, \ldots, S_k\}$ over a data stream $\{x_t\}_{t=1}^{\infty}$ such that

$$S_t = \arg\max_{S \in \mathcal{H}_t} q_t(S),$$

where $q_t(S)$ is a stream-updated quality measure, without recomputing from scratch at each time step?

RQ2: *Drift Detection via Subgroup Quality:* Given subgroup scores $q_t(S_i)$ evolving over time, can we detect concept drift δ_t if

$$\exists S_i : \left| q_t(S_i) - \mathbb{E}[q_{t-w:t-1}(S_i)] \right| > \tau,$$

where τ is a statistical threshold and w a sliding window size?

RQ3: *Resource-Performance Trade-off:* What are the trade-offs between subgroup quality $q(S)$, update latency L, and memory M under constrained compute C, i.e.,

$$\max q(S) \quad \text{s.t. } L \leq L_{\max},\ M \leq M_{\max},\ C \leq C_{\max}\,?$$

RQ4: *Early Fault Detection:* Can subgroup-based drift metrics

$$D_t(S) = |q_t(S) - q_{t-1}(S)|$$

detect emerging anomalies earlier than global drift detectors (e.g., ADWIN, KL divergence)?

2.3 Expected Contributions

- An incremental EMM algorithm with anytime guarantees under streaming constraints (Meegle, n.d.; Sakthivel, Nithish, Tharun Kumar, Nivas, & Subhashini, 2023; Tangwongsan, Hirzel, & Schneider, 2022).
- A drift-aware mechanism for subgroup descriptions.
- Empirical validation on synthetic and real-world streaming datasets.
- Analysis of computational efficiency for edge deployment. This does not seem feasible right now. Aniket's company struggled with this, and there was a lot of manual effort.

2.4 Evaluation Plan

- **Datasets:** GE and Power Electronics refused to give us their data. Thus, we need to find relevant sensor data, preferably time series. Long or wide-format data will work, with data engineering, which is fine.
- **Baselines:** Batch EMM reruns, static subgroup miners, standard drift detectors. So, we could reduce the scope to make it fit the timeline.
- **Metrics:** Detection latency, subgroup stability. Future tests: memory usage, compute cost.

3 Option B: Model Residuals + Rich Descriptions

3.1 Motivation

Deep learning and other black-box models (Pimentel et al., 2022; Remil et al., 2021) are widely used in forecasting and prediction. By analyzing residuals as the target of interest, we can identify interpretable subgroups where the model consistently struggles—revealing data regions or feature combinations that cause poor predictions.

Classical SD uses conjunctive descriptions (e.g., $f_1 > a \wedge f_2 = b$). Extending SD to richer description languages like short decision trees, or symbolic expressions (Bach, 2024; L. Liu, n.d.), could capture more complex but still interpretable patterns. This could also help us find biases in the model, aiding in model fairness analysis.

Personal Motivation This idea is connected to the solar plant case described earlier. If we had analyzed model residuals directly, we might have quickly spotted that the high errors were confined to subgroups with unusually high ambient temperatures. That would have saved significant debugging time and clarified that the problem was with model generalization, not the solar panels.

Another reason was a downtime detection system Aniket built in a past project. By what he explained to the group, initially, it was a hard-coded chain of `elif` ladders running over aggregated time-series data. Later, Aniket made it configurable so that asset owners could define their own conditions like "Gen Speed > 300," "Wind Speed ≥ 3", and "Wind Speed ≤ 25," and "Pitch Angle ≥ 75". It worked (to an extent?), but writing and maintaining these rules was painful and heavily dependent on domain experts. If subgroup discovery could automatically find/learn such complex relationships from data, especially using richer description languages, it could help uncover operational rules directly, reducing manual effort and improving coverage. It could also aid in prescriptive analysis if we imagine it correctly.

3.2 Research Questions

RQ1: *Residual-based Exceptionalness:* Given model residuals $r(x) = |y - \hat{y}(x)|$, can we find subgroups

$$S = \{x \in \mathcal{X} : d(x) = 1\}$$

such that $\mathbb{E}_{x \in S}[r(x)]$ significantly deviates from $\mathbb{E}[r(x)]$?

RQ2: *Expressive Descriptions:* Which description languages $\mathcal{L}$ (e.g., conjunctions, polynomial predicates, shallow decision trees) optimize

$$\max_{d \in \mathcal{L}} q(d) = \text{effect size}(r|d)$$

subject to interpretability constraints $|\text{desc}(d)| \leq L$?

RQ3: *Interpretability-Quality Trade-off:* How does increasing expressiveness of $\mathcal{L}$ affect subgroup interpretability $I(d)$ and quality $q(d)$, e.g.,

$$\text{maximize } \lambda \cdot q(d) + (1 - \lambda) \cdot I(d), \quad \lambda \in [0, 1]?$$

3.3 Expected Contributions

- Empirical demonstration that residual-based EMM identifies meaningful model failure modes.
- Evaluation of alternative subgroup description languages for capturing nonlinear relationships.
- Comparative analysis of interpretability vs. subgroup quality.

3.4 Evaluation Plan

- **Datasets:** Public tabular datasets (UCI, Kaggle) with pretrained models.
- **Baselines:** Axis-aligned EMM rules, decision tree explanations.
- **Metrics:** Effect size on residuals, rule complexity, user interpretability score.

4 Comparison of Options

Aspect	**Option A (Streaming Drift)**	**Option B (Model Residuals)**
Novelty	High (underexplored area: incremental EMM under drift)	Moderate (extension of existing EMM to new description languages)
Feasibility (3 weeks)	Medium (requires simulation + incremental update design)	High (can use existing EMM frameworks + pretrained models)
Interpretability	Moderate	High
Evaluation complexity	High (streaming experiments)	Moderate (static datasets)
Publishability	High (novel systems/algorithmic paper)	High (interpretable ML application paper)

5 Recommendation

Given the timeline, **Option B** offers a safer path with faster implementation and clear results. **Option A** is more ambitious and suited for a systems-oriented paper if simulation and drift evaluation can be implemented in parallel.

6 Next Steps

1. Decide between Option A and Option B.
2. Conduct a focused literature review for the chosen direction.
3. Secure appropriate datasets (synthetic or real).
4. Implement minimal working prototype for evaluation.

References

Acheli, M., Grigori, D., & Weidlich, M. (2024). *Mining a minimal set of behavioral patterns using incremental evaluation.* Retrieved from https://arxiv.org/abs/2402.02921

Ahmed, C. F., Tanbeer, S. K., Jeong, B.-S., & Lee, Y.-K. (2009). Efficient tree structures for high utility pattern mining in incremental databases. *IEEE Transactions on Knowledge and Data Engineering*, *21*(12), 1708-1721. doi: 10.1109/TKDE.2009.46

Bach, J. (2024). Using constraints to discover sparse and alternative subgroup descriptions. *arXiv (Cornell University)*. doi: 10.48550/arxiv.2406.01411

Bifet, A., & Gavalda, R. (2007). Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 siam international conference on data mining* (p. 443—448). Retrieved from https://riverml.xyz/0.10.1/api/drift/ADWIN/#fnref:1

Duivesteijn, W., Feelders, A. J., & Knobee, A. (2015). Exceptional model mining. *Data Mining and Knowledge Discovery*, *30*(1), 47—98. doi: 10.1007/s10618-015-0403-4

Giobergia, F., Pastor, E., de Alfaro, L., & Baralis, E. (2025, July). Detecting interpretable subgroup drifts. In *Proceedings of the 31st acm sigkdd conference on knowledge discovery and data mining v.1* (p. 366—377). ACM. Retrieved from http://dx.doi.org/10.1145/3690624.3709259 doi: 10.1145/3690624.3709259

Herrera, F., Carmona, C. J., González, P., & Del Jesus, M. J. (2010). An overview on subgroup discovery: foundations and applications. *Knowledge and Information Systems*, *29*(3), 495—525. doi: 10.1007/s10115-010-0356-2

Kearns, M., Neel, S., Roth, A., & Wu, Z. S. (2018). *An empirical study of rich subgroup fairness for machine learning.* Retrieved from https://arxiv.org/abs/1808.08166

Lee, C., Ryu, T., Kim, H., Kim, H., Vo, B., Lin, J. C.-W., & Yun, U. (2022). Efficient approach of sliding window-based high average-utility pattern mining with list structures. *Knowledge-Based Systems*, *256*, 109702. doi: 10.1016/j.knosys.2022.109702

Lee, C.-H., Lin, C.-R., & Chen, M.-S. (2001). Sliding-window filtering: an efficient algorithm for incremental mining. In *Proceedings of the tenth international conference on information and knowledge management* (p. 263–270). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/502585.502630 doi: 10.1145/502585.502630

Liu, A., Song, Y., Zhang, G., & Lu, J. (2017). Regional concept drift detection and density synchronized drift adaptation. In *Proceedings of the twenty-sixth international joint conference on artificial intelligence, IJCAI-17* (pp. 2280–2286). Retrieved from https://doi.org/10.24963/ijcai.2017/317 doi: 10.24963/ijcai.2017/317

Liu, L. (n.d.). *Causal learning for heterogeneous subgroups based on nonlinear causal kernel clustering.* https://arxiv.org/html/2501.11622v1#abstract. (2025)

Meegle. (n.d.). *Data mining for edge computing.* https://www.meegle.com/en_us/topics/data-mining/data-mining-for-edge-computing. (n.d.)

Pesaranghader, A., Viktor, H., & Paquet, E. (2018). *Mcdiarmid drift detection methods for evolving data streams.* Retrieved from https://arxiv.org/abs/1710.02030

Pimentel, J., Azevedo, P. J., & Torgo, L. (2022). Subgroup mining for performance analysis of regression models. *Expert Systems*, *40*(1). doi: 10.1111/exsy.13118

Remil, Y., Bendimerad, A., Plantevit, M., Robardet, C., & Kaytoue, M. (2021). *Interpretable summaries of black box incident triaging with subgroup discovery.* Retrieved from `https://arxiv.org/abs/2108.03013`

Sakthivel, M., Nithish, R., Tharun Kumar, A., Nivas, S., & Subhashini, N. (2023). Improving accuracy of sensor data by frequent pattern mining algorithm using edge computing. *IRO Journal on Sustainable Wireless Systems*, *5*(4), 360—376. doi: 10.36548/jsws.2023.4.006

Tangwongsan, K., Hirzel, M., & Schneider, S. (2022). Sliding-window aggregation algorithms. *Encyclopedia of Big Data Technologies*, 1—7. doi: 10.1007/978-3-319-63962-8_157-2

Vadlamudi, S. G., Chakrabarti, P. P., & Sarkar, S. (2012). Anytime algorithms for mining groups with maximum coverage. In *Proceedings of the tenth australasian data mining conference - volume 134* (p. 209—219). AUS: Australian Computer Society, Inc. Retrieved from `10.5555/2525373.2525398`