SUPERCOOL TITLE LATER

Aniket Mishra
a.mishra3@student.tue.nl
Eindhoven University of Technology
Eindhoven, The Netherlands

Cristiana Cărbunaru
c.carbunaru@student.tue.nl
Eindhoven University of Technology
Eindhoven, The Netherlands

Abstract

ToDo

Keywords

Exceptional Model Mining, Subgroup Discovery, Residual Analysis, Interpretability

ACM Reference Format:
Aniket Mishra and Cristiana Cărbunaru. 2026. SUPERCOOL TITLE LATER. In *KDD '26: ACM SIGKDD Conference on Knowledge Discovery and Data Mining, August 09–13, 2026, Jeju, Korea.* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Overview

In this paper, we confirm that we have chosen option 2 for the scope of **Exceptional Model Mining (EMM)** and **Subgroup Discovery (SD)** [6, 8]. The aim of the project is to develop interpretable data mining techniques for black box models using residuals [2, 10, 15, 16]., and then figure out which EMM type is better + check the tradeoff between interpretability and the expressiveness of the language [14].

2 Introduction

ToDo

3 Theoretical Background

3.1 Model Residuals + Rich Descriptions

Deep learning and other black-box models [15, 16] are widely used in forecasting and prediction. By analyzing residuals as the target [9], we can get interpretable subgroups where the model consistently struggles, revealing data regions or feature combinations that cause poor predictions.

Classical SD uses conjunctive descriptions [8] (e.g., $f_1 > a \wedge f_2 = b$). Extending SD to richer description languages, such as short decision trees or symbolic expressions [2, 12], could capture more complex but still interpretable patterns. This could also help us find biases in the model, aiding in model fairness analysis [10].

ToDo More Later

3.2 Related Work

To add using references, later.

4 Problem Statement

Insights: Aniket's experience. 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 to 6 months. After investigating it for a while, we realised that it was happening because the average ambient temperature increased from 50 to 60 + degrees Celsius, 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.” “

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 example would be 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.

4.1 Research Questions and Hypotheses

Therefore, inspired by the aforementioned real-life issues, the following research questions are proposed, RQ1 representing the main question:

RQ1: Can we identify interpretable subgroups where black-box model residuals exhibit exceptional patterns, revealing systematic model failures or biases [9, 17]?

RQ2: Which description languages balance interpretability and expressiveness for capturing non-linear model failure patterns? Which description languages $\mathcal{L}$ (e.g., conjunctions, polynomial predicates, shallow decision trees) optimize

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

KDD '26, Jeju, Korea

ACM ISBN 978-1-4503-XXXX-X/2026/06
https://doi.org/XXXXXXX.XXXXXXX

subject to interpretability constraints $|\mathrm{desc}(d)| \leq L$ [1, 3–5, 7, 13, 14]?

RQ3: How does the expressiveness of the description language $\mathcal{L}$ influence the balance between subgroup quality and interpretability, and which configurations are optimal for different domains [3, 11, 17]?

Based on the research questions, hypotheses were composed as follows:

H1: Richer languages ($\mathcal{L}_{\text{poly}}, \mathcal{L}_{\text{tree}}, \mathcal{L}_{\text{symb}}$) will tend to increase $q(d)$ but decrease $I(d)$.

H2: Shallow trees or low-degree polynomials may achieve the best balance for practical domains (e.g., energy forecasting and fault analysis).

We will map the Pareto front $\{(q(d), I(d))\}$ per $\mathcal{L}$ to identify domain-appropriate operating points.

5 Solution Approach

ToDo

6 Experimental Results

6.1 Methodology

This section describes the methodology to solve our research questions. We formalize the problem, define the description languages for subgroup discovery, introduce the quality and interpretability measures, formulate the trade-off objective, and describe the optimization strategy.

6.1.1 Descriptions of Research Questions.

RQ1: *Residual-based Exceptionalness:*

Given a trained black-box model $f : \mathcal{X} \rightarrow \mathbb{R}$ and a dataset $D = \{(x_i, y_i)\}_{i=1}^{n}$, define residuals as:

$$r_i = |y_i - f(x_i)| \quad \text{or} \quad r_i = (y_i - f(x_i))^2$$

or some other form of loss calculation.

We aim to find subgroups [6, 8] $S = \{x \in \mathcal{X} : d(x) = 1\}$, where the subgroup description d maximizes the residual exceptionalness score:

$$q_{\text{residual}}(S) = \frac{|\mathbb{E}_{x\in S}[r(x)] - \mathbb{E}_{x\in D}[r(x)]|}{\sqrt{\mathrm{Var}_{x\in D}[r(x)]/|S|}}$$

This measure quantifies how **exceptional** the average residual in subgroup S is compared to the overall dataset D:

- $\mathbb{E}_{x\in S}[r(x)]$: average residual within subgroup S;
- $\mathbb{E}_{x\in D}[r(x)]$: average residual across the entire dataset;
- The numerator $|\mathbb{E}_{x\in S}[r(x)] - \mathbb{E}_{x\in D}[r(x)]|$ measures the magnitude of deviation between the subgroup and the global average;
- The denominator $\sqrt{\mathrm{Var}_{x\in D}[r(x)]/|S|}$ acts as a normalization factor, adjusting for global variance and subgroup size, similar to a *standard error*;
- The $q_{\text{residual}}(S)$ behaves like a z-score, identifying subgroups where residuals deviate significantly beyond what is expected by chance.

A high $q_{\text{residual}}(S)$ would show a subgroup where the model systematically underperforms (large residuals) or overperforms (small residuals), showing bias (underfitting) or overfitting.

RQ2: *Expressive Descriptions:*

We consider several hypothesis language classes for interpretable subgroup discovery:

(1) **Conjunctive (baseline):**

$$\mathcal{L}_{\text{conj}} = \{f_1\, \theta_1\, v_1 \;\wedge\; \cdots \;\wedge\; f_k\, \theta_k\, v_k\}$$

This language uses simple conjunctions of attribute–value conditions. It is the most interpretable form, expressing subgroups as a set of jointly satisfied predicates.

Example (solar data):

$$(\text{Irradiance} > 1500) \;\wedge\; (\text{PanelTemperature} > 60)$$

identifies a subgroup of solar panels operating under high irradiance and temperature.

Example (wind data):

$$(\text{WindSpeed} > 20) \;\wedge\; (\text{TurbineType} = \text{SWT-3.0-101})$$

defines wind turbines of type SWT-3.0-101 getting high wind speeds. Turbines generally get cut off at 25kmph wind speeds to prevent damage. This is not really an useful rule, I just couldn't think of anything rn.

(2) **Polynomial:**

$$\mathcal{L}_{\text{poly}} = \left\{ a_0 + \sum_i a_i f_i + \sum_{i,j} a_{ij} f_i f_j \;\leq\; \tau \right\}$$

The language captures nonlinear relationships through polynomial combinations of features [4]. It allows expressing curved decision boundaries or interaction effects.

Example (based on solar data):

$$\begin{aligned} 0.5 \cdot \text{POAIrradiance} - 0.3 \cdot \text{PanelTemperature} \\ + 0.2 \cdot \text{POAIrradiance} \times \text{AmbientTemperature} \\ \leq 1000 \end{aligned}$$

identifies conditions where the combined influence of irradiance and temperatures crosses a threshold. I can see the vision on this.

Example (wind data):

$$0.4 \cdot \text{WindSpeed}^2 - 0.6 \cdot \text{AmbientTemp} \;\leq\; 500$$

represents nonlinear effects between wind speed and temperature. The above I have (kind of) used to make a model (without the 2).

(3) **Decision Tree:**

$$\mathcal{L}_{\text{tree}} = \{\text{shallow trees with depth } \leq 3\}$$

Subgroups are represented as paths in a decision tree of limited depth. Each path corresponds to a conjunction of conditions learned from data, offering a balance between interpretability and expressiveness [7].

Example path (based on solar data):

$$\text{Irradiance} > 700 \;\rightarrow\; \text{Temperature} > 55 \;\rightarrow\; \text{ResidualError} > 10$$

defines a subgroup where the model performs poorly under high irradiance and temperature.

Example path (based on wind data):

$$\text{WindSpeed} > 10 \;\longrightarrow\; \text{BladePitch} > 20 \;\longrightarrow\; \text{PowerOutput} < 200$$

captures unexpected underperformance at high wind speeds and blade pitch angles.

(4) **Symbolic Expressions:**

$$\mathcal{L}_{\text{symb}} = \{\text{combinations of arithmetic and logical operations}\}$$

This language allows flexible symbolic expressions (e.g., ratios, differences, aggregates) generated through symbolic regression [1, 5], providing interpretable yet expressive formulas.

Example (based on solar data):

$$\frac{\text{PowerOutput}}{\text{Irradiance}} < 0.7$$

indicates panels with low conversion efficiency.

Example (wind data):

$$\text{WindSpeed} - 2 \times \text{RotorSpeed} > 5$$

captures turbines where rotor speed does not scale as expected with wind speed, because of mechanical or other issues.

The goal is to optimize the following objective:

$$d^* = \arg\max_{d \in \mathcal{L}} \lambda \cdot q(d) + (1 - \lambda) \cdot I(d)$$

where:

- $q(d)$: subgroup quality (effect size on residuals) [13]
- $I(d)$: interpretability score (inverse of description complexity) [3]
- λ: trade-off parameter

Constraints:

- Maximum of 3 to 5 predicates per description.
- Readable variable names and thresholds.
- Capability for visual representation.

RQ3: *Interpretability–Quality Trade-off:*

Using RQ2's languages ($\mathcal{L}_{\text{conj}}, \mathcal{L}_{\text{poly}}, \mathcal{L}_{\text{tree}}, \mathcal{L}_{\text{symb}}$), we formalize a trade-off between:

- $q(d)$: **quality** of a description d (e.g., residual exceptionalness / effect size on residuals);
- $I(d)$: **interpretability** of d, modeled as a decreasing function of its complexity.

Interpretability model. Let complexity(d) aggregate simple, auditable counts:

$$\begin{aligned}\text{complexity}(d) = w_1 &\cdot \#\text{predicates}(d)\\ + w_2 &\cdot \#\text{operators}(d)\\ + w_3 &\cdot \text{depth}(d)\\ + w_4 &\cdot \text{precision}(d)\end{aligned}$$

where precision(d) penalizes overly precise thresholds (e.g., many decimal places). We instantiate $I(d)$ as

$$I(d) = \frac{1}{1 + \text{complexity}(d)} \quad \text{or} \quad I(d) = \exp(-\beta \cdot \text{complexity}(d)),$$

with $\beta > 0$ controlling how sharply interpretability decays with complexity [11].

Complexity components: We define each component so that interpretability scoring is auditable and intuitive:

- **#predicates(d): number of atomic conditions:** Counts the number of simple boolean tests (e.g., $f > \tau$, $f = v$). Example: (Irradiance $>$ 800) $\wedge$ (PanelTemp $>$ 60) $\Rightarrow$ #predicates = 2.
- **#operators(d): arithmetic/logical operators used:** Includes $\{+, -, \times, \div, \wedge, \vee, <, \leq, >, =\}$. Example: (WindSpeed $>$ 12) $\wedge$ (TurbineType = V90) $\Rightarrow$ #operators = 3 ($>, =, \wedge$).
- **depth(d): path length in a decision tree:** Number of decisions from root to leaf. Example: Irradiance $>$ 700 $\rightarrow$ Temperature $>$ 55 $\rightarrow$ ResidualError $>$ 10 $\Rightarrow$ depth = 2.
- **#splits(d): number of tests in a tree path**: Equal to the number of internal nodes along a path (often same as depth).
- **precision(d): threshold granularity penalty:** Penalizes overly precise thresholds. Example: Irradiance $>$ 800.25 (1 extra decimal) adds penalty compared to Irradiance $>$ 800.
- **#terms(d): number of monomials in a polynomial:** Example: $a_0 + a_1 f_1 + a_2 f_2 + a_{12} f_1 f_2 \Rightarrow$ #terms = 4.
- **degree(d): polynomial degree:** Example: $0.4 \cdot \text{WindSpeed}^2 - 0.6 \cdot \text{Temp} \Rightarrow$ degree = 2.
- **#tokens(d): symbolic token count:** Total of operands (variables, constants) + operators. Example: $\frac{\text{Power}}{\text{Irradiance}} < 0.7$ $\Rightarrow$ #tokens = 5 (Power, Irradiance, 0.7, /, <). [17]

We then compute:

$$\begin{aligned}\text{complexity}(d) = w_1 &\cdot \#\text{predicates}(d)\\ + w_2 &\cdot \#\text{operators}(d)\\ + w_3 &\cdot \text{depth}(d)\\ + w_4 &\cdot \text{precision}(d)\end{aligned}$$

and derive interpretability:

$$I(d) = \frac{1}{1 + \text{complexity}(d)}.$$

Language-specific operationalization.

$$\begin{aligned}\mathcal{L}_{\text{conj}} &: \ \text{complexity}(d) = \#\text{predicates}(d).\\ \mathcal{L}_{\text{tree}} &: \ \text{complexity}(d) = \text{depth}(d) + \#\text{splits}(d).\\ \mathcal{L}_{\text{poly}} &: \ \text{complexity}(d) = \#\text{terms}(d) + \text{degree}(d).\\ \mathcal{L}_{\text{symb}} &: \ \text{complexity}(d) = \#\text{tokens}(d)\ (\text{operands} + \text{operators}).\end{aligned}$$

Trade-off objective. We search for descriptions that balance quality and interpretability:

$$d^* \;=\; \arg\max_{d \in \mathcal{L}} \; \lambda \cdot q(d) \;+\; (1 - \lambda) \cdot I(d), \qquad \lambda \in [0, 1].$$

Constraints (readability guards).

- #predicates$(d) \leq 5$ (concision);
- depth$(d) \leq 3$ for trees; polynomial degree$(d) \leq 2$;
- thresholds quantized to sensible units (e.g., 0.5°C, 0.5 m/s).

At each iteration, candidate subgroups are scored by $q(d)$ and $I(d)$, and Pareto-optimal solutions are retained to visualize the interpretability–quality frontier.

7 Conclusions

ToDo

References

[1] Guilherme Seidyo Imai Aldeia and Fabricio Olivetti de Franca. 2024. Interpretability in Symbolic Regression: a benchmark of Explanatory Methods using the Feynman data set. arXiv:2404.05908 [cs.LG]

[2] Jakob Bach. 2024. Using Constraints to Discover Sparse and Alternative Subgroup Descriptions. *arXiv (Cornell University)* (2024). doi:10.48550/arxiv.2406.01411

[3] Pablo Barceló, Mikaël Monet, Jorge Pérez, and Bernardo Subercaseaux. 2020. Model Interpretability through the Lens of Computational Complexity. arXiv:2010.12265 [cs.AI]

[4] Aleksey Buzmakov. 2020. Towards Polynomial Subgroup Discovery by means of FCA. In *CEUR Workshop Proceedings*, Vol. 2729. 1–12.

[5] F. O. de Franca, M. Virgolin, M. Kommenda, et al. 2023. Interpretable Symbolic Regression for Data Science: Analysis of the 2022 Competition. arXiv:2304.01117 [cs.LG]

[6] Wouter Duivesteijn, Ad J. Feelders, and Arno Knobee. 2016. Exceptional Model Mining. *Data Mining and Knowledge Discovery* 30, 1 (2016), 47—-98. doi:10.1007/s10618-015-0403-4

[7] Jack H. Good, Torin Kovach, Kyle Miller, and Artur Dubrawski. 2023. Feature Learning for Interpretable, Performant Decision Trees. In *Advances in Neural Information Processing Systems*, Vol. 36. 1–11.

[8] Francisco Herrera, Cristóbal José Carmona, Pedro González, and María José Del Jesus. 2010. An overview on subgroup discovery: foundations and applications. *Knowledge and Information Systems* 29, 3 (2010), 495—-525. doi:10.1007/s10115-010-0356-2

[9] Kimia Kamal and Bilal Farooq. 2022. Ordinal-ResLogit: Interpretable Deep Residual Neural Networks for Ordered Choices. arXiv:2204.09187 [cs.LG]

[10] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. 2018. An Empirical Study of Rich Subgroup Fairness for Machine Learning. arXiv:1808.08166 [cs.LG] https://arxiv.org/abs/1808.08166

[11] John P. Lalor and Hong Guo. 2022. Measuring algorithmic interpretability: A human-learning-based framework and the corresponding cognitive complexity score. arXiv:2205.10207 [cs.AI]

[12] Lu Liu. [n. d.]. Causal Learning for Heterogeneous Subgroups Based on Nonlinear Causal Kernel Clustering. https://arxiv.org/html/2501.11622v1#abstract. 2025.

[13] Cristian Munoz, Kleyton da Costa, Bernardo Modenesi, and Adriano Koshiyama. 2023. Evaluating Explainability in Machine Learning Predictions through Explainer-Agnostic Metrics. arXiv:2302.12094 [cs.LG]

[14] W. James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. 2019. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences* 116, 44 (2019), 22071–22080. doi:10.1073/pnas.1900654116

[15] João Pimentel, Paulo J. Azevedo, and Luís Torgo. 2022. Subgroup mining for performance analysis of regression models. *Expert Systems* 40, 1 (2022). doi:10.1111/exsy.13118

[16] Youcef Remil, Anes Bendimerad, Marc Plantevit, Céline Robardet, and Mehdi Kaytoue. 2021. Interpretable Summaries of Black Box Incident Triaging with Subgroup Discovery. arXiv:2108.03013 [cs.AI] https://arxiv.org/abs/2108.03013

[17] Marco Virgolin, Andrea De Lorenzo, Eric Medvet, and Francesca Randone. 2020. Learning a Formula of Interpretability to Learn Interpretable Formulas. In *Parallel Problem Solving from Nature – PPSN XVI*. Springer, 79–93.

Received ?? October 2025; revised ????; accepted ????