Explaining the Impact of Data Characteristics on Process Mining Algorithms

Dissertation zur Erlangung des Doktorgrades
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig-Maximilians-Universität München

vorgelegt von

Andrea Maldonado Hernández

München, den 26. Oktober 2025

Erstgutachter:	Univ.-Prof. Dr. Thomas Seidl Ludwig-Maximilians-Universität München
Zweitgutachter:	Univ.-Prof. Dr. Agnes Koschmider Universität Bayreuth
Drittgutachter:	Prof. Dr. Jochen De Weerdt Katholieke Universiteit Leuven
Vorsitz:	Prof. Dr. Sven Strickroth

Tag der Einreichung:	26. Oktober 2025
Tag der Disputation:	19. Dezember 2025

To all the girls from Guatemala.

Eidesstattliche Versicherung

(siehe Promotionsordnung vom 12.07.2011, § 8, Abs. 2 Pkt. 5)

Hiermit erkläre ich an Eides statt, dass die Dissertation von mir selbstständig und ohne unerlaubte Beihilfe angefertigt wurde.

München, den 26. Oktober 2025

Andrea Maldonado Hernández

Abstract

Process mining enables us to extract critical insights from event data, which consists of records of process executions. Nevertheless, the trustworthiness of its findings is threatened by a central challenge: A lack of a standardized, comprehensive evaluation framework for process mining algorithms. Since the set of real datasets is limited and access to real event data is scarce, a common practice is to evaluate algorithms on simplified, selective datasets. Such analyses fail not only to accurately assess algorithm results but also to capture the impact of key data characteristics, such as complexity, incompleteness, and statistical irregularities, on evaluation measures. Therefore, we require new robust evaluation methods that return valid and reliable results. This thesis addresses this critical gap by developing a data-driven methodology to systematically quantify the impact of data characteristics on process mining methods beyond single datasets. Three focus areas, *Data Characterization*, *Bias Mitigation*, and *Experimentation*, connect the contributions of this doctoral thesis:

First, we establish a foundation for robust data representation through structure-aware data characterization. Our paper on *Structure-Aware Principal Component Analysis for ordered data (DROPP)* addresses preserving intrinsic structural characteristics in dimensionality reduction by incorporating order, enabling an explainable visual comparison of datasets with low reconstruction errors and good compression. For event data, we introduce *FEEED: Feature Extraction from Event Data*, a domain-agnostic approach for extracting interpretable meta-features from event logs at multiple granularities, enabling the reproducible measurement and categorization of event data characteristics.

Second, to mitigate representational bias, we introduce frameworks for *Generating Event Data Intentionally (GEDI)*. *GEDI* and its interactive extension, *iGEDI*, enable process miners to generate event data with intentional meta-features, addressing scarcity and diversity of existing benchmarks. This work is extended by *Know Your Streams*, which conceptualizes, characterizes, and generates intentional event streams to address validity concerns in online process mining evaluations.

Finally, we provide an empirical method for measuring and explaining the impact of characteristics on process mining algorithms. The *SHAining on Process Mining* approach operationalizes explainability using *Shapley Value* analysis to systematically quantify how individual and interacting event data feature values impact core process discovery metrics, such as fitness, precision, and F-score for an underlying data-generating process.

The applicability of our framework, demonstrated on the process discovery task, spans major algorithmic design paradigms, such as top-down vs. bottom-up, as well as various event data types, providing interpretable insights into the robustness and associated trade-offs of process mining algorithms. Together, the papers present a comprehensive methodology for empirical evaluation in Process Mining, advancing the field toward more reproducible, valid, and generalizable research. By systematically linking data characteristics, algorithm behavior, and evaluation metrics, this thesis provides a valuable tool for researchers and practitioners to better understand and trust the results generated by their process mining tools. All methods presented in this thesis are provided in our open-source packages and respective repositories.

Zusammenfassung

Process Mining ermöglicht es uns, wichtige Erkenntnisse aus Ereignisdaten zu gewinnen, die in Aufzeichnungen von Prozessausführungen enthalten sind. Die Zuverlässigkeit der Ergebnisse wird jedoch durch eine zentrale Herausforderung beeinträchtigt: Es fehlt ein standardisierter, umfassender Bewertungsrahmen für Process-Mining-Algorithmen. Da nicht nur die Menge der realen Datensätze begrenzt ist, sondern auch der Zugang zu realen Ereignisdaten rar ist, ist es gängige Praxis, Algorithmen anhand vereinfachter, selektiver Datensätze zu bewerten. Solche Analysen versagen nicht nur bei der präzisen Bewertung der Algorithmusergebnisse, sondern auch bei der Erfassung der Auswirkungen wichtiger Datenmerkmale wie Komplexität, Unvollständigkeit und statistische Unregelmäßigkeiten auf die Bewertungsmaßnahmen. Daher benötigen wir neue robuste Bewertungsmethoden, die valide und zuverlässige Ergebnisse liefern. Diese Arbeit befasst sich mit dieser kritischen Lücke, indem sie eine datengesteuerte Methodik entwickelt, um den Einfluss von Datenmerkmalen auf Process-Mining-Methoden über einzelne Datensätze hinaus hinweg systematisch zu quantifizieren. Drei Schwerpunktbereiche, *Datencharakterisierung*, *Bias-Minderung* und *Experimentieren*, verbinden die Beiträge dieser Doktorarbeit:

Zunächst schaffen wir durch eine strukturbewusste Datencharakterisierung eine Grundlage für eine robuste Datendarstellung. Unsere Arbeit zum Thema *Strukturbewusste Hauptkomponentenanalyse für geordnete Daten (DROPP)* befasst sich mit der Erhaltung intrinsischer Strukturmerkmale bei der Dimensionsreduktion durch die Einbeziehung der Reihenfolge, was einen erklärbaren visuellen Vergleich von Datensätzen mit geringen Rekonstruktionsfehlern sowie eine gute Komprimierung ermöglicht. Für Ereignisdaten führen wir *FEEED: Feature Extraction from Event Data* ein, einen domänenunabhängigen Ansatz zur Extraktion interpretierbarer Metafunktionen aus Ereignisprotokollen mit mehreren Granularitäten, der die reproduzierbare Messung und Kategorisierung von Ereignisdatenmerkmalen ermöglicht.

Zweitens führen wir zur Bekämpfung von Darstellungsverzerrungen Frameworks für *Generating Event Data Intentionally (GEDI)* ein. *GEDI* und die interaktive Erweiterung *iGEDI* ermöglichen Ereignisdaten mit absichtlichen Metamerkmalen zu generieren, wodurch die Knappheit und mangelnde Vielfalt bestehender Benchmarks behoben werden. Diese Arbeit wird durch *Know Your Streams* erweitert, das absichtliche Ereignisströme konzeptualisiert, charakterisiert und generiert, um Validitätsprobleme bei der Bewertung des Echtzeit-Process-Minings zu beheben.

Schließlich bieten wir eine empirische Methode zur Messung und Erklärung der Auswirkungen von Merkmalen auf Process-Mining-Algorithmen. Der

Ansatz *SHAining on Process Mining* operationalisiert die Erklärbarkeit mithilfe der *Shapley-Wert-Analyse*, um systematisch zu quantifizieren, wie sich die Werte einzelner und interagierender Ereignisdaten auf zentrale Prozesserkennungsmetriken wie *Fitness*, *Präzision* und *F-Score* für einen zugrunde liegenden datengenerierenden Prozess auswirken.

Die Anwendbarkeit unseres Frameworks, die anhand der Prozesserkennungsaufgabe demonstriert wird, erstreckt sich auf wichtige algorithmische Designparadigmen wie Top-down vs. Bottom-up sowie auf verschiedene Ereignisdatentypen und liefert interpretierbare Einblicke in die Robustheit von Process-Mining-Algorithmen und die damit verbundenen Abwägungen. Zusammen bieten die vorgestellten Arbeiten eine umfassende Methodik für die empirische Bewertung im Process Mining und bringen das Fachgebiet in Richtung reproduzierbarer, valider und verallgemeinerbarer Forschung. Durch die systematische Verknüpfung von Datenmerkmalen, Algorithmusverhalten und Bewertungsmetriken bietet diese Arbeit ein wertvolles Werkzeug für Forschende und Praktikende, um die von ihren Process-Mining-Werkzeugen generierten Ergebnisse besser zu verstehen und ihnen mehr Vertrauen entgegenzubringen. Alle in dieser Arbeit vorgestellten Methoden sind in unseren Open-Source-Paketen und entsprechenden Repositorien verfügbar.

Resumen

La minería de procesos nos permite extraer información crítica a partir de los datos de eventos, que consisten en registros de ejecución de procesos. Sin embargo, la fiabilidad de sus resultados se ve amenazada por un reto fundamental: La falta de un marco de evaluación estandarizado y completo para los algoritmos de minería de procesos. Dado que no solo el conjunto de datos reales es limitado, sino que también el acceso a datos de eventos reales es escaso, una práctica habitual consiste en evaluar los algoritmos con conjuntos de datos simplificados y selectivos. Estos análisis no solo no evalúan con precisión los resultados de los algoritmos, sino que tampoco captan el impacto de las características clave de los datos, como la complejidad, la incompletitud y las irregularidades estadísticas, en las medidas de evaluación. Por lo tanto, necesitamos nuevos métodos de evaluación robustos que arrojen resultados válidos y fiables. Esta tesis doctoral aborda esta brecha crítica mediante el desarrollo de una metodología basada en datos para cuantificar de manera sistemática el impacto de las características de los datos en los métodos de minería de procesos, más allá de los conjuntos de datos individuales. Tres focos de interés, *Caracterización de datos*, *Mitigación de sesgos* y *Experimentación*, conectan las contribuciones de esta tesis doctoral:

En primer lugar, establecemos una base para la representación robusta de datos mediante la caracterización consciente de la estructura de los datos. Nuestro artículo sobre *Análisis de Componentes Principales con Reconocimiento de Estructura para Datos Ordenados (DROPP)* aborda la conservación de las características estructurales intrínsecas en la reducción de dimensionalidad. Lo hace mediante la incorporación del orden, lo que permiten una comparación visual explicable de conjuntos de datos con bajos errores de reconstrucción y una buena compresión. Para los datos de eventos, presentamos *FEEED: Extracción de Características de Datos de Eventos*, un enfoque independiente del dominio para extraer metacaracterísticas interpretables a partir de registros de eventos con múltiples granularidades. Esto permite la medición y la categorización reproducibles de las características de los datos de eventos.

En segundo lugar, para combatir el sesgo representacional, presentamos marcos para la *Generación Intencionada de Datos de Eventos (GEDI)*. *GEDI* y su extensión interactiva, *iGEDI*, permiten generar datos de eventos con metacaracterísticas intencionales, abordando la escasez y la falta de diversidad en los puntos de referencia existentes. Este trabajo se amplía con *Know Your Streams*, que conceptualiza, caracteriza y genera flujos de eventos intencionales para abordar cuestiones de validez en las evaluaciones de minería de procesos en línea.

Por último, proporcionamos un método empírico para medir y explicar el im-

pacto de las características en los algoritmos de minería de procesos. El enfoque *SHAining on Process Mining* pone en práctica la explicabilidad utilizando el análisis del *Valor de Shapley*. Así, cuantificamos sistemáticamente cómo los valores de las características de los datos de eventos, individuales e interactivos afectan las métricas básicas de descubrimiento de procesos, como *la aptitud*, *la precisión* y *la puntuación F* para un proceso subyacente de generación de datos.

La aplicabilidad de nuestro marco, demostrada en la tarea de descubrimiento de procesos, abarca los principales paradigmas de diseño algorítmico, como el enfoque ascendente frente al descendente, así como diversos tipos de datos de eventos. Esto proporciona información interpretable sobre la solidez de los algoritmos de minería de procesos y de las compensaciones asociadas. En conjunto, las publicaciones presentadas proporcionan una metodología completa para la evaluación empírica en la minería de procesos, lo que permite que este campo avance hacia una investigación más reproducible, válida y generalizable. Al vincular sistemáticamente las características de los datos, el comportamiento de los algoritmos y las métricas de evaluación, esta tesis proporciona una herramienta valiosa para que personas, que trabajan en la industria o investigación, comprendan mejor y puedan confiar en los resultados generados por sus herramientas de minería de procesos. Todos los métodos presentados en esta tesis se proporcionan en nuestros paquetes de código abierto y en sus respectivos repositorios.

Acknowledgments

Many remarkable people accompanied me during my doctoral journey. I am deeply grateful to all of you:

... my doctoral supervisor, Prof. Dr. Thomas Seidl, for giving me the opportunity to pursue my Ph.D. under your guidance and for your invaluable mentorship.

... my reviewers, Prof. Dr. Agnes Koschmider and Prof. Dr. Jochen De Weerdt, for our insightful conversations at conferences and for generously dedicating your time to evaluate my thesis.

... Prof. Dr. Sven Strickroth, for serving as chairman of the examination committee and for your thoughtful and considerate moderation during my defense.

... my brilliant co-authors and research colleagues—Ludwig, Florian, Anna, Simon, Zhicong, Christian, Hendrik, and Stephan—for the stimulating discussions that helped our ideas blossom, as well as my wonderful working students Anirudh, Nikolina, and Hasan, for their dedication, curiosity, and enthusiasm over the years.

... Susanne, Franz, and Joao, for your support with technical and administrative tasks during my time at the Data Mining and AI group.

... Prof. Dr. Martin Werner, for the opportunity to continue my research at the Technical University of Munich.

... my loving parents and grandparents, who gave me an education, a childhood full of memories, and wings beyond what they had.

... my inspiring sisters, whose support and own achievements continue to encourage me during challenging times.

... my dear friends, my extended family, and those closest to me, for their love, kindness, and for bringing laughter and smiles when I needed them most. Without you, this work would not have been possible. Ana, Keith, and especially Chris—thank you for your unwavering support and encouragement throughout this doctoral journey.

Finally, I thank all the other wonderful people with whom I have taught, learned, or simply shared memorable moments during this time.

Bibliographic Note

The topics discussed in this cumulative thesis have already been presented at prestigious scientific conferences and published in conference proceedings. The content is based on the following publications:

Contributions for this Dissertation

[MFA+25] Maldonado, A., Frey, C. M. M., Aryasomayajula S. A., Fahrenkrog-Petersen S. A., Zellner, L., Seidl, T. *SHAining on Process Mining: Explaining Event Log Characteristics Impact on Algorithms* in *Proceedings of the International Conference on Process Mining*, Montevideo, Uruguay, ICPM 2025

[MIR+25] Maldonado A., Imenkamp, C., Reiter, H., Seidl, T., Hasselbring, W., Werner, M., Koschmider, A. *Know Your Streams: On the Conceptualization, Characterization, and Generation of Intentional Event Streams* in *International Conference on Process Mining - Workshops*, Montevideo, Uruguay, ICPM 2025

Equal contribution by the first two authors.

[MFT+24] Maldonado, A., Frey, C. M. M., Tavares, G. M., Rehwald, N., Seidl, T. *GEDI: Generating Event Data with Intentional Features for Benchmarking Process Mining* in *Proceedings of the International Conference on Business Process Management*, Krakow, Poland, pages 221–237, BPM 2024

[MAF+24] Maldonado, A., Aryasomayajula, S. A., Frey, C. M. M., Seidl, T. *iGEDI: interactive Generating Event Data with Intentional Features* in *International Conference on Process Mining (ICPM) - Demo*, Copenhagen, Denmark, volume 3783, ICPM 2024

[BPM+24] Beer, A., Palotás, O., Maldonado, A., Draganov, A., Assent, I. *DROPP: Structure-Aware PCA for Ordered Data: A General Method and its Applications in Climate Research and Molecular Dynamics* in *Proceedings of the IEEE International Conference on Data Engineering*, Utrecht, Netherlands, pages 1143-1156, ICDE 2024

[MTO+23] Maldonado, A., Tavares, G. M., Oyamada, R. S., Ceravolo, P., Seidl, T. *FEEED: Feature Extraction from Event Data* in *International Conference on Process Mining (ICPM) - Demo*, Rome, Italy, volume 3648, ICPM 2023

Further Contributions

[RFM+25] Rauch, S., Frey, C. M. M., Maldonado, A., Seidl, T. *BEST: Bilaterally Expanding Subtrace Tree for Event Sequence Prediction* in *Proceedings of the International Conference on Business Process Management*, Sevilla, Spain, pages 415–432, BPM 2025

The article has received a **Runner-Up Best Student Paper Award**.

[RIL+25] Reiter, H., Imenkamp, C., Landsiedel, O., Maldonado, A., Rathje, P., Hasselbring, W. *The PM-EdgeMap: Towards Real-Time Process Mining on the Edge-Cloud Continuum* in *International Conference on Business Process Management - Workshops*,Seville, Spain, BPM 2025

[MZSS23] Maldonado, A., Zellner, L., Strickroth, S., Seidl, T. *Process Mining Techniques for Collusion Detection in Online Exams* in *International Conference on Process Mining (ICPM) - Workshops*, Rome, Italy, pages 336–348, ICPM 2023

[ZRS+20] Zellner, L., Richter, F., Sontheim, J., Maldonado, A., Seidl, T. *Concept Drift Detection on Streaming Data with Dynamic Outlier Aggregation* in *International Conference on Process Mining (ICPM) - Workshops*, Virtual Event, Padua, Italy, pages 206–217, ICPM 2020

[RMZS20] Richter, F., Maldonado, A., Zellner, L., Seidl, T. *OTOSO: Online Trace Ordering for Structural Overviews* in *International Conference on Process Mining (ICPM) - Workshops*, Virtual Event, Padua, Italy, pages 218–229, ICPM 2020

[MSRS20] Maldonado, A., Sontheim, J., Richter, F., Seidl, T. *Performance Skyline: Inferring Process Performance Models from Interval Events* in *International Conference on Process Mining (ICPM) - Workshops*, Virtual Event, Padua, Italy, pages 230–242, ICPM 2020

Contents

Abstract ix

Zusammenfassung xi

Resumen xiii

Acknowledgments xv

Bibliographic Note xvi

1 Introduction 1
- 1.1 Scientific Scope 4
- 1.2 Thesis Structure 5

2 Background 7
- 2.1 Foundations for Sequential and Event Analysis 7
  - 2.1.1 Sequential and Event Data 8
  - 2.1.2 Data Preprocessing 11
- 2.2 Process Mining 12
  - 2.2.1 Process Discovery 13
  - 2.2.2 Event Data Generation 18
- 2.3 Empirical Evaluation of Algorithms 20
  - 2.3.1 Experimental Reliability 22
  - 2.3.2 Methodological Validity 22
- 2.4 Representational Bias in Process Mining Evaluations 24
  - 2.4.1 Representational Bias 25
  - 2.4.2 Challenges and Implications 29

3 Contributions 33
- 3.1 Structure-aware Data Characterization 36
- 3.2 Intentional Event Data Generation 37
- 3.3 Measuring the Impact of Data Characteristics 41

4 Conclusion 45
- 4.1 Summary 45
- 4.2 Threats to validity 46
- 4.3 Future Work 47

Appendix 49

A DROPP: Structure-Aware PCA for Ordered Data: A General Method and its Applications in Climate Research and Molecular Dynamics . 49

B FEEED: Feature Extraction from Event Data 51

C Suppl. Materials to "FEEED: Feature Extraction from Event Data" . 53

D GEDI: Generating Event Data with Intentional Features for Benchmarking Process Mining 55

E iGEDI: interactive Generating Event Data with Intentional Features 57

F Know Your Streams: On the Conceptualization, Characterization, and Generation of Intentional Event Streams 59

G SHAining on Process Mining: Explaining Event Log Characteristics Impact on Algorithms 61

References 63

List of Figures 77

List of Tables 78

Chapter 1

Introduction

"All my life through, the new sights of Nature made me rejoice like a child."

– *Marie Curie*

The joy of discovering something new, something hidden in plain sight, is at the heart of science. When we work with complex real-world data, we aim to find a structure that will tell us something deeper. True discovery comes from careful work in shaping raw observations into something meaningful [104, 128].

This doctoral thesis is situated within *process mining*, a research area focused on extracting actionable insights from complex sequential *event data*, to understand and improve, e.g., business processes [111, 131]. However, a key challenge lies in the *validity* and *reliability* of evaluation practices for process mining algorithms. How can we know whether a method is capturing the true structure of noisy, complex event data? How can we trust the results? Trust matters. It involves the degree to which a human is willing to rely on a machine regarding a situation or a task at hand, consequently often shaping how that human decides to act [51]. From the reliability of a financial transaction [19] to the integrity of a healthcare diagnosis [6], algorithmic systems are increasingly trusted to make human-impacting decisions. For example, the automation market, driven by such systems, is rapidly expanding in a multi-billion-dollar[1] industry [55]. Lipton et al. [82] challenge simplistic claims about the inherent interpretability of linear approaches, and point out that trust in algorithmic system outputs is important yet slippery. Consequently, without consistent transparency, weakly evaluated algorithms risk bias, oversimplification, and ultimately a collapse of user trust [43, 51, 85]. While research on data uncertainty [23, 46, 81] in process mining develops models for stochastic processes and probabilistic event logs, this thesis addresses the complementary challenge of improving evaluative certainty.

Specifically, in process mining, the lack of a standardized, data-driven, and generalizable evaluation framework created a significant gap [108, 112]. Many algorithms are tested on narrow algorithm-specific datasets that do not reflect the

[1] 1 billion dollars = 10^9 dollars

full complexity and volume of current event data. As process miners develop increasingly sophisticated algorithms [7, 77, 79], the lack of comprehensive data hinders our ability to accurately assess algorithm performance [37, 123], compare different methods [15], and understand how various data characteristics, such as complexity, volume, and statistical irregularities, impact a results' generalizability [52, 71, 108].

Over the past decades, the relentless growth of data has transformed every scientific domain. We have moved from analyzing simple, structured datasets to grappling with high-dimensional, complex, and often noisy data [15]. Across domains the need to extract meaningful patterns from sequential data is a growing and fundamental challenge. *Sequential data* records interacting states, embedded in causal, temporal, and sometimes concurrent relationships [39], where the behavior of individual components and their interactions evolve over time, introducing complexity to the analysis. This complexity manifests across diverse domains, including *molecular dynamics simulations*, which model atomic movements as high-dimensional sequences [89, 102, 118], and *climate time series* of measurements —e.g., temperature, and radiation — which lead to massive pools of evolving data to capture non-linear, temporal spatial interactions [30, 53, 86].

This evolution is also particularly evident in process mining. While traditional data mining methods, such as *sequential pattern mining* [5], often focus on patterns within static or strictly sequential data, process mining requires one step further for capturing the dynamic and intertwined nature of real-world processes [78, 144]. Rather than isolated incidents, events in processes belong to a broader, interconnected system, whose behavior emerges from complex [15], nonlinear [41] relationships, such as concurrency, loops, and choices [111, 117, 131].

Taking these multiple variabilities into account highlights the need for robustness and realistic conditions in evaluations to ensure reliable and trustworthy findings. Thus, this thesis focuses on building a clear, reliable way to evaluate process mining algorithms, grounded in the characteristics of the data they operate on. In doing so, it aims to bridge the gap between discovery and trust, ensuring that the insights produced by algorithms can be assessed with transparency and confidence.

The central aim is to develop and validate a transparent, data-driven methodology that systematically explains how event data characteristics impact the evaluation of process mining algorithms, centering on the following question:

Research Question

How can a data-driven methodology systematically explain the impact of event data characteristics on the evaluation of process mining algorithms?

To address this question, we target the following three core research objectives, which were systematically defined, methodically implemented, and empirically tested following the principles of the *Research Science* [95] cycle:

Objective 1 *Method for Structure-Aware Data Characterization.*

The first objective focuses on establishing robust and interpretable data representations that preserve intrinsic structural properties of sequential and event data. Operationalizing event data characteristics as meta-features serves as a suitable domain-independent, measurable representation for subsequent steps of this thesis. For periodic sequential data, such as molecular dynamic trajectories or climate time series, *DROPP: Structure-Aware Principal Component Analysis for ordered data* [18] incorporates order into dimensionality reduction via Gaussian kernels. It performs *Principal Component Analysis* (PCA) [66] while preserving the intrinsic order-dependent structure with low reconstruction error. Results highlight the benefits of preserving the underlying characteristic structures of ordered data. For event data, *FEEED: Feature Extraction from Event Data* [94] systematically extracts interpretable meta-features at multiple granularities, enabling reproducible measurement and categorization of data properties. Together, the works in Section 3.1 establish the justification for a data-driven evaluation and provide robust and interpretable data representations to compare datasets in subsequent steps.

Objective 2 *Intentional Data Generation to Mitigate Bias.*

The second objective addresses the scarcity and lack of diversity in available event data benchmarks. This introduces representational bias, i.e. the gap between the available data samples and the true underlying distribution. Building on previously discussed *FEEED* [94], *GEDI: Generating Event Data Intentionally* [92] and its interactive extension *iGEDI* [90] introduce frameworks for controlled data generation via intentional meta-features. This enables the exploration of underrepresented regions of the event data design space and reduce representational bias. *Know Your Streams* [93] extends this work to event streams, conceptualizing, characterizing, and generating intentional datasets, which include streaming-specific challenges, such as out-of-order events. This extension moves beyond static logs and addresses validity concerns in online process mining evaluation.

Objective 3 *Quantified Explainability of Algorithmic Impact.*

Finally, the third objective aims to develop an explainable method to systematically measure and interpret how data characteristics impact algorithmic evaluation measurements. The *SHAining on Process Mining: Explaining Event Log Characteristics Impact on Algorithms* [91] method operationalizes explainability using Shapley value analysis, quantifying how individual and interacting event data features impact metric results, such as fitness, F-score, model size, and execution time of various process discovery algorithms. We provide interpretable insights into algorithm robustness against feature value variations and associated trade-offs.

The combination of these works establishes the individual building blocks for a comprehensive methodology for empirical evaluation in process mining. By systematically linking data characteristics, algorithms, and evaluation metrics, the thesis provides insights and a framework that enable more transparent, reliable, and generalizable studies.

1.1 Scientific Scope

A multitude of systematic reviews [8, 13, 68, 71, 107, 108, 113, 138] report that a central challenge of process mining methods lies in the validity and reliability of evaluation practices. To this end, the work *Process Mining Crimes* [107] highlight risky practices, such as using unrepresentative data, misleading metrics, or incomplete evaluations, that result in representational bias and threaten validity, on multiple levels. Therefore, the scientific scope of this thesis encompasses the systematic development of a data-driven, transparent, and reliable methodology [108], as a structured system for studying methods – supported by justification, experimentation, and framework – to gain empirical knowledge about process mining algorithm designs [96]:

- *Justification* denotes the theoretical grounding and logical validation of methodological choices, ensuring that hypotheses are convincingly supported by deductive reasoning from prior knowledge and underlying theoretical assumptions [76].

- *Experimentation* refers to the systematic implementation of controlled studies that test how well methodological assumptions describe empirical reality, thereby generating evidence on algorithm behavior under varying data conditions [35].

- *Framework* represents the structured system for the experimental execution, to provide coherent, interpretable, and generalizable empirical findings about process mining algorithm evaluations [113].

While classical process discovery serves as the primary case, our methodology extends to other tasks and beyond static event logs, reflecting the increasing importance of online and real-time process analysis [4, 27, 75, 87]. This work is founded on structure-aware feature characterization of different kinds of sequential data, based on both the identification and representation of the intrinsic structure, to ensure that algorithm evaluation is grounded in accurate, interpretable, and reproducible representations of the underlying datasets. While static process discovery serves as the primary case, the methodology extends beyond static event logs to *event streams*, reflecting the increasing importance of online and real-time process analysis [4, 27, 75, 87]. A key foundation of this methodology is structure-aware feature characterization, the systematic identification and representation of the intrinsic structure of different kinds of data. This pillar addresses the need to ensure that algorithm evaluation is grounded in accurate, interpretable, reproducible, and generalizable of the underlying datasets.

1.2 Thesis Structure

The remainder of this dissertation is structured as follows: In Chapter 2, we provide the foundational concepts, covering the fundamentals of sequential data, process mining, and the principles of algorithm evaluation, validity, and representation bias, as well as discuss current challenges. Chapter 3 first presents the broader empirical methodology, and subsequently provides an overview of the included publications in three subsections, structured based on the objectives, presented in Chapter 1. A summary of the main findings, complemented by a reflection on threats to validity and potential opportunities for future work, is given in Chapter 4. The original publications on which this cumulative dissertation is based can be found in Section 4.3.

Chapter 2

Background

"Only if we understand, can we care."

– Jane Goodall

In the spirit of Jane Goodall's quote, this chapter explores the fundamental concepts underlying process mining and the empirical evaluation of algorithms. Understanding these fundamentals is essential for situating the contributions of this thesis within a robust theoretical and methodological framework. In this chapter, we systematically build that framework, guiding the reader through four interconnected sections. This structure is designed to equip the reader with a coherent understanding of the theoretical, methodological, and practical dimensions that underpin this thesis.

We begin in Section 2.1 with the fundamentals of data mining, introducing sequential and event data, preprocessing strategies, and event data generation. Section 2.2 presents process mining as a discipline, including the process discovery task and representative discovery algorithms. In Section 2.3, we discuss the empirical evaluation of algorithms, focusing on experimental reliability, and methodological validity. Finally, Section 2.4 presents notions and definitions for representational bias and addresses current challenges in process mining evaluations resulting from it, emphasizing the importance of generalizable benchmarks.

2.1 Foundations for Sequential and Event Analysis

This section provides the methodological foundation for working with complex datasets, particularly sequential and event data, which are the central focus of this thesis. Data mining [48] is the process of discovering patterns and insights from large datasets using a variety of methods that draw from statistics, machine learning, and database systems. Given the nature of sequential and event data, the specific data mining tasks discussed in this section revolve around extracting temporal, behavioral, and structural insights from sequential data [32, 38, 39].

Next, Section 2.1.1 explores the fundamental concepts and data types. In Section 2.1.2, it also presents preprocessing techniques necessary for the analysis of sequential and event data. Lastly, Section 2.2.2 introduces the use of synthetic data generation as a controlled evaluation strategy for developing and testing algorithms.

2.1.1 Sequential and Event Data

Figure 2.1 shows a hierarchical specialization from general data to event-specific representations. From the broadest category data, this thesis specializes in *ordered data*, where entries are arranged or sorted, according to some criterion, without necessarily any temporal or intrinsic dependency behind it [57, 134]. *Sequential data* is a subset of ordered data with explicit sequence semantics, meaning that dependencies across positions are meaningful [38]. All sequential data is ordered, but not all ordered data exhibits inter-element dependencies.

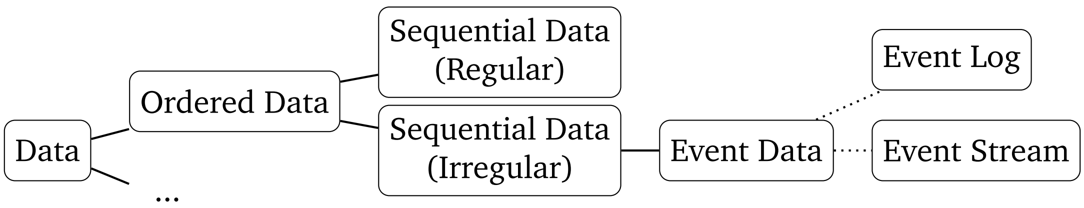

Figure 2.1: *Hierarchical specialization of data types, from ordered data to event data representations (logs and streams).*

Regular sequences are ordered measurements collected at uniform intervals in one or more dimensions, such as equidistant timestamps, spatial positions, or indexing steps. *Irregular sequences* are ordered data with variable intervals between measurements, such as asynchronous sensor readings or event-triggered measurements. Even with varying intervals, the sequential nature preserves trends, temporal or spatial continuity, and dependencies between successive observations. Beyond simple statistical averages, sequences require techniques that can parse and model the evolving dynamics of the enveloping system.

Examples of sequential data include molecular folding trajectories [49], climate measurements time series [62], as well as event execution records [133]:

In molecular dynamics simulation [89, 118], scientists model physical movements of atoms as high-dimensional sequences. Distilling a coherent picture of the system behavior by analysing the vast amount of sequential information, scientists can understand how microscopic interactions give rise to macroscopic properties. In Figure 2.2, a *protein folding for porcine NK-lysin trajectory* [67] demonstrates sequential data, exemplified by the position of the indexed atoms in three dimensions, as well as the temporal progression of the protein's movement. The recent Nobel Prize in Chemistry 2024 highlights the enormous potential of modelling these intricate, high-dimensional sequences, advancing knowledge about diseases like Parkinson's [59] and drug discovery [125]. These datasets often represent the

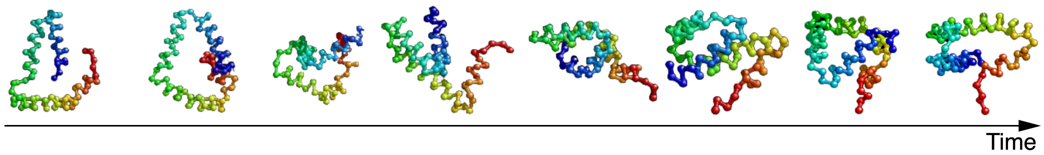

Figure 2.2: *Folding trajectory for a small alpha-helical protein [67].*

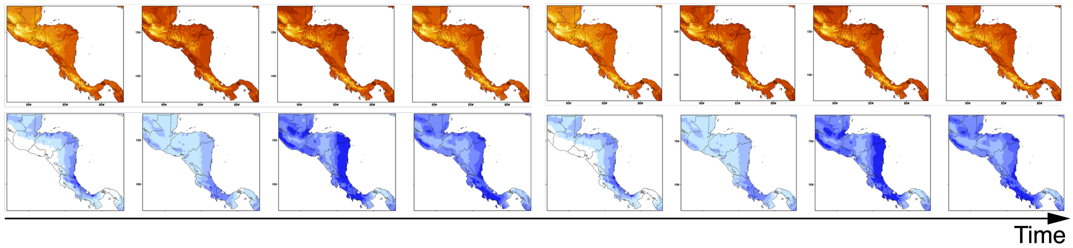

Figure 2.3: *Temperature and precipitation time series [62].*

positions of several hundred atoms over tens of thousands of steps in time. Molecular dynamics data often requires advanced analysis methods, to handle their high dimensionality, as they involve intricate interactions, vast datasets, and substantial computational demands [118].

In a different domain, climate researchers [86] have collected measurements, e.g., temperature, radiation, pressure, and wind speed across time and countries, leading to massive pools of high-dimensional sequences [30, 54]. In Figure 2.3, a high spatial resolution of *Temperature and Precipitation Time Series of Central America* [62] depicts suitable climate sequential data, containing spatial dimensions and several measure dimensions. As the work in [53] recognizes, climate data requires sophisticated methods to uncover its evolving dynamics, as it is shaped by temporal dependencies, feedback loops, and non-linear interactions between variables, and noise.

Formally, sequential data can be represented as

$$x_{1:T} = \langle x_1, x_2, ..., x_T \rangle, \quad x_t \in \mathbb{R}^d, \quad t \in \{1, ..., T\}$$

where t indexes the ordering variable, T is the sequence length, and each x_t is a d-dimensional measurement. The index t preserves data structure, ensuring successive elements carry meaningful relationships [38, 97]. Although we recognize that sequences may include categorical data $x_{td_i} \in \{1, \ldots, m\}$, e.g., label encodings or one-hot vectors, in this thesis, we focus on numerical sequential data.

Event data [3, 128] focuses on timestamped, discrete occurrences rather than periodic measurements, constituting a subset of sequential data. Each event records an activity and may include contextual attributes such as patient ID, procedure, or attending staff. Event data is irregular by default, as events occur asyn-

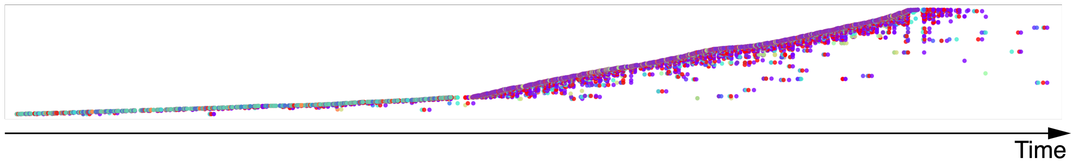

Figure 2.4: *Request for reimbursement at TU/e as dotted chart [133].*

chronously and may interleave across cases.

An *event* is the atomic instance of event data, formally represented as

$$e_i = (c_i, a_i, t_i), \quad c_i \in \mathcal{C}, \quad a_i \in \mathcal{A}, \quad t_i \in \mathcal{T}, \tag{2.1}$$

where $c_i \in \mathcal{C}$ denotes the case ID, $a_i \in \mathcal{A}$ the activity label, $t_i \in \mathcal{T}$ the timestamp. While an event can include additional contextual attribute values, we focus on the *control-flow* perspective of event data, which is exclusively structural and consists of case ID, activity label, and timestamps to order events. Furthermore, a *trace* σ_j of a case $c_j \in \mathcal{C}$ is the totally ordered sequence of events associated with that case:

$$\sigma_j = \langle e_1, e_2, \ldots, e_{m_j} \rangle, \quad \text{where } \forall i \in \{1, ..., m_j\}, \quad c_i = c_j, \quad t_1 \leq t_2 \leq ... \leq t_{m_j}. \tag{2.2}$$

In a trace, the sequence of events is ordered by ascending timestamps, and the length of the trace $|\sigma_j| = m_j$ is defined by the number of events in that trace.

In event data, we recognize several dimensions of sequences and order. On the one level, it presents sequences of activities within one trace. On another level, multiple process executions can start, or end, in a particular order. Due to the discrete nature of events, events of several traces of the same process can and often temporally overlap each other. Finally, there is also the higher level of order affecting both former ones, with the order of passing of time. The dotted chart [121] in Figure 2.4 displays an event log, with over 37,000 events and 7,000 cases, of *TU Eindhoven's Reimbursement Process* [133].

An *Event Log* is a finite set of n traces:

$$\begin{aligned} L = \{\sigma_1, \sigma_2, \ldots, \sigma_n\} = \Big\{ &\langle e_1^1, \ldots, e_{m_1}^1 \rangle, \\ &\langle e_1^2, \ldots, e_{m_2}^2 \rangle, \\ &\vdots \\ &\langle e_1^n, \ldots, e_{m_n}^n \rangle \Big\}, \end{aligned} \tag{2.3}$$

where $N = \sum_{j=1}^{n} m_j$ denotes the number of events in an event log L. Finally, the set of events in a log L is $\{e_i = (c_i, a_i, t_i)\}_{i=1}^{N}$, where $\forall e_i \in \sigma_j \in L$ [3, 128]. Although the dotted chart is suitable for visualizing event distribution and spotting timing patterns, it lacks deeper structural insight for further analysis. Event data can be analyzed along the structural — i.e., control-flow —, temporal, and organizational dimensions to capture activity sequences, bottlenecks, and involved resources [26, 131]. Static event logs examples include hospital patient logs [103], clinical interventions [116], and administrative records, such as the TU/e reimbursement histories [133] in Figure 2.4. In contrast, an event stream of the TU/e reimbursement process would represent the continuous flow of submission, approval, and payment events as they occur in real time, supporting online monitoring and anomaly detection. Consequently, measurements from IoT sensors and continuous monitoring systems are typical examples of event streams [22].

Opposed to event logs, *event streams* are real-time sequences of events that arrive continuously, and potentially without end [26]. Formally, we define an event stream

$$S = \langle \dots, e_{t_r-2}, e_{t_r-1}, e_{t_r}, e_{t_r+1}, e_{t_r+2}, \dots \rangle, \tag{2.4}$$

with each event, as in Equation (2.1), and in respect to a reference point in time $t_r \in \mathcal{T}$. The subset of events observed within a *window* time interval $[t_o - \delta, t_o]$ is

$$S_{[t_o-\delta,\, t_o]} = \langle e_i \mid t_o - \delta \leq t_i \leq t_o \rangle,$$

where $t_o - \delta$ marks the start of observation time, δ denotes the duration, and t_o the current time of observation. Unlike a finite event log L in Equation (2.3), an event stream is unbounded. Additionally, events in a stream may arrive asynchronously, out of order, with varying inter-arrival times, and interleaved across multiple cases $c_{t_o} \in \mathcal{C}$. I.e., directly consecutive events observed in an event stream may neither have the same case ID nor belong to the same trace.

2.1.2 Data Preprocessing

Preprocessing is a crucial step for data analysis, as raw datasets often contain heterogeneity, noise, and inconsistencies that may distort subsequent analysis [105]. For sequential data, common preprocessing operations include normalization, segmentation, aggregation, and dimensionality reduction [88, 97].

- *Normalization* [74] ensures that features with different scales contribute comparably.
- *Segmentation* techniques like *windowing* [63] split continuous data into interval units to reveal local patterns.
- *Aggregation* reduces high-frequency fluctuations by summarizing observations.
- *Dimensionality reduction* alleviates the curse of dimensionality by extracting compact yet informative representations. [122, 149] To preserve the

structure of ordered data, ideally, a dimensionality reduction approach, as any other preprocessing approach, should not break that invariant structure across data instances.

Event data preprocessing addresses imperfections that arise from recording process executions. Typical steps involve filtering infrequent or irrelevant traces, reducing noise by removing duplicates or imputing missing values, and homogenizing attributes such as timestamps or activity labels [42]. These operations are essential to ensure the interpretability and reliability of process models derived from event logs [9, 109, 130]. Complementary, trace encoding approaches such as n-grams, bag-of-activities, or embedding-based methods (e.g., act2vec, trace2vec, log2vec) map variable-length event traces into numerical representations suitable for machine learning [17, 40, 114].

Preprocessing not only prepares data for algorithmic consumption but also shapes model performance, interpretability, and evaluation outcomes. Choices at this stage can improve robustness and convergence of learning algorithms, yet inappropriate or inconsistent preprocessing may bias comparisons or threaten the validity of empirical studies [107, 113]. Consequently, preprocessing constitutes a foundational aspect of process mining and sequential data analysis, determining the reliability of both methodological advances and experimental results.

2.2 Process Mining

Process mining bridges data mining and process science. It aims to extract knowledge from event logs, which record the execution of activities in information systems, and transform this knowledge into process models and insights [131]. The main goals of process mining include supporting process understanding, identifying deviations from prescribed procedures, and enabling continuous improvement of operational performance. From an algorithm engineering perspective [96], process mining can be understood as a collection of tasks, comprising data assumptions, goals, algorithms, and evaluation metrics, which vary on the particular process mining task. Table 2.1 provides a short overview for the application task of this thesis, process discovery in violet, as well as two other examples tasks, conformance checking, and predictive monitoring, alongside a non-exhaustive, yet illustrative selection of their components.

While the goals, data assumptions, algorithmic paradigms and evaluation measures differ depending on the process mining task, all these tasks present clear values for each specification. Furthermore, all tasks utilize event data, e.g., event logs, as input, setting the foundation for a data-centric, task-independent approach. This thesis demonstrates the applicability of the framework for process mining tasks in process discovery, which will be introduced next.

Process Mining Task	**Goals**	**Data Assumptions**	**Algorithms**	**Evaluation Measures**
Process Discovery	Build a process model capturing actual behavior.	Event logs with case ID, activity, and timestamp	Inductive Miner, ILP Miner, Split Miner, etc.	Fitness, precision, model size, etc.
Conformance Checking	Detect and quantify deviations between log and model.	Event log and process model	Token-based replay, alignment-based checking, declarative rules	Fitness, precision, deviation counts, cost of deviations
Predictive Monitoring	Forecast future behavior (e.g., remaining time, next activity, outcome).	Event logs with patterns, outcomes or attributes	Machine learning (e.g., LSTM, random forest), prefix-based models	Accuracy, MAE/RMSE, precision/recall, timeliness of predictions

Table 2.1: *Illustrative overview of main process mining tasks.*

2.2.1 Process Discovery

The goal of process discovery [77, 135] is to derive a *process model* M, expressed in a formalism, such as *Petri net* [129], *BPMN* [77], or *process tree* [73], that reproduces the behavior observed in a log L, as defined in Equation (2.3) with high quality. Model quality is typically assessed along multiple dimensions [65]:

- *Fitness:* The extent to which traces in L can be replayed by M.
- *Precision:* The degree to which M avoids allowing behavior not seen in L.
- *Generalization*: How well M can capture likely but unobserved behavior.
- *Simplicity*: The structural complexity of M. Measured e.g. by model size [13].

Formally, the process discovery problem can be framed as an optimization problem

$$M^* = \arg\max_{M \in \mathcal{M}} Q(L, M), \tag{2.5}$$

where $\mathcal{M}$ is the set of candidate models and Q is a quality function aggregating measures for fitness, precision, generalization, and simplicity.

Process discovery algorithms can be broadly categorized into imperative and declarative families [110]. *Imperative approaches*, such as Petri net or BPMN discovery, aim to model explicit relations between activities, while *declarative lan-*

guages, such as Declare [34], capture flexible behavioral constraints. These imperative models rely on *control-flow* relation patterns between activities to define behavior: *sequence* ($A \rightarrow B$) means activity A must be followed by B; *exclusive choice/or* ($A \otimes B$) means either A or B can occur, but not both; and *concurrency* ($A \oplus B$) means A and B can occur in any order or simultaneously. The corresponding BPMN element for concurrency is formally named the *parallel gateway*, as depicted in Figures 2.5 to 2.7.

Among imperative methods, algorithms differ fundamentally in their approaches: *Top-down techniques*, such as the *Inductive Miner* [77], start from a high-level process structure, based on Directly-Follows graphs [80], and iteratively refine it into smaller, block-structured, sound components.

In contrast, *bottom-up techniques* start with a fine-grained analysis, which progressively integrates low-level behavior into non-local control-flow patterns, composing a comprehensive model. These bottom-up techniques can differ further in their underlying principles, ranging from formally grounded optimization-based miners to heuristic, frequency-driven ones. *Formally grounded* approaches, like *Integer Linear Programming (ILP) Miner* [135], rely on mathematical optimization and region theory to ensure maximal precision and soundness. They offer strong behavioral guarantees at the cost of computational efficiency and robustness to noise. In contrast, *heuristic methods*, such as the *Split Miner* [14], approximate the discovery process using frequency- and structure-based heuristics, prioritizing scalability, model simplicity, and practical interpretability over formal optimality.

In this thesis, we focus on imperative discovery approaches, which are particularly suitable for generating descriptive process models from complex event logs. We demonstrate the application of the following algorithms using an example log: For visual consistency and enhanced interpretability in this thesis, the models discovered by all algorithms—including the natively produced Process Trees and Petri Nets—are uniformly represented using the industry-standard Business Process Model and Notation (BPMN) throughout the following examples.

Example 2.2.1

$$\begin{aligned} L = \{&\langle A, B, C, D, E\rangle, \\ &\langle A, C, B, D, E\rangle, \\ &\langle A, F, E\rangle\} \end{aligned}$$

Among imperative techniques, the **Inductive Miner** is a top-down algorithm that guarantees sound and block-structured models [77]. The application of the Inductive Miner on Example 2.2.1 is shown in Figure 2.5. Its working principle can be summarized as follows:

1. **Base Case:** If the log is empty or has a single activity, return a simple model. In Example 2.2.1, we start from a single first activity A, thus not a base case.

2. **Find a Split:** Segment the log by sequence, concurrency, loops, and choices. In example 2.2.1, we find the first split after A into: A is followed by three options: $A \rightarrow (B, C, D, E)$ or $A \rightarrow (C, B, D, E)$ or $A \rightarrow (F, E)$.

3. **Recurse:** Apply the same procedure recursively to each sublog. For example 2.2.1, we investigate the three options of the first split to identify the next one. Thus, resulting in a choice of two branches: $A \rightarrow (B \oplus C) \rightarrow D \rightarrow E)$ (left branch)— involving parallel activities B, C — and $A \rightarrow F \rightarrow E)$ (right branch), containing sequences only.

4. **Reconstruct:** Combine results into a process tree for Petri nets or BPMN. Finally, for Example 2.2.1, the model is reconstructed by joining both splits, as shown in the last step of Figure 2.5: $A \rightarrow (((B \oplus C) \rightarrow D) \otimes F) \rightarrow E$

This recursive decomposition particularly guarantees soundness for the Inductive Miner. Variants like IMf [77] filter out infrequent behavior to avoid overfitting, which makes it robust to noise. This process is like sorting a deck of cards based on a complex set of rules. The step-by-step, top-down approach allows the Inductive Miner to discover structured, hierarchical process models from complex and noisy event logs.

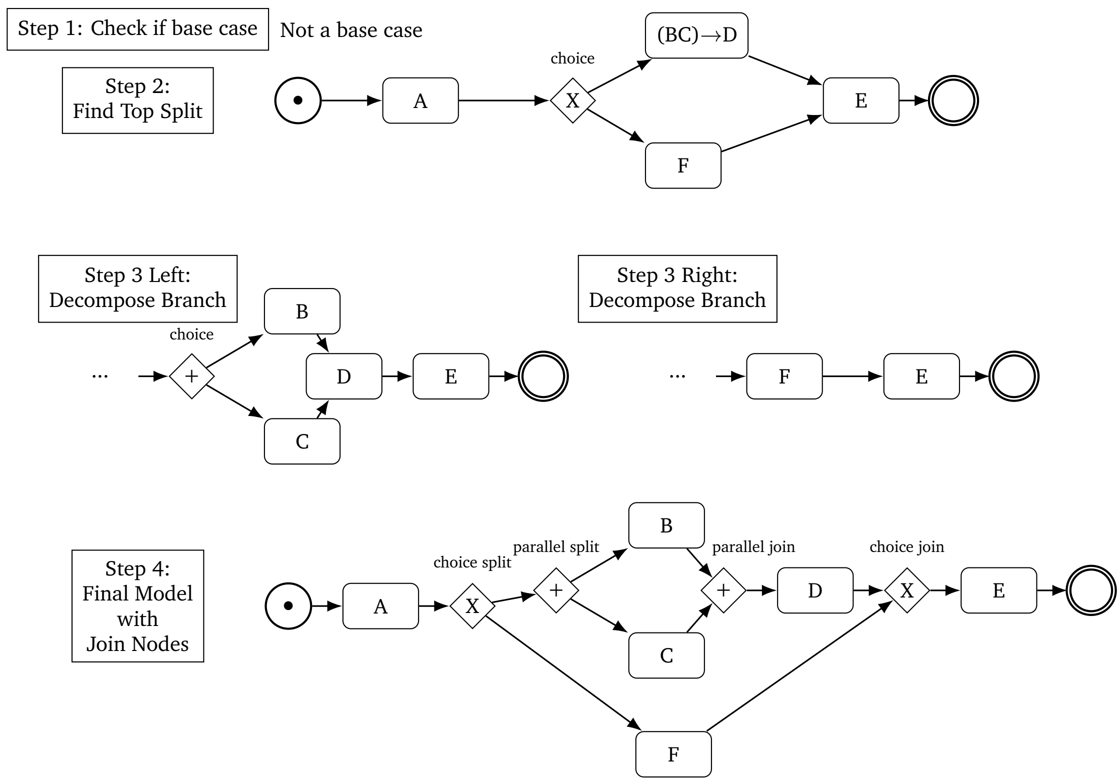

Figure 2.5: *Inductive Miner [77] example in BPMN notation in 5 steps*

In contrast, the **Integer Linear Programming Miner (ILP)** [135] is a bottom-up algorithm that discovers complex, non-local control-flow patterns that top-down algorithms may miss. It achieves this by constructing Petri nets using region theory and *integer linear programming*. Figure 2.6 shows the application on Example 2.2.1 as a BPMN, for consistency. Its working principle can be summarized as the following:

1. **Identify Dependencies:** The miner analyzes the event log to find all possible sequential relations between activities. In Example 2.2.1 that is all directly follow relations: $(A \to B, A \to C, B \to D, C \to D, D \to E, A \to F, F \to E)$

2. **Interpretation of Discovered Places:** Each place identified is interpreted as enforcing a specific sequential or control-flow constraint, as parallel or choice relations, between activities. An Integer Linear Programming (ILP) solver determines the minimal assignment of places and arcs that satisfy all constraints, while also ensuring that only traces in the log can be executed, i.e., precision. For Example 2.2.1 in Figure 2.6, the discovered places can be grouped and interpreted as follows: $p_1, p_2 : A \to (B \oplus C)$, $p_3, p_4 : (B \oplus C) \to D$, $p_5 : D \to E$, $p_6 : A \to F$, $p_7 : F \to E$. This step reveals the logical structure of the process: after activity A, activities B and C may occur in parallel and are later synchronized by D, while an alternative path $A \to F \to E$ also exists.

3. **Mapping Places to BPMN Gateways:** The discovered places are then mapped to BPMN control-flow elements, where non-sequential activity relations correspond to gateways that control splits and joins. This mapping step translates the Petri-net level semantics produced by the ILP Miner into BPMN constructs, making concurrency and choice explicit. In Example 2.2.1 (p_1, p_2, p_6) correspond to an *XOR(⊗)–AND(⊕) split* following activity A, expressing that A can either trigger the parallel branch $(B \oplus C)$ or the alternative sequential branch F. Additionally, (p_3, p_4) correspond to an *AND join* before D, synchronizing the completion of B and C. Finally, (p_5, p_7) represent an *XOR join* at E, merging the branches from D and F.

4. **Filtering infrequent constraints:** Infrequent or redundant ILP constraints may be removed to simplify the resulting model. In this example, all relations are consistent with the event log, so no filtering occurs.

5. **Final Model Construction:** The final BPMN model combines all identified splits and joins. It captures precisely the observed behavior in the log, providing a sound and precise BPMN representation. In our example: $A \to (((B \oplus C) \to D) \otimes F) \to E$, includes all discovered places $p_1, p_2, p_3, p_4, p_5, p_6, p_7$.

The Integer Linear Programming (ILP) Miner's strength lies in its unique ability to cover intricate, non-local control-flow patterns; however, its reliance on perfectly replaying the event log often leads to overfitting in noisy environments. The algorithm's computational complexity can also limit its scalability with large logs. To better understand how the ILP Miner works, think of it as a detective trying to figure out an instruction manual by only looking at the finished LEGO models. This process allows the miner to work backward from the final results, using a special kind of "math puzzle" to find the best rules that explain how the models were built.

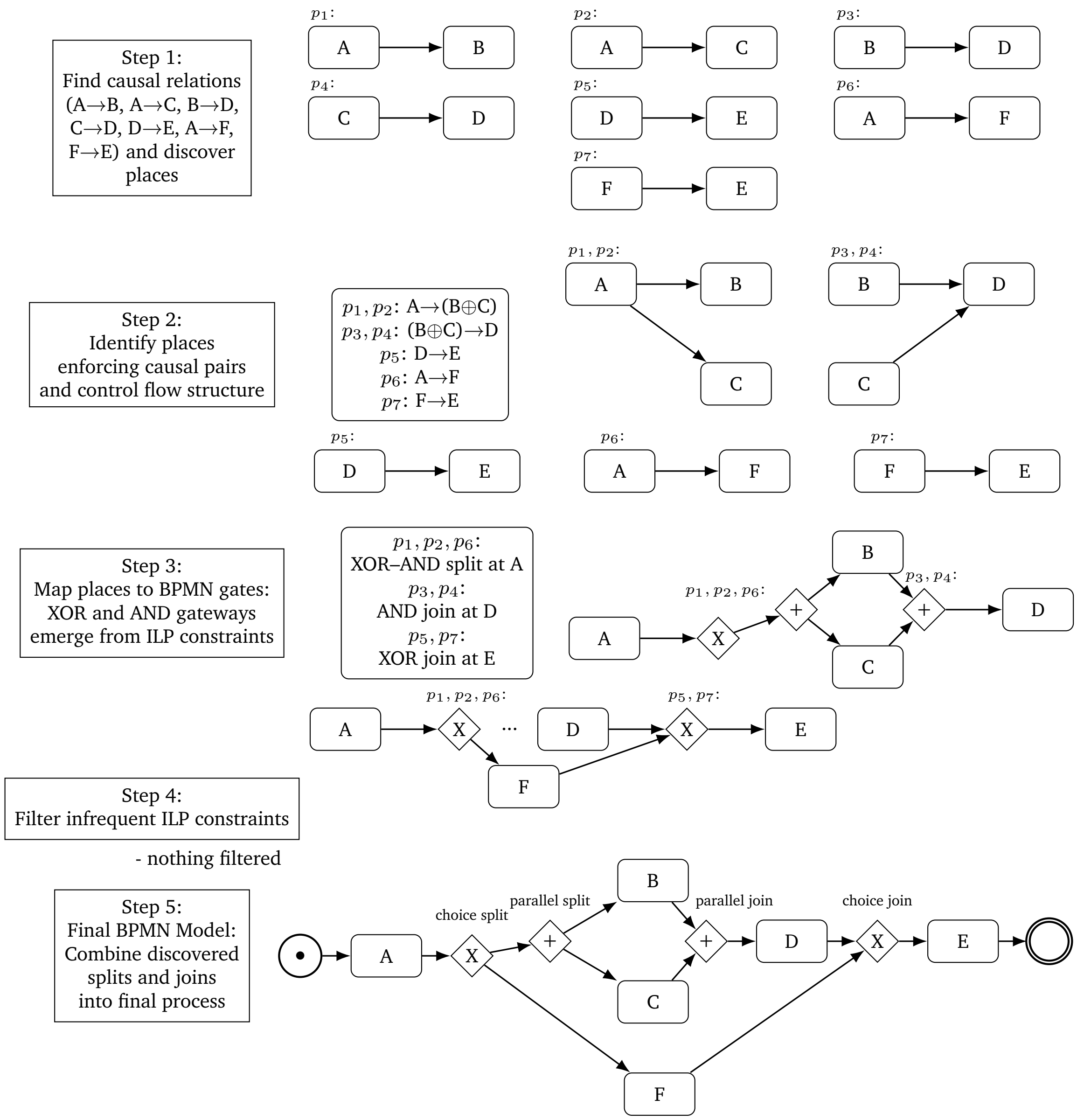

Figure 2.6: ILP Miner [135] example in BPMN in 5 steps

The **Split Miner** is a bottom-up algorithm that constructs a sound and simple BPMN model from an event log in five main steps [14], as exemplified in Figure 2.7. It achieves this by identifying and filtering sequential, concurrent, and loop-based relationships. Its working principle can be summarized as follows:

1. **Graph Construction:** The Split Miner first constructs a *Directly-Follows Graph (DFG)* that captures all directly-follow relations, loops, and then identifies concurrences in the event log. In Example 2.2.1, this yields the relations $(A \rightarrow B, A \rightarrow C, A \rightarrow F, B \leftrightarrow C, B \rightarrow D, C \rightarrow D, D \rightarrow E, F \rightarrow E)$. While no loops are detected, the bidirectional relation $B \leftrightarrow C$ indicates concurrency between B and C.

2. **Prune Concurrency:** Pairs of activities that directly follow each other in both directions are classified as *concurrent*, meaning that their order of execution is interchangeable. In this example, the concurrency between B and C is pruned, as shown in step 2 of Figure 2.7.

3. **Filtering:** The algorithm filters the Pruned DFG, based on a configurable frequency threshold to remove infrequent behaviors and noise. In Figure 2.7, all relations are frequent, so no filtering needs to be applied.

4. **Splits Discovery:** It classifies outgoing branching points in split gateways (*XOR* or *AND*) to capture choices and parallel executions. In this example, activity A introduces a combined *XOR–AND split*, meaning A can either start the parallel branch $(B \oplus C)$ or the alternative sequential branch F. This step corresponds to the partial BPMN model shown in Step 4 of Figure 2.7.

5. **Joins Discovery:** Finally, matching *AND joins* and *XOR joins* are added to merge incoming paths, ensuring the final model soundness. In the final BPMN model, B and C synchronize before D through an *AND join*, while the alternative path via F merges with the former flow through an *XOR join* before E. The resulting model captures the most frequent and logical consistent behavior observed in the log: $A \rightarrow (((B \oplus C) \rightarrow D) \otimes F) \rightarrow E$ providing a simple yet precise representation of the process.

The process of the Split Miner is analogous to a smart program creating a simple, easy-to-read map of your daily chores. It identifies the most common paths and relationships, prunes away redundant and rare deviations, and adds gateways to represent choices and parallel actions. Its main strength is its ability to produce highly interpretable BPMN models that balance key quality metrics, but it may produce unsound models, or underfit in complex or noisy scenarios, due to lacking a strict block structure and presenting ambiguous joins, from its heuristic gateway handling.

2.2.2 Event Data Generation

As real-life event logs are often scarce, sensitive due to privacy regulations [1, 2], or lack the necessary ground truth for a definitive algorithm evaluation, synthetic

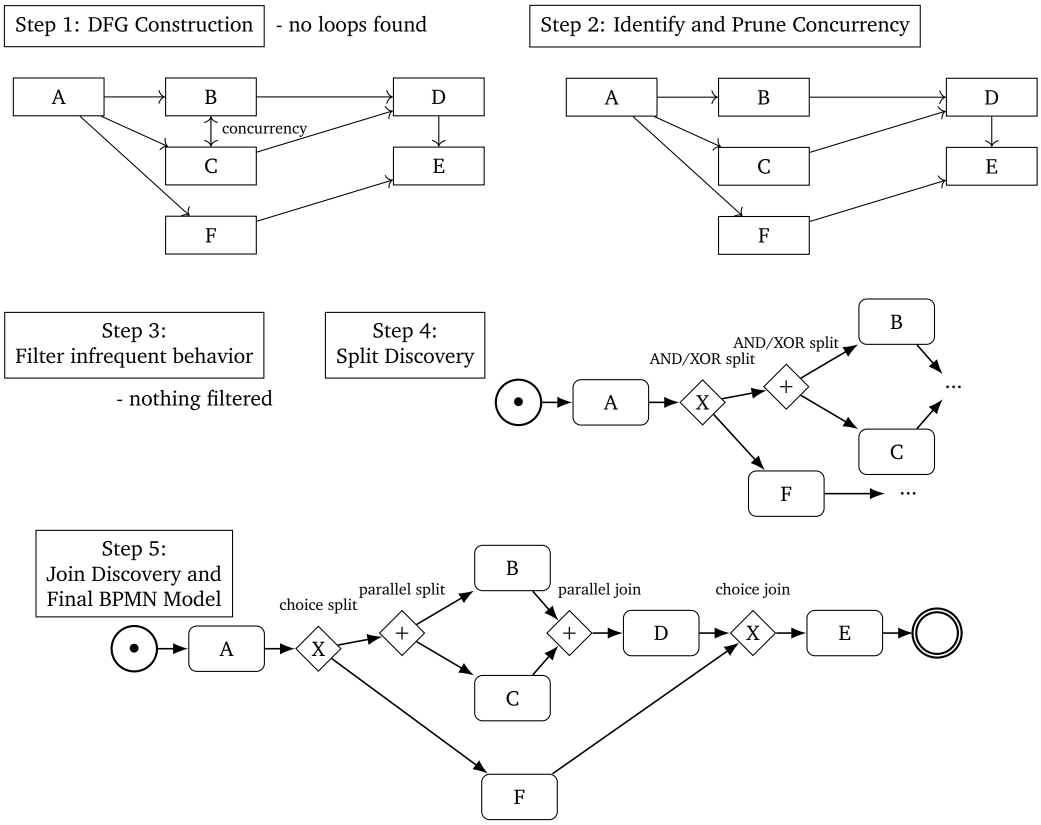

Figure 2.7: *Split Miner [14] example in BPMN notation in 5 steps*

event data generation emerges as a fundamental strategy to provide controlled, reproducible, and diverse datasets for rigorous algorithm evaluation [28, 98].

For the primary objective of benchmarking and algorithm evaluation, the field predominantly relies on simulation based on a defined process model. A *process model*, as in Equation (2.5), is a formal representation that describes the intended sequences and logic of activities within a system. Process model simulation creates synthetic logs by reproducing the specified process model, which provides a controllable ground truth essential for a definitive of algorithm assessment [29, 25, 98]. These formalisms typically capture control-flow logic, using either Petri nets [29, 56, 137], BPMNs [20, 28, 98], or Declare [21, 33]. Beyond controlled testing, the utility of generated data is a critical dimension of its evaluation. This is often measured by applying a process mining algorithm to the synthetic log and assessing the quality of the results using metrics, like fitness, precision, and F1-score [69], as presented for multiple tasks in Table 2.1.

Alternatively, to assess the utility of generated data, data-learned generative models, such as GANs [50, 101] or sequence models [140, 146, 147], create realistic synthetic data from statistical patterns and distributions from real event logs. This generated data is then used to train process mining or predictive approaches, which are subsequently evaluated against real datasets to quantify the utility of the generated data.

The field faces challenges in producing logs with sufficient data richness, particularly in incorporating timestamps and additional attributes [11, 69, 99, 101, 142, 148]. There are also difficulties in accurately generating complex control-flow dynamics, such as long traces and event repetitions [11, 50, 84, 99, 101, 106]. Additionally, there is a need to reduce the high manual modeling effort required for knowledge-driven approaches [56, 137, 148].

Consequently, while generation offers significant benefits for reproducible research, it still faces key challenges in the generalizability of its approaches. We need more sophisticated approaches that can generalize real-world complexities in a controlled manner, and still simplify the generation process, which motivates the development of generation approaches in this thesis.

2.3 Empirical Evaluation of Algorithms

Algorithm engineering [96] is a comprehensive research framework, which is structured around three philosophical dimensions: ontology, epistemology, and methodology. While *ontology* defines the nature of algorithmic reality and *epistemology* considers how we can know about algorithms, the *methodological* perspective focuses on how knowledge about algorithms can be systematically developed, extended, and validated. The methodological perspective addresses how knowledge about algorithms can be systematically enhanced, in four categories:

Knowledge Category	**Research Methods**
Algorithmic Tasks	Inductive methods applied to qualitative empirical data (e.g., case studies, focus groups, interviews).
Algorithm Designs	Deductive (from general principles), inductive (from specific instances), abductive (from anomalies), or analogy-based reasoning (transferring solutions from other domains).
Formal Knowledge (about Algorithmic Tasks and Algorithmic Designs)	Formal analysis methods, algorithm theory, theorems, and mathematical proofs (e.g., asymptotic analysis, correctness proofs).
Empirical Knowledge (about Algorithm Designs)	Hypothesis-driven research: exploratory, correlational, and experimental designs; controlled experiments with manipulation of factors.

Table 2.2: *Knowledge extension categories and corresponding research methods in algorithm engineering.*

Mendling et al. [96] conceptualize that the body of knowledge can be expanded on algorithmic tasks and algorithm designs, as well as theoretical and empirical insights about them, as listed in Table 2.2. On the one hand, *algorithmic tasks* define the specific computational problems to be solved, e.g., process discovery [128]. On the other hand, *algorithm designs* refer to the concrete construction and structure of the solution method, e.g., the Inductive Miner[77]. In contrast, *formal knowledge* entails the theoretical and mathematical truths about correctness and complexity of tasks and designs, established through analysis and proofs. For example proving the soundness of the Inductive Miner [77], or subprocess model properties as inheretance [72] expand formal knowledge about algorithms and process model. Finally, *empirical knowledge* results from observation and systematic experimentation with algorithm designs under varying conditions, focusing on the robustness and generalizability of measured results, e.g., a bechmark study of process discovery algorithms [13].

As shown in Table 2.2, each of these knowledge extension categories is associated with specific research methods. Typically, novel process mining algorithms results focus mainly on two of these categories: While developing novel process mining algorithms results in new algorithm design knowledge, highlighted in blue, their validity is often evaluated employing empirical research methods, highlighted in violet. Empirical research is established by developing and testing hypotheses and distinguishes between exploratory, correlational, and experimental research designs. This classification reflects three distinct purposes: exploratory designs generate insights and hypotheses, correlational designs measure associations between variables without manipulation, and experimental designs manipulate one or multiple factors under controlled conditions to establish causal relationships.

As the refinement of empirical knowledge increases, the technical sophistication, data requirements, and scientific rigor of the research increase simultaneously, as shown in Figure 2.8. This increase in methodological sophistication not only enables more precise and controlled investigations but also typically enhances the reliability and validity of the resulting knowledge.

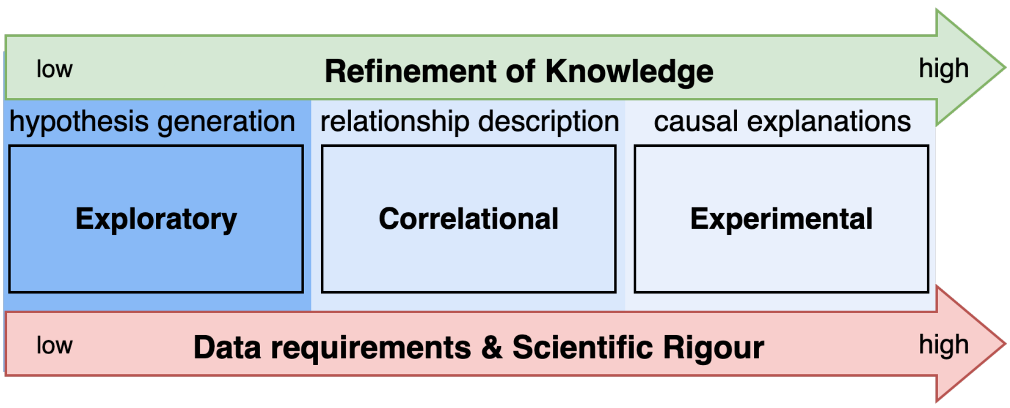

Figure 2.8: *In [96], advances in the refinement of knowledge in empirical research come with increasing technical sophistication, data requirements, and scientific rigor.*

2.3.1 Experimental Reliability

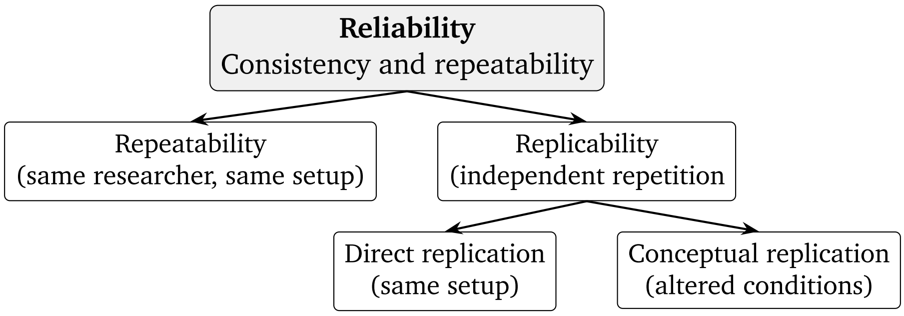

Figure 2.9: *Adapted from Rehse et al. [108], conceptual structure of reliability in experimental research.*

In extending the body of knowledge, it is crucial to ensure that the generated knowledge is both reliable and valid [96, 108]. ***Reliability***, defined as the consistency and repeatability of measurements [124]. Both generally improve with increasing technical sophistication in empirical research. Exploratory designs tend to prioritize breadth and insight, often trading off some reliability, while experimental designs emphasize control and precision, maximizing reliability.

As depicted in Figure 2.9, reliability differentiates between repeatability and replicability. *Repeatability* refers to the consistency of results when the same researcher reproduces a study under identical conditions and *replicability* to the consistency when the study is repeated independently, also called internal reliability. Replicability can be further divided into *direct replication*, which attempts to reproduce results with the same setup, and *conceptual replication*, which tests the robustness of findings under altered conditions, e.g., different datasets, instruments, or contexts. Reliability is a prerequisite to validity.

2.3.2 Methodological Validity

Validity refers to the extent to which a measure or method accurately captures the concept or phenomenon it is intended to represent [124]. Methodological validity in algorithm engineering concerns whether the research methods used to develop, evaluate, and generalize knowledge about algorithms are sound, rigorous, and appropriate for the type of knowledge being produced [96]. Rather than formal proofs of algorithms, this work focuses on the validity concerns in their empirical evaluation, as depicted in Table 2.2.

As illustrated in Figure 2.10, methodological validity spans the entire algorithm research pipeline, from problem formulation to knowledge claims, and can be categorized into several types that align with specific stages of research. Each validity type applies at the stage, where the corresponding research object is developed or evaluated. Color highlights central stages of empirically evaluated algorithm development, as the objects of analysis in this thesis.

At the level of algorithmic tasks, *ecological validity* assesses whether the tasks and evaluation setups reflect realistic problem contexts [96]. Moving to algorithm design, *design validity* ensures that the internal logic and structure of the design are coherent, justified, and explainable. During implementation, *implementation validity* evaluates whether the code faithfully realizes the intended design and behaves as expected. Key validity concerns arise in empirical evaluation:

1. *Internal validity:* Extent to which an experiment's outcomes can be confidently attributed to the manipulated variables, rather than confounding variables [108].
2. *Construct validity:* Whether the measurements and evaluation metrics accurately capture the intended properties of algorithms [36, 96].
   (a) *Soundness:* Metric only identifies properties that truly exist (no false positives).
   (b) *Completeness:* Metric identifies all instances of the target property (no false negatives).
3. *External validity:* Whether results generalize across datasets, domains, or populations [115].
4. *Conclusion validity:* Reliability of statistical inferences and the support they provide for hypothesized relationships [119].

Finally, at the stage of knowledge claims, deductive reasoning is critically assessed. *Justification validity* examines whether hypotheses or theorems are convincingly supported by theoretical reasoning. *Logical validity* evaluates whether the deductive steps, or syllogisms, used in proofs preserve truth, following Aristotelian principles [45].

Taken together, these validity types provide a comprehensive framework for systematically enhancing knowledge about algorithms, linking the type of knowledge being generated, i.e., task-level, design-level, formal, or empirical, to the appropriate methodological safeguards.

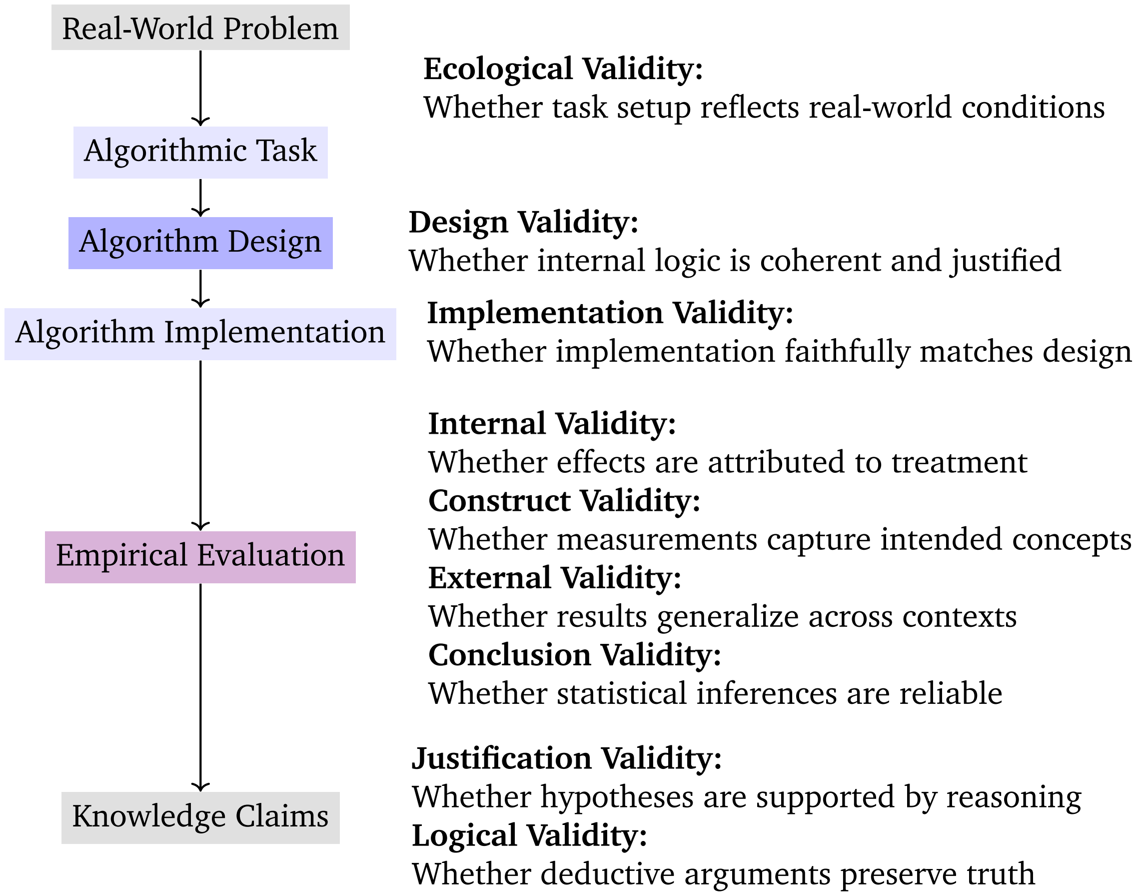

Figure 2.10: *Algorithm research pipeline and associated validity concerns.*

2.4 Representational Bias in Process Mining Evaluations

Bias in representation is a well-known concern in machine learning and statistics. [60] In those fields, *sampling bias* refers to a systematic mismatch between an available sample and the true underlying data distribution, which distorts inferences and reduces generalizability.

In contrast, in process discovery, the term *representational bias* has historically referred to the structural constraints of an algorithm's modeling formalism, as Petri nets, BPMN models, process trees, or Declare shape different inherent constraints in what process behaviors can be represented [127].

In this thesis, the concept is extended and redefined to address bias in event data representation used for evaluating process mining algorithms and the implications of this concept of representational bias on process mining evaluations is discussed.

2.4.1 Representational Bias

This thesis formalizes bias on data representation at multiple levels: local, global, and universal. Figure 2.11 illustrates the hierarchical progression of representative bias, from traces in a single event log (local), to an event log collection, i.e., a data set from a single process (global), and finally an event log collection representing the universe of all possible processes. These are interrelated and progressively broader in scope. Next, these multiple levels are formalized in colored blocks corresponding to their depiction in Figure 2.11.

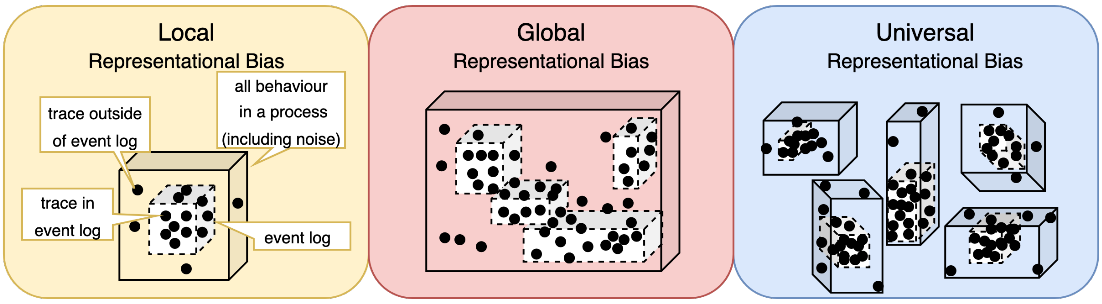

Figure 2.11: *Local, global, and universal representational bias in process mining research.*

Local Representational Bias captures deviations within a single event log, arising when the log fails to reflect the true diversity of possible executions of the underlying process. This may involve frequency deviations of activity sequences or their complete absence in the log, which compromises evaluations based on that log and produces misleading assessments of algorithmic performance [71, 47, 127]. It is the foundational concept upon which broader forms of bias are defined.

Formal Definition of Local Representational Bias

An event log is a finite set of traces $L = \{\sigma_1, \ldots, \sigma_n\}$ (cf. Equation (2.3)). Let $\mathcal{P}$ be a fixed underlying process, and let $\mathcal{L}(\mathcal{P})$ denote the generating function for possibly sampling an infinite set of finite event logs the process can generate under realistic conditions.
Formally, the generating function for event logs is:

$$L \sim \mathcal{L}(\mathcal{P}) \quad \text{with} \quad L \coloneqq \{\sigma_1, \ldots, \sigma_n\}, \quad \sigma_i \in \Sigma^*_{\text{loc}}, \quad 1 \leq i \leq n$$

where σ_i is drawn from the distribution $\pi_{\mathcal{P}}$, i.e., $\sigma_i \sim \pi_{\mathcal{P}}$ is an i.i.d. sample from the probability measure $\pi_{\mathcal{P}}$ induced by $\mathcal{P}$, denoting the process' trace distribution, over the set of traces Σ^*_{loc}.

For an observed event log $L_{\text{obs}} = \{\sigma_1, \ldots, \sigma_k\} \subset \Sigma^*_{\text{loc}}$, we define the empirical distribution as follows:

$$\pi_{L_{\text{obs}}}(\sigma) = \frac{1}{k} \sum_{i=1}^{k} \mathbb{1}\left[\sigma_i = \sigma\right], \quad \forall \sigma \in \Sigma^*_{\text{loc}},$$

where $\mathbb{1}[\cdot]$ denotes the indicator function. We define the **local representational bias** as the mismatch of the trace distribution of the log L_{obs} to the one of the process $\mathcal{L}(\mathcal{P})$, i.e.:

$$LRB(L_{\text{obs}}, \mathcal{P}) \coloneqq d_{\text{div}}(\pi_{L_{\text{obs}}}, \pi_{\mathcal{P}}),$$

where $d_{\text{div}}(\cdot, \cdot)$ denotes any divergence metric on probability measures (e.g., Jensen-Shannon divergence, Wasserstein-distance, etc.).

Global Representational Bias generalizes local bias to the level of a collection of event logs from a fixed underlying process $\mathcal{P}$. It occurs when the available set of multiple event logs $\mathcal{D}$ systematically deviates from $\mathcal{L}(\mathcal{P})$, thus extending the notion of local bias to an event log level sampling problem [70, 47]. This involves frequency deviations of complete concepts in event logs due to, e.g., seasonality [145], which compromises the validity of the resulting model and the algorithmic evaluation results. This is analogous to the previous local representational bias, but includes structural omissions and distortions across the entire collection of event logs [96, 108].

Formal Definition of Global Representational Bias

We define a set of sampled event logs $\mathcal{D} = \{L_1, ..., L_m\}$ where each event log is independently sampled from the generating function $\mathcal{L}(\mathcal{P})$ for a fixed process $\mathcal{P}$, i.e.:

$$L_j \sim \mathcal{L}(\mathcal{P}), \quad \text{with} \quad L_j = \{\sigma^j_1, \ldots, \sigma^j_{n_j}\}$$

with $\sigma^j_i \in \Sigma^*_{\text{glo}}$ traces from the global setting, and where $\sigma^j_i \sim \pi_{\mathcal{P}}$ with $\pi_{\mathcal{P}}$ is the trace distribution adopted from the local representational bias.
For an observed collection of event logs $\mathcal{D}_{obs} = \{L_1, ..., L_\ell\}$ with $N = \sum_{i=1}^{\ell} n_i$. denoting the total number of traces in $\mathcal{D}_{\text{obs}}$, we define the trace-aggregated empirical distribution as follows:

$$\pi_{\mathcal{D}_{\text{obs}}}(\sigma) = \frac{1}{N} \sum_{j=1}^{\ell} \sum_{i=1}^{n_j} \mathbb{1}\{\sigma^j_i = \sigma\}, \qquad \forall \sigma \in \Sigma^*_{\text{glo}},$$

where $\mathbb{1}[\cdot]$ denotes the indicator function, i.e., the relative frequency of trace σ among all traces in the observed collection. A **global representational bias** occurs when the trace aggregated distribution of the log collection $\mathcal{D}_{obs}$ does not adequately match the one of the generating process $\mathcal{L}(\mathcal{P})$. Formally:

$$GRB(\mathcal{D}_{obs}, \mathcal{P}) \;:=\; d_{\text{div}}\big(\pi_{\mathcal{D}_{obs}}, \,\pi_{\mathcal{P}}\big),$$

where $d_{\text{div}}(\cdot,\cdot)$ is any suitable divergence metric on probability measures (e.g., Jensen–Shannon divergence, Wasserstein distance).

Note that global representational bias builds directly upon local representational bias: local deviations in individual logs accumulate into global deviations across datasets. Furthermore, it serves as the basis for the next and final level of abstraction.

Universal Representational Bias further generalizes the concept to theoretically encompass all possible processes. It occurs when a collection of event logs $\mathcal{D}$ fails to represent the universe $\mathcal{L}^*$ of all possible event logs $\mathcal{L}(\mathcal{P}^*)$ generated by any conceivable process $\mathcal{P}^*$. This is the broadest form of representational bias and captures fundamental epistemic limitations of empirical evaluation [60, 96].

Formal Definition of Universal Representational Bias

Let $\mathcal{P}^*$ denote the set of all possible processes. Define the universe of all possible event logs as:

$$\mathcal{L}^* = \bigcup_{\mathcal{P} \in \mathcal{P}^*} \mathcal{L}(\mathcal{P}).$$

We define the likelihood of a process with $P \sim \Pi$, where Π denotes the probability measure over process, i.e., a process prior. The universal trace distribution is therefore given as:

$$\pi_U(\sigma) = \int_{\mathcal{P}^*} \pi_{\mathcal{P}}(\sigma)\; d\Pi(\mathcal{P}),$$

where $\pi_{\mathcal{P}}$ denotes the probability measure of a trace given a specific trace $\mathcal{P}$. Therefore, the sampling process of an event log is defined as:

$$\mathcal{L}_U : \text{sample } \mathcal{P} \sim \Pi, \text{ then } L \sim \mathcal{L}(\mathcal{P})$$

A dataset of m logs is drawn from the universe as :

$$L_j \sim \mathcal{L}_U, \quad \text{with} \quad L_j = \{\sigma_1^j, \ldots, \sigma_{n_j}^j\}$$

For an observed collection of event logs $\mathcal{D}_{\text{obs}} = \{L_1, \ldots, L_r\} \subset \mathcal{L}_U$, with total number of traces $N = \sum_{i=1}^{\ell} n_i$, the empirical universe-level trace distribution is:

$$\pi_{\mathcal{D}_{\text{obs}}}(\sigma) \;=\; \frac{1}{N} \sum_{i=1}^{r} \sum_{j=1}^{n_i} \mathbb{1}\{\sigma_i^j = \sigma\}, \qquad \forall \sigma \in \Sigma^*_{\text{uni}},$$

where $\mathbb{1}[\cdot]$ denotes the indicator function. The **universal representational bias** quantifies the divergence between the empirical distribution $\pi_{\mathcal{D}_{\text{obs}}}$ and the aggregate universe distribution π_U:

$$URB(\mathcal{D}_{\text{obs}}, \mathcal{P}^*) \;:=\; d_{\text{div}}\big(\pi_{\mathcal{D}_{\text{obs}}},\, \pi_U\big),$$

where $d_{\text{div}}(\cdot, \cdot)$ is a suitable divergence metric for probability measures (e.g., Jensen–Shannon divergence, Wasserstein distance).

Universal representational bias subsumes both local and global bias, and is the central focus of this thesis (see Section 1.1). Generalizable evaluation of process mining algorithms requires controlling for universal representational bias as it concerns finding the limitations, where evaluation results can hold across all representative processes, as a subset of the theoretical possible processes, and generalizable conditions [96, 108, 107]. We acknowledge that, especially when researching the theoretical universe of all possible processes, not all of them might be realistic or relevant in practice. For this purpose, $\mathcal{L}^*$ should be defined per evaluation by controlling relevant features and characteristics in the particular process mining task. This thesis approaches mitigating universal representational bias via an exploratory data generation of controlled samples from $\mathcal{L}^*$.

Representational bias manifests at local, global, and universal levels (cf. Figure 2.11).

- **Local representational bias (LRB)** quantifies deviations within a single event log L_{obs} relative to the distribution π_P of the underlying process P [127].

- **Global representational bias (GRB)** aggregates deviations across a collection of event logs $\mathcal{D}_{\text{obs}}$ from a fixed process P, comparing the empirical distribution $\pi_{\mathcal{D}_{\text{obs}}}$ to π_P.

- **Universal representational bias (URB)** generalizes this notion to the full universe of possible processes $\mathcal{P}^*$ and their logs $\mathcal{L}^*$, comparing the empirical dataset $\mathcal{D}_{\text{obs}}$ to the aggregate universe-level distribution π_U.

Formally, the levels satisfy the following hierarchy:

$$LRB(L_{\text{obs}}, P) \;\subseteq\; GRB(\mathcal{D}_{\text{obs}}, P) \;\subseteq\; URB(\mathcal{D}_{\text{obs}}, \mathcal{P}^*),$$

Universal representational bias integrates the preceding notions and defines the scope for generalizable evaluation, which is the focus of this thesis. It threatens the validity of process mining evaluations by limiting generalizability to unseen processes and conditions, and by introducing dataset artifacts that can distort algorithmic performance [96, 120, 108]. The remainder of this thesis explicitly addresses this type of bias, and henceforth *representational bias* refers to universal representational bias unless otherwise stated. These effects underpin many of the challenges discussed in the next subsection, where links to process mining crimes and threats to construct, internal, external, and conclusion validity are systematically analyzed.

2.4.2 Challenges and Implications

The evaluation of process mining algorithms is a fundamental challenge, as documented by systematic reviews and benchmark studies [8, 13, 138, 107, 71, 108]. From a methodological perspective, validity and reliability threats in process discovery have been conceptualized as *process mining crimes*, i.e., unintentional but systematic mistakes that undermine empirical evaluation results [107]. Several of these crimes are directly linked to the challenge of data quality and representational bias. Table 2.3 presents a conceptual mapping of process mining crimes to specific validity threats and highlights whether these crimes can be connected to representational bias. A "•", represents a *directly connected* crime, i.e., one inherently caused by representational bias. Whereas "○" represents a *indirectly connected* crime, i.e., a consequence or amplifier of representational bias, but which might not primarily be a data problem itself, e.g., using selective metrics can hide flaws caused by unrepresentative data, but is not a data selection concern itself. Finally, "–" represents crimes, which are not connected to representational bias. This mapping builds on prior work on methodological validity in process discovery [96, 108] and integrates empirical findings [107] with theoretical considerations. The table also links each crime category to construct validity, internal validity, external validity, and conclusion validity, which are key to empirical evaluation [96].

Implications on Process Mining Crime: Several key insights emerge from Table 2.3. Category 1, *Using the wrong evaluation data*, shows the strongest direct connection to representational bias. This highlights that the selection of datasets is a foundational methodological concern [70, 108]. Here, the lack of justification for dataset choice (1a), *reliance on micrologs* (1b), *evaluation with simplified simulations* (1c), and *misleading logs* (1d) all directly compromise representativeness. These practices result in evaluations that fail to capture the diversity and complexity of real-world processes, thereby weakening both construct validity and external validity. Categories 2, *Misleading quality assessment*, and 4, *Incomplete*

Table 2.3: Process mining crimes' connection to validity concerns and representational bias.

Crime category and crime	Short Explanation	Validity Concerns	Rep. Bias
1 Using wrong evaluation data			
1a Choice without justification	Data choice not justified	Construct, External	•
1b Micrologs not representative	Distorting generalization	Construct, External	•
1c Simplified simulated logs	Lacks real complexity	External, Construct	•
1d Misleading logs	Distorts measurement	Construct, Internal	•
2 Misleading quality assessment			
2a Selective metrics	Omits dimensions	Construct, Conclusion	○
2b Matching metrics to desired outcomes	Tailored metrics	Construct	○
2c Only partial dimensions	Ignores relevant dimensions	Construct, Conclusion	○
3 Scientific inaccuracies			
3a No quality degradation tests	Misses relevant effects	Construct, Internal	○
3b Creative result accounting	Manipulates results	Conclusion	–
3c Claims without verification	Lacks empirical support	Conclusion	○
4 Incomplete evaluations			
4a No significance indication	Lacks statistical reliability	Conclusion	○
4b No assumptions on noise	Ignores confounders	Internal	○
4c No incremental testing	Effects not isolated	Internal, Construct	○
5 Improper comparisons			
5a No proper comparison	Limits generalizability	Internal, External	○
5b Only self-evaluation	Risks overfitting claims	External, Construct	○
5c Unfair competitor evaluation	Bias distorts conclusions	Internal, Conclusion	–
6 Missing information			
6a Missing hardware specs	Reproducibility threatened	Conclusion	–
6b Missing software specs	Reproducibility threatened	Conclusion	–
6c Missing individual measures	Completeness unverifiable	Construct, Conclusion	○
6d Relative numbers only	Lack of absolute context	Construct, Conclusion	○

evaluations, exhibit indirect connections, indicating that the lack of representative and diverse data can propagate methodological shortcomings. For instance, *selective use of quality metrics* (2a–c) or *incomplete evaluations* (4a–c) can exacerbate the effects of unrepresentative data by hiding deficiencies that only emerge under diverse conditions. Similarly, *Scientific inaccuracies* (3a, 3c) and *Improper comparisons* (5a, 5b) propagate biases when conclusions are drawn from narrow or unrepresentative samples. Even *Missing information* (6c, 6d), though not inherently a data problem, can prevent researchers from assessing whether results depend on specific datasets or whether findings generalize beyond them. In contrast, crimes such as *creative result accounting* (3b), *unfair competitor evaluation* (5c), *or missing technical specifications* (6a, 6b) primarily threaten reproducibility or fairness, but are not directly tied to dataset representativeness. Taken together, this analysis highlights that representational bias not only affects the initial choice of datasets but also interacts with other methodological shortcomings, amplifying threats to validity across the evaluation pipeline.

Implications on Validity: Representational bias compromises multiple dimensions in terms of validity in empirical evaluation [96, 108, 70].

- **Construct validity** is threatened when evaluation metrics do not align with their intended goals, for example, when process discovery evaluations rely solely on fitness measured against biased or overly simplified event logs. Such logs may fail to capture the real complexity of operational processes, leading to misleading conclusions about algorithm performance [96].

- **Internal validity** is compromised when confounding dataset properties caused by representational bias, such as noise levels, incompleteness, or preprocessing choices, are not controlled. This can result in incorrect causal attributions, where observed differences in performance are due to dataset artifacts rather than genuine algorithmic improvements [108].

- **External validity** is fragile when benchmarks systematically underrepresent the diversity of real-world operational processes. Representational bias in curated datasets, particularly in streaming and online process mining, creates unrealistic evaluation settings due to the scarcity of realistic event streams, limiting the generalizability of conclusions to actual operational environments [70, 87, 4, 113, 108].

- **Conclusion validity** is equally at risk when statistical analyses and evaluation claims are drawn without sufficient rigor, especially when evaluations are based on biased datasets that do not capture process variability. This includes omissions such as failing to report significance tests or selectively publishing results that fit expectations, which can produce unfounded generalizations about algorithm effectiveness [96].

These threats collectively illustrate why addressing representational bias through careful dataset selection, diverse log generation, and robust metric design is central to ensuring methodological rigor in process mining evaluations.

Addressed Gaps: Presented validity concerns and discussed process mining crime implications demonstrate a systemic problem, rooted in the limitations of existing benchmarks. These motivate the primary contributions of this thesis, which address the following gaps:

- Despite initiatives like the *BPI Challenges*[2], *the Process Discovery Challenge* [31] or by Costa et al. [37], benchmarks often cover only a narrow subset of process characteristics, leaving representational bias unresolved.

[2]`https://www.tf-pm.org/resources/logs`

- Algorithm robustness depends critically on log diversity [107, 139], which necessitates systematic studies under controlled variations in log properties such as noise, incompleteness, and structural complexity [10, 15, 139].
- The difficulty of systematically exploring the full event-log design space using traditional methods remains a barrier to comprehensive algorithm testing [24, 94].
- Deficiencies persist in creating controlled, robust, and reproducible datasets required to strengthen the validity of empirical evaluations [90, 92, 93, 115].
- A rigorous framework is lacking to reliably explain the impact of specific log characteristics on algorithm evaluation measurements [91].

Chapter 3

Contributions

Nothing comes without its world.

– Donna Jeanne Haraway

Haraway's observation [58] reminds us that knowledge is always situated within the contexts and practices that generate it. In process mining, new algorithms cannot be meaningfully evaluated in isolation, but only in relation to the data, benchmarks, and methodological choices that shape their performance [113]. Similarly, the contributions presented in this chapter are interconnected elements of a broader methodology, which refers to a structured system for studying process mining methods, in this case, process discovery algorithms. [96]

The six systematic workflow steps for generation of empirical knowledge, according to algorithm engineering [96] are illustrated in Figure 3.1. Next, each of these steps is presented and discussed in the general scope of this thesis:

(1) *Developing hypotheses*, which derive from and operationalize the research questions to be investigated. The overarching hypothesis of this dissertation, derived from the research question in Chapter 1 and literature observations about impacts of data characteristics in process mining evaluations (see [8, 13, 68, 71, 107, 108, 113, 138]), is the following:

General Hypothesis: Because event data characteristics systematically impact algorithmic evaluation results, a data-driven evaluation framework will reliably identify and explain their impact on the evaluation measurements and validity of process discovery algorithms.

(2) *Deriving a research design*, specifying how these hypotheses will be tested. In this thesis, we adopt a data-driven empirical design that integrates data characterization, intentional data generation, and explainable evaluation to systematically test how event data characteristics impact algorithm evaluation measurements, corresponding to objectives in Section 1.1.

(3) Building an *implementation for instrumentation*, enabling systematic observation and measurement of algorithm behavior. To this end, we implement open-source instrumentation frameworks that operationalize data characterization, controlled experimentation, and explainable impact analysis across multiple process discovery algorithms.

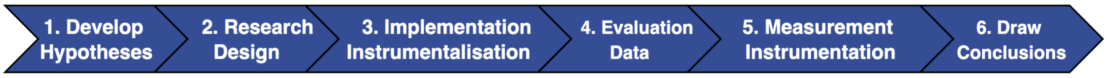

Figure 3.1: *Conceptual workflow of empirical steps from algorithm engineering [96]*

(4) Choosing *evaluation data*, providing the experimental setting in which the instrumentation will operate. As a central topic, we employ both available real-world and synthetically generated event data. Employing *GEDI* [92] and *iGEDI* [90], we ensure controlled variation of data characteristics, generalizability, and reproducibility in evaluating process discovery algorithms.

(5) Conducting *measurement and instrumentation*, where performance data is collected under controlled conditions. In *SHAining* [91], we perform systematic experiments to capture algorithm evaluation measures as fitness, precision, model size, and execution time for multiple algorithms. Controlled hardware configurations, as well as intentional variation of event data characteristics, ensure reproducible results for quantified characteristic impact explanations.

(6) *Drawing conclusions*, in which results are analyzed to contribute new empirical knowledge to the field. To interpret the empirical findings of this thesis, we relate our observations on algorithm robustness, validity, and generalizability back to our general hypothesis. To confirm the *General Hypothesis*, we investigate a set of three hypothesis aligned with the objectives in Section 1.1. The results of this analysis are briefly discussed in the following sections of this chapter.

Ultimately, our results for each objective confirm that the proposed systematic, data-driven evaluation framework reliably identifies and explains how event data characteristics affect process discovery algorithm evaluation measurements in a generalizable way for an underlying data-generating process.

In Figure 3.2, we depict which empirical steps, previously presented, were executed in individual contributions. Colors associated with each contribution represent the role they play in the general scope of this thesis. Star figures after a contribution, represent points of produced new insights. These include what feature value combinations are feasible after *GEDI* [92] and *Know Your Streams* [93], as well as validated statistical claims about the impact of event log feature values on particular process discovery algorithms' evaluation measurements after *SHAining* [91]. We position the included publications in the following two dimensions: The vertical dimension classifies the works in one of three interdependent methodological principles, presented in Section 1.1, and presented in different background shades in the figure:

1. **Justification** — While *DROPP* [18] improves data analysis by preserving structure-aware characteristics after dimensionality reduction. It forms the justification for this thesis' investigation in the context of sequential data by demonstrating the impact of structure preservation on reconstruction error,

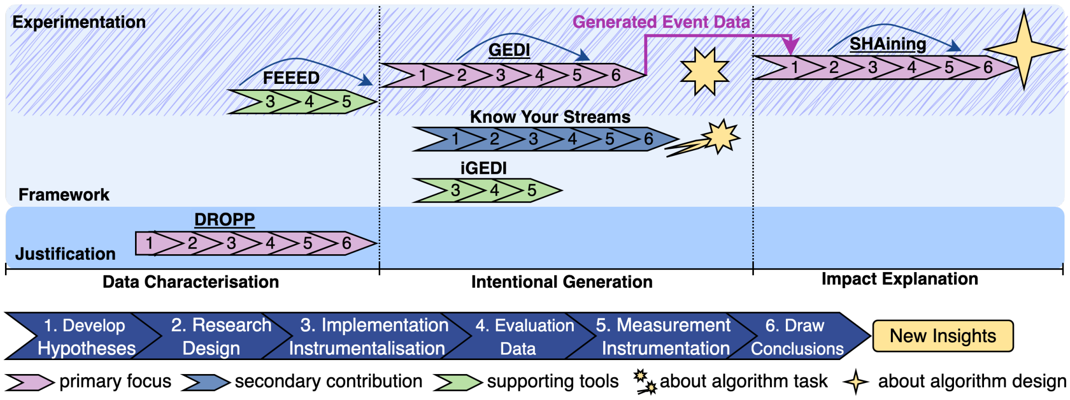

Figure 3.2: *Conceptual workflow of the dissertation's approach, showing cumulative empirical steps (x-axis) and paper-specific components (y-axis, cf. Section 1.1).*

as well as proving the feasibility to identify and preserve these invariant structural characteristics.

2. **Framework** — This dissertation provides a reusable framework for robust empirical process mining research. It unites the operationalization of structure-aware event log characterization in the supportive tool from *FEEED* [94], the generation of diverse and intentional event logs in *GEDI* [92], and the systematic evaluation of event data characteristics impact on process mining algorithms in *SHAining* [91]. Furthermore, the supportive interactive ready-to-use web application in *iGEDI* [90], and the extension to realistic event streams in *Know Your Streams* [93] enable extensions to other process mining tasks, including in online scenarios.

3. **Experimentation** - A subset of the contributions that are part of the previously presented framework, belong to the overarching experiment. Designing and executing controlled experiments of this thesis includes multiple design decisions. Therefore, blue arrows in Figure 3.2 illustrate how the divergent exploration and convergent synthesis of feature dimensions converge at key design junctures, which guide methodological choices such as, the selection of representative features after *FEEED* [94] for event data generation, the filtering of feasible event logs in *GEDI* [92], and the aggregation of feature impact values in *SHAining* [91] to interpretable evaluation results.

The horizontal dimension, classifies the presented works in relation to Objective 1-3, as presented in Chapter 1. The rest of this chapter presents how a series of key papers collectively construct the described full-cycle methodology for explainable and robust algorithm evaluations, with each contribution serving a distinct, yet interconnected, component within the framework, and briefly discusses each of the empirical workflow steps.

3.1 Structure-aware Data Characterization

Objective 1 of our research scope, as presented in Section 1.1, is to develop a structure-aware method for data characterization. For this, we consider:

> **Hypothesis 1** *Structure-aware data characterization methods reliably capture and preserve intrinsic properties of sequential and event data across domains, forming a robust foundation for subsequent algorithm evaluation.*

We start by addressing the foundational problem of dimensionality reduction techniques, which often fail to preserve the intrinsic structure of ordered data. The paper ***DROPP**: Structure-Aware PCA for Ordered Data, a General Method and Its Applications in Climate Research and Molecular Dynamics* [18] handles this problem by following the previously presented steps of an experimental design approach. It explicitly hypothesizes (1) that a structure-aware method will outperform traditional dimensionality reduction techniques. The experimental research design compares (2) our implemented *DROPP* approach, using Gaussian kernels to preserve the order structure in dimensionality reduction, (3) against PCA [44], ICA [61], and tICA [100] on synthetic and real datasets across various applications (4) in terms of reconstruction error and similarity of principal components (5) to demonstrate its superior performance (6). This foundational work on data characterization laid the groundwork for the envisioned data-driven approach by highlighting the importance of preserving the sequential data's inherent structure.

Figure 3.3 depicts two example sequential process models, representing structurally similar event data sets, with e.g., number of variants equal to 1 each, yet stemming from different domains – healthcare and baking. While the realities of both processes are likely to differ, data mining algorithms are likely to approach them in very similar ways [15]. Figure 3.4 shows two additional example process models, structurally different from the previous ones in Figure 3.3. The additional examples are also structurally similar to each other, with e.g., the number of variants strictly higher than one, as both contain a parallel split after their first activity (see Section 2.2.1). Both *healthcare* process models on the left, and both *baking* process models on the right, are topologically different to each other, and may be approached fundamentally differently by data mining algorithms. These figures exemplify impactful structural features of event data extracted by *FEEED* [94] beyond domain knowledge.

Furthermore, to address the need to robustly capture interpretable structural data properties from event data, we present ***FEEED**: Feature Extraction from Event Data* [94], which computes interpretable meta-features from event logs at multi-

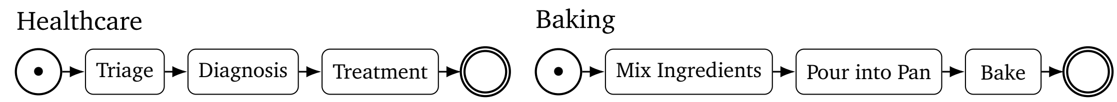

Figure 3.3: *Sequential processes in healthcare and baking*

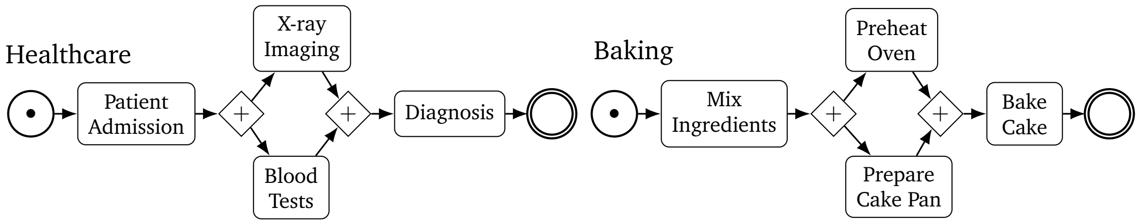

Figure 3.4: *AND-branching processes in healthcare and baking*

ple granularities. *FEEED* [94] hypothesis (1) and research design (2) are derived from further literature identifying domain-independent intrinsic properties of an event log [15, 16, 143]. The supportive work in this contribution is a critical piece of the framework, as it provides the implementation (3) for posterior algorithm analysis and benchmarking. It contributes to operationalizing evaluation data characteristics as meta-features (4), enabling similarity comparisons between event datasets (5). Thus, it offers a method for reliably capturing structural event data characteristics.

Both papers confirm Hypothesis 1 by measuring and preserving structure-aware characteristics. While *DROPP* [18] justifies the preservation of the invariant structure of the underlying data-generating process across dimensions during analysis, *FEEED* [94] forms the first part of the framework by capturing domain-independent, yet human-interpretable, representative structural features from event logs.

3.2 Intentional Event Data Generation

For Objective 2 of creating frameworks for intentional event data generation to mitigate representational bias, we formulate a second hypothesis.

Hypothesis 2 *Intentional event data generation enables systematic creation of diverse, feature-controlled datasets that mitigate representational bias and improve the validity of process mining algorithm evaluation.*

After measuring structure-aware meta-feature values for 25 publicly available real-world event logs in *FEEED* [94], we analyze the inter-feature correlation between features and between semantically associated feature groups, such as activity-, trace-, and entropy-based features.

Figure 3.5 shows the complete correlation matrix using Pearson correlation between feature values of 25 public available event logs. Features are clustered across all groups using hierarchical clustering with single linkage distance [83], revealing overall similarity patterns. In the axes, features from specific groups, as defined in [94], are represented by different colors. Dots indicate feature types as defined in [94], with larger dots marking representative features. The selected features for the rest of the experiment are shown in Figure 3.6.

The correlation with a full list of features can be found in Figure 4.1. Observing the light-colored squares close to the diagonal in Figure 3.5, we can identify multiple clusters. Colored blocks of feature groups along the axes reveal that features in the same group often correlate with other features in the same group. Nevertheless, almost no groups of features, besides *eventropies*, correspond completely to the discovered clusters, as colored blocks along the axes often have more than one appearance per color.

We applied a greedy feature selection procedure to obtain a representative subset of features, as elaborated in *SHAining* [91]. The method iteratively adds features that contribute maximal diversity while minimizing redundancy, based on correlation analysis of feature values for the 25 real-world event logs, most used in evaluations, across feature groups. This selection process continues until a defined threshold is reached, which in this case yielded eight meta-features using the elbow method [126]. Notably, the selected features resulting from this procedure, depicted as bigger dots in Figure 3.5 and specified in Figure 3.6, often correspond to one bigger cluster, depicted as a light square close to the diagonal. Selected features, employed for posterior steps in the overarching experiment of this dissertation, as well as their correlation matrix, are shown in Figure 3.6. These selected features are critical because they serve as the essential quantitative variables that enable both the intentional data generation (Objective 2) and the quantified explainability (Objective 3)in the overarching methodology.

The limitations of current benchmarks in process mining, namely their lack of diversity and controllability, created a significant gap in available choices for the evaluation data selection phase (4). This gap justifies the explorative data generation approach of our experiment and framework. ***GEDI**: Generating Event Data with Intentional Features for Benchmarking Process Mining* [92] directly addresses this gap by introducing a method for intentional data generation on multiple simultaneous dimensions. It develops the hypothesis that intentional data generation can overcome benchmark event data generalizability and diversity limitations (1), testing the feasibility of feature value combinations, assessing the coverage of the feature space, and investigating the changes in metric correlation tests by considering the exploratory data sample (2). Implementation involves building the *GEDI* [92] framework using hyperparameter optimization on parameterizable event data generators (3), which is directly reused in the following papers, as depicted by the purple arrows to *iGEDI* [90], and *Know Your Streams* [93] in Fig-

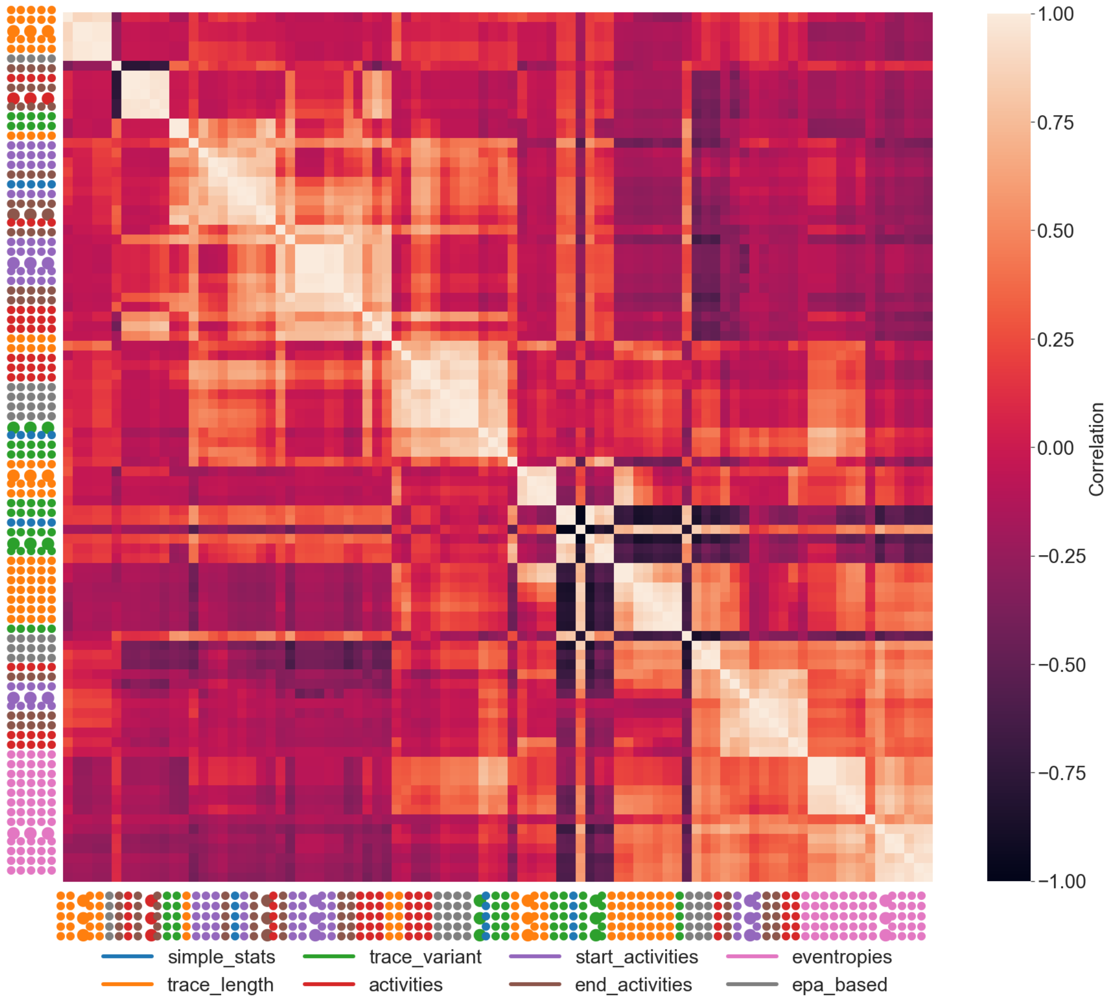

Figure 3.5: *Clustered correlation matrix using hierarchical clustering with single linkage and Pearson correlation as distance [83] between event data features [94]*

ure 3.2. Its core contribution is enabling the generation of intentional event data with controllable characteristics (4), which is explicitly employed in the next step, as depicted by the dotted purple arrow between *GEDI* [92] and *SHAining* [91] in Figure 3.2. Measurement validates the feasibility and coverage of the event data feature space by the generated data (5). Conclusions establish *GEDI*'s effectiveness in creating diverse and tailored evaluation datasets for rigorous experimentation (6), thus sufficiently confirming Hypothesis 2.

Building upon *GEDI* [92], the tool ***iGEDI*** *[90]: interactive Generating Event Data with Intentional Features* [90] addresses the need for interactive exploration of the feature space. It supports the implementation phase, providing instrumentation to generate data (3), contributing to operationalizing evaluation data characteristics (4) through a user-friendly interface on an web application. This enables researchers to design targeted benchmarks more easily, thereby streamlining the process of deriving a research design.

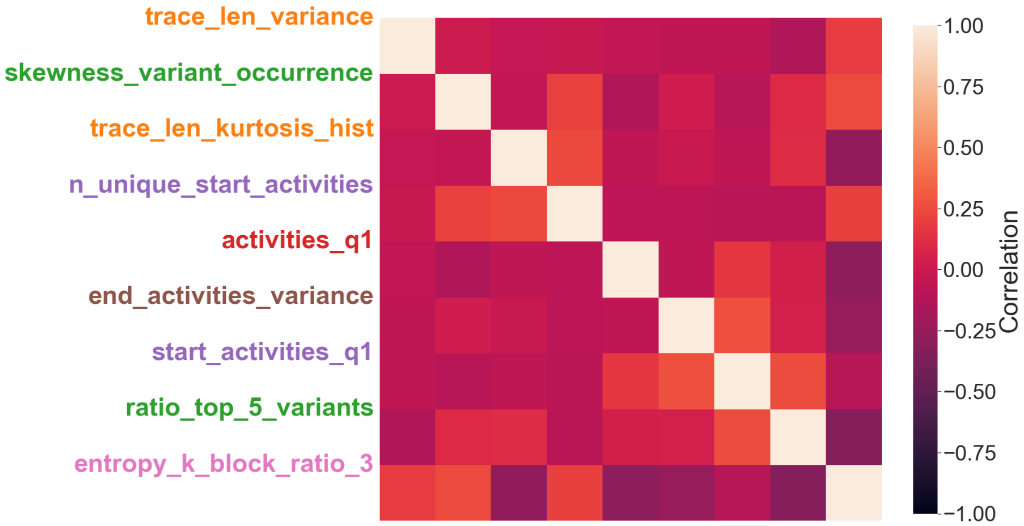

Figure 3.6: *Correlation plot for selected features from Figure 3.5*

Furthermore, the inadequacy of current evaluation practices for streaming process mining also shows a significant gap in online scenarios. The contribution ***Know Your Streams****[93]: On the Conceptualization, Characterization, and Generation of Intentional Event Streams* [93] addresses this issue by extending the data-driven methodology to the streaming context. It addresses multiple methodological steps simultaneously, hypothesizing about the inadequacy of current evaluation practices (1). The research design combines a literature review for identifying characteristics not well understood in streaming contexts (2), as well as the implementation of the *Stream of Intent* prototype generator, which produces realistic event streams (3). Data generation for evaluation focuses on creating realistic event streams with intentional characteristics (4). Measurement defines and operationalizes stream characteristics like temporal dependencies and out-of-order events (5). Conclusions establish a conceptual framework for event streams and demonstrate the prototype's effectiveness (6). This contribution demonstrates the adaptability and generalizability of the overarching methodology beyond process mining algorithm paradigms, from static to dynamic.

Overall, these contributions enable the systematic generation of diverse feature-controlled datasets to mitigate representational bias. Confirming Hypothesis 2, *GEDI* [92], *iGEDI* [90] and *Know Your Streams* [93] present scalable essential parts of the framework, which can be used to improve the validity of process mining algorithm evaluations in various process mining tasks. Particularly, *GEDI* [92] generational capabilities are leveraged in the experimental design of this thesis. This is further elaborated in the next section.

3.3 Measuring the Impact of Data Characteristics

Finally, Objective 3 targets quantifying and explaining the impact of event data characteristics on algorithm evaluation measurements. They are investigated through the following:

Hypothesis 3 *Interpretable aggregation of feature impact values through* SHAining *enables systematic explanations of how event data characteristics influence algorithmic evaluation results, supporting generalizable and reliable conclusions.*

Intentional event data generation can systematically create diverse, feature-controlled datasets that reduce representational bias and enhance the validity of process mining algorithm evaluation.

The challenge of understanding how specific event log characteristics impact the performance of process mining algorithms is the main goal of the contribution ***SHAining*** *on Process Mining: Explaining Event Log Characteristics Impact on Algorithms* [91]. This contribution serves as the culmination of the entire framework and directly implements the complete empirical knowledge generation experiments (1-6) by leveraging the feature computation of *FEEED* [94] as well as, data generation capabilities of *GEDI* [92]. It develops hypotheses (1) about how event log characteristics impact algorithm performance, such as "trace length variance has a higher impact on ILP miner's fitness than the other selected features". The research design (2) involves feature combinations, event log generation, process discovery execution, culminating in Shapley analysis and correlation tests. Implementation (3) involves instrumenting contributions presented in the previous framework contributions, as well as multiple process mining algorithms, and the overarching Shapley value analysis.

Evaluation data (4) comprises over 22,000 intentionally generated logs from *GEDI* [92] and 25 real event logs from BPI Challenges[3], as shown in Figure 3.7. Starting from 8 selected features, we generate an 8-dimensional space with 10 values per feature, which results in 80 initially generated event logs. To employ Shapley values, we require event logs, displaying each feature value singularly as well as all possible combinations of feature values of disjunct features simultaneously, resulting in this case in three levels. As shown in the feasibility evaluation in *GEDI* [92], feature values of disjunct features may stand in contradiction to each other. For example, the value for maximum number of activities in a trace cannot be lower than the value of minimum number of activities in a trace. For this reason, 58,302, 1-3 level feature value combinations result in feasible event logs. On these feasible event logs 3 algorithms are applied. As hardware constraints might not be manageable by all algorithms for all event datasets, approximately 58% of

[3] https://www.tf-pm.org/resources/logs

possible executions actually yield a valid output, resulting in 101,445 process discovery output models. All executions, which yielded valid output, are evaluated in terms of 6 process discovery measures. Due to soundness, 529,416 of 608,607 yield actual measurements. Each evaluation measure is regarded separately by the Shapley value analysis. This analysis employs combinatorial lattices consisting of combinations of measurement results and corresponding feature values. Whenever a value of a combinatorial lattice is missing, that Shapley value is skipped. Accumulated infeasibility propagates restrictively at this point. In our setup, we end up with 313,176 Shapley value results from 22,000 event logs to consider in the correlation tests. Although feasible Shapley values only cover under 36% of all possible values, Figure 3.8 shows the coverage of our 22,000 involved generated event logs in orange, compared to real event logs in blue. Compared to scarce real event logs, we observe that the generated event logs cover and enrich the feature space, providing a more generalizable samples than in prior contributions.

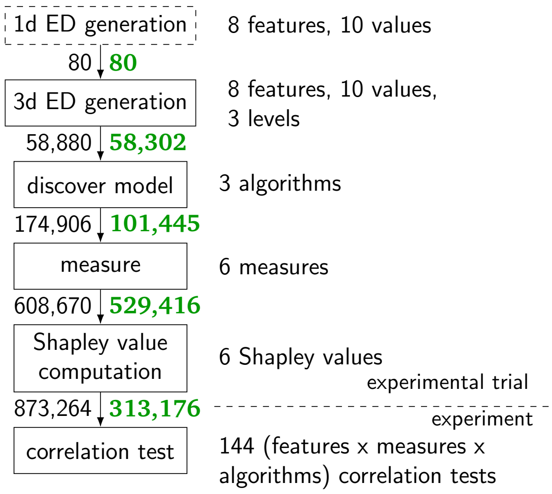

Figure 3.7: Visual representation of our experimental trial set-up. Annotations in black indicate how many combinations result, and in green indicate how many of these were feasible.

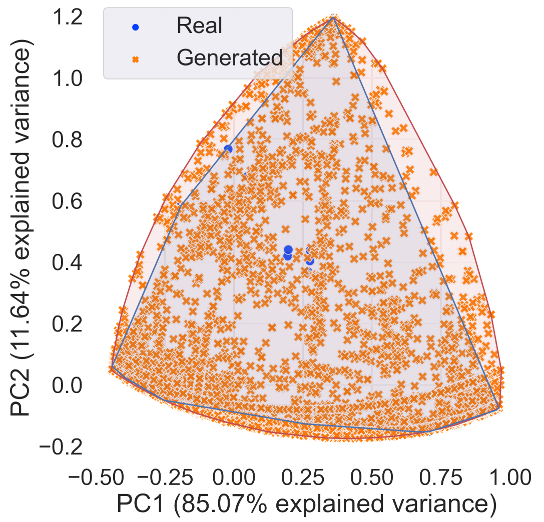

Figure 3.8: Our 22,000 generated and tested event logs cover and enrich the feature space.

Involved process discovery evaluation measurements (5) operationalize quality, simplicity, and performance of algorithms through standard metrics like fitness, precision, F-score, size of the resulting model, and execution time. Through the boxplots in Figure 3.9, we observe that for most combinations of algorithms and metrics, results using the generated event logs (in orange) cover and go beyond the previously available results from real event logs (in blue). This demonstrates the representational bias of current event data benchmarks, which was the gap addressed in the previous subsection. Particularly for the combination of output model size and Split Miner (sm1), feasibility restrictions to discover more models from more complex event logs with limited hardware constraints become evident.

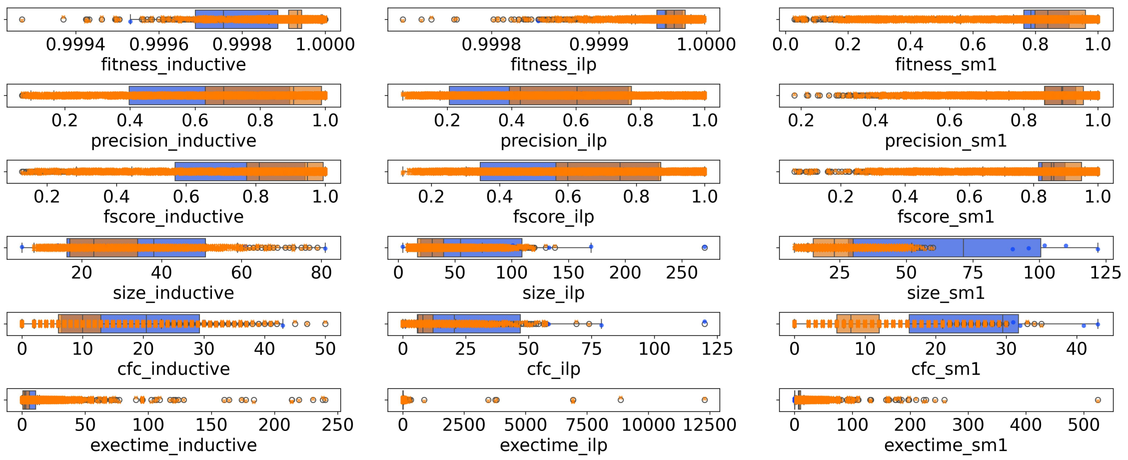

Figure 3.9: *Evaluation metrics of process discovery algorithm results from our 22,000 generated (orange) vs. real event logs (blue).*

Conclusions of this contribution provide answers to validate or disprove specific hypotheses about particular features, algorithms, and evaluation metric combinations. More generally, we validate Hypothesis 3, i.e., using *SHAining* [91], as we can measure how event log characteristics impact algorithm metrics for specific algorithms (6), contributing new empirical knowledge about algorithm robustness, generalizability, and the correlation of structural feature values and impact on evaluation results for specific metric-algorithm setups. This paper acts as the proof of concept and first systematic experiment of this methodology, using the previous insights and frameworks to produce final, conclusive knowledge.

Taken together, these papers provide a full-cycle methodology for algorithm engineering in process mining. They move the field towards more systematic, transparent, and reproducible research by providing experiments and frameworks for generating and analyzing data-driven behavior of algorithms. Consequently, we can confirm the *general hypothesis* from these results, answering our original *research question* in Chapter 1. This collective body of contributions lays the foundation for a new generation of empirical studies that can lead to more robust and reliable process mining algorithms.

Chapter 4

Conclusion

> Understand well as I may, my comprehension can only be an infinitesimal fraction of all I want to understand.
>
> *– Ada Lovelace*

In keeping with Ada Lovelace's profound insight, this thesis regards the vast, unknown space of knowledge not as a sobering predicament but as an encouraging invitation to broaden our understanding and embrace complexity. This chapter provides a culminating summary of our research findings and an overall synthesis of the presented methodology. We will address the justification, experimental value, and contributions to the field. Afterwards, we will reflect on its limitations, particularly validity concerns, as our own "process mining crimes". Finally, we will point out new research opportunities that arise from our work.

4.1 Summary

This thesis has successfully addressed the feasibility challenge of understanding the complex inner workings of process discovery algorithms by developing a novel explainability methodology based on Shapley values.

The justification of our methodology has two parts: On the one hand, identifying [94] and preserving [18] structure-aware data characteristics benefits ordered data analysis. On the other hand, as heavily pointed out by the process mining community, due to real event data scarcity, current process mining research lies in the persistent lack of systematic and generalizable evaluation datasets. Our work responds to this methodological gap by providing a data-driven approach to explaining algorithms' effectiveness and robustness in process discovery. The experimentation followed a full-cycle methodology grounded in algorithm engineering principles. Through systematic manipulation of event log characteristics—including noise, incompleteness, and structural complexity—, we generated more than 22,000 synthetic logs with intentional features, using a novel approach [92]. The experimental design in [91] allowed us to test hypotheses about the influence of log properties on algorithm outcomes, while ensuring reliability and reproducibility across a representative selection of algorithms and across different algorithmic paradigms, including top-down (Inductive Miner [77]), and bottom-up (ILP Miner [136], and Split Miner [14]), to showcase its general applicability. The resulting framework integrates these contributions into a modular and extensible pipeline. It combines structure-aware data characterization, intentional data generation, and Shapley-based impact measurement into a coherent methodology for empirical evaluation. This framework not only enabled us to demonstrate the robustness and trade-offs across algorithmic paradigms in [91], offline and online [93], but also provides a reusable foundation for future research. To this end, we introduce the Python packages *FEEED*[4] and *GEDI*[5]; the online application *iGEDI*[6], as well as our repository *SHAining*[7].

By uniting justification, experimentation, and framework, the thesis establishes a transparent methodology for understanding algorithm behavior in process mining. This thesis provides an affirmative answer to the guiding *Research Question*, presented in Chapter 1, demonstrating that by systematically connecting event data characteristics with algorithmic evaluation outcomes, a data-driven approach can indeed explain, in a reliable and generalizable way, the impact of data properties on process discovery evaluation. This work contributes to building trust in automated decision-making systems by making their evaluation more interpretable, reliable, and generalizable of real-world complexity. Ultimately, the ability to explain algorithm limitations and robustness in regards to data characteristics is crucial for fostering greater trust in process mining algorithm design.

4.2 Threats to validity

Nevertheless, how much can we trust our methodology? To assess the validity of our contribution in this dissertation, we employ the checklist by Rehse et al. [108]. While our experimentation and framework demonstrate strong safeguards against common pitfalls, its validity remains bounded by computational limitations, synthetic evaluation settings, and limited real-world validation. A first concern is **internal validity**. Although our experiments carefully controlled for noise, volume, and structural complexity, residual confounding effects from preprocessing choices cannot be fully excluded. A second concern is **construct validity**, as process discovery measures have been criticised for not measuring the intended matter [12, 65, 113]. This threat is reduced by including a comprehensive collection of widely accepted evaluation metrics for process discovery. As a third concern,

[4] `https://pypi.org/project/feeed/`
[5] `https://pypi.org/project/gedi/`
[6] `https://huggingface.co/spaces/andreamalhera/igedi`
[7] `https://github.com/andreamalhera/SHAining`

we confront **external validity**. With *GEDI* [92] and *iGEDI* [90], we systematically generated over 58,000 diverse synthetic logs to mitigate representational bias and validated our framework across three distinct algorithmic paradigms. Yet, the generalization of these findings to characteristics beyond structural ones may contain irregularities that were not fully captured in our experiments. To mitigate these threats, the modular nature of our framework easily allows for the integration of additional measures, algorithms, and characteristics in the future. Considering **conclusion validity**, a fourth concern involves large-scale problems. Due to computational constraints of discovering models for an exponentially increasing number of possible coalitions, approximations of Shapley values may introduce statistical uncertainty. Finally, on **ecological validity**, we acknowledge that our experiments were conducted in controlled synthetic settings, and whether they capture the complexity of operational environments and real-world contexts remains to be validated. Taken together, these limitations do not undermine the presented contributions but highlight important boundaries for interpreting results and avenues for methodological refinement.

4.3 Future Work

The proposed methodology offers several opportunities for further research projects. We want to present some ideas briefly:

Exploring Computational Optimizations: As noted in our limitations, computational constraints are a key area for improvement. We plan to address this by exploring refined sampling methods, using a priori reasoning, and leveraging domain-specific assumptions to reduce the number of logs required for each Shapley value computation while maintaining the reliability of our insights.

Expanding Algorithmic and Data Diversity: Currently, our framework has been applied to three representative process discovery algorithms, and the underlying pipeline relies on a structure-driven event data-generating system. Besides testing on a broader collection of algorithms, a valuable next step would be to generate intentional event data beyond event logs and data streams, including OCEL [22, 132], federated, ambiguous, and collaborative data. Another direction is generating intentional event data that embeds contextual constraints or accounts for additional event attributes, concerning, e.g., time constraints, payloads and resources.

Measuring Representativeness of Event Logs: A key challenge is defining and measuring the representativeness of a given event log. It would be valuable to explore new metrics for this purpose, for example, by using Hill diversity, and generalizing existing work [70, 71].

Integration into Tools and Other Process Mining Tasks: As a scientific community service, we suggest the integration of our explainability framework into existing process mining tools and challenges [31]. This would provide a good basis for comparing the interpretability of different procedures and for creating insightful benchmarks. To this end, we have introduced the *AVOCADO challenge* [64], a standardized evaluation framework integrating criteria for streaming algorithms. We have presented this challenge as a poster in at *ICPM 2025 Stream Management & Analytics for Process Mining Workshop*[8] in Montevideo, Uruguay.

Studying Fairness-Related Metrics: Finally, the framework establishes a unique foundation for studying fairness metrics by integrating fairness with utility and resemblance into a single evaluation methodology. Currently, a critical gap is that these dimensions are analyzed in isolation [141]. Future doctoral work can leverage our comprehensive framework to establish standardized evaluation protocols for diverse datasets and systematically detect and mitigate bias, thereby advancing the trustworthiness and generalizability of AI systems.

[8] `https://sma4pm.github.io/2025/`

Appendix

A DROPP: Structure-Aware PCA for Ordered Data: A General Method and its Applications in Climate Research and Molecular Dynamics

Authors

Anna Beer, Olivér Palotás, Andrea Maldonado, Andrew Draganov, and Ira Assent

Venue

Proceedings of the IEEE International Conference on Data Engineering, Utrecht, Netherlands, pages 1143-1156, ICDE 2024.
CORE ranking: A*. Average acceptance rate: 15.8%

DOI

https://doi.org/10.1109/ICDE60146.2024.00093

Code

`https://github.com/poliver269/DROPP`

Declaration of Authorship

This work was a joint effort with colleagues from Aarhus University, the University of Vienna, and LMU Munich. The author of this thesis made significant contributions by generalizing the approach from a molecular dynamics application to a broader method for ordered data. She also implemented and designed the evaluation of the general method on climate research data. All co-authors discussed the results periodically and finalized the manuscript in collaboration. Andrea Maldonado made significant contributions by generalizing the approach from a molecular dynamics application to a broader method for ordered data; she also implemented and designed the evaluation of the general method on climate research data.

Thesis Reference

[18], [BPM+24]

B FEEED: Feature Extraction from Event Data

Authors

Andrea Maldonado, Gabriel Marques Tavares, Rafael Seidi Oyamada, Paolo Ceravolo, and Thomas Seidl

Venue

Proceedings of the ICPM 2023 Tool Demonstration Track, co-located with the 5th International Conference on Process Mining (ICPM 2023), Rome, Italy, October 27, 2023. CEUR Workshop Proceedings, Volume 3648, pages 6598.

URL

https://ceur-ws.org/Vol-3648/paper_6598.pdf

Code

https://github.com/lmu-dbs/feeed/tree/demo-icpm23

Declaration of Authorship

This work was a joint collaboration with former colleagues from LMU Munich and Università degli Studi di Milano. The author of this thesis proposed, developed, and conceptualized the idea. She also implemented the library, designed the experiments and wrote the manuscript. All co-authors revised the manuscript in collaboration.

Thesis Reference

[94], [MTO+23]

C Suppl. Materials to "FEEED: Feature Extraction from Event Data"

Clustered correlation matrix generated using full hierarchical clustering with single linkage and Pearson correlation distance [83]. Features are clustered across all groups to reveal global similarity patterns. Dots indicate feature types as defined in [94], with uppercase and prefixed features marking representative features.

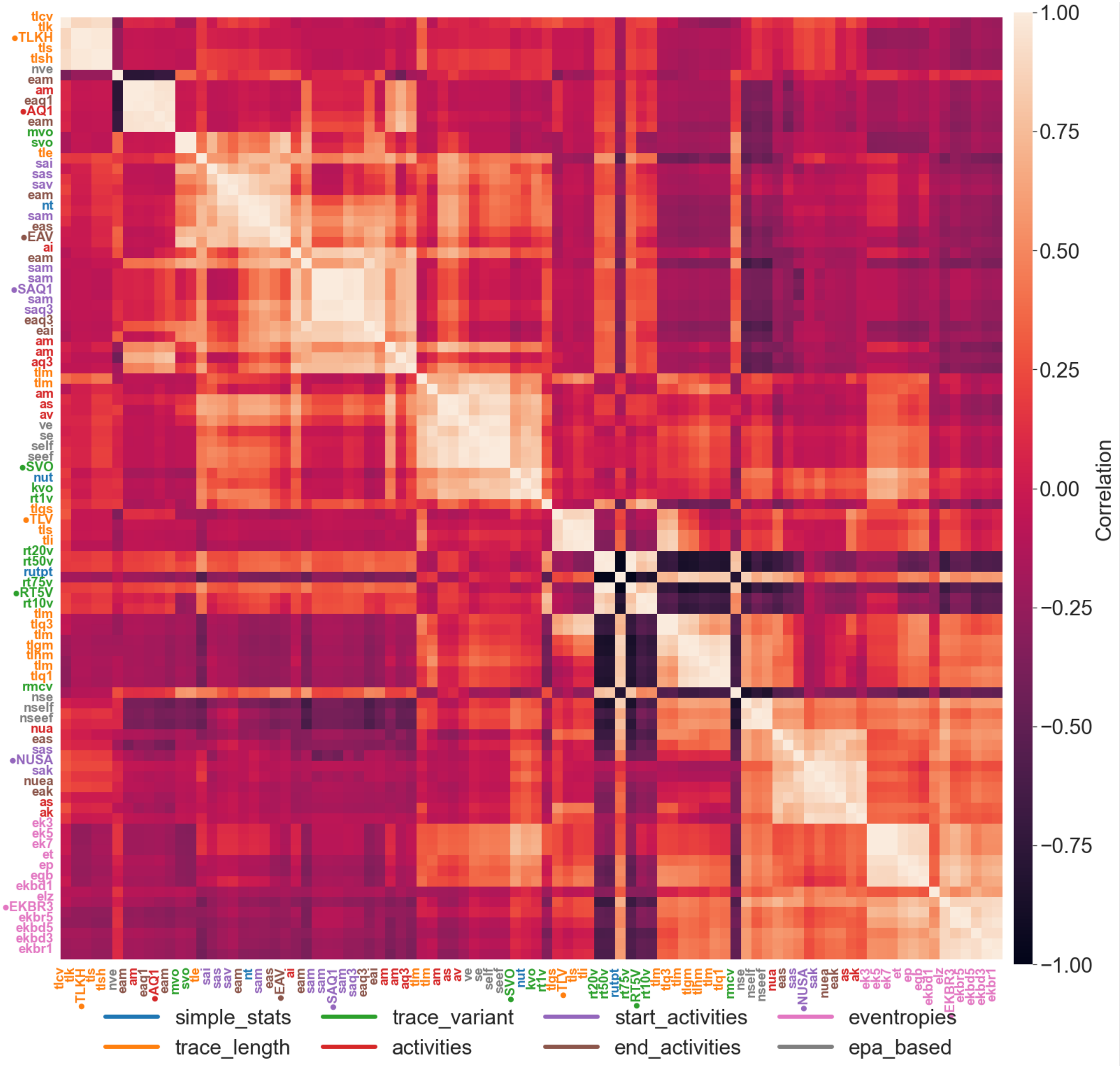

Figure 4.1: Extended clustered correlation matrix generated using full hierarchical clustering with single linkage and Pearson correlation distance [83].

The full list of all feature names [9], as well as the code to produce this plot, can be found in our repository [10].

[9] `https://github.com/lmu-dbs/feeed`

[10] `https://github.com/andreamalhera/SHAining/blob/main/notebooks/section5_greedy_feature_selection.ipynb`

D GEDI: Generating Event Data with Intentional Features for Benchmarking Process Mining

Authors

Andrea Maldonado, Christian M. M. Frey, Gabriel M. Tavares, Nikolina Rehwald, and Thomas Seidl

Venue

Proceedings of Business Process Management, Krakow, Poland, pages 221–237, BPM 2024 [92].
CORE ranking: A, a top conference in process mining.
Average acceptance rate: 20%.

DOI

https://doi.org/10.1007/978-3-031-70396-6_13

Code

https://github.com/lmu-dbs/gedi/tree/bpm24/notebooks
This contribution earned the “Artifacts Available” badge during the conference.

Declaration of Authorship

This work was a joint effort with former colleagues from LMU Munich and Fraunhofer IIS. The author of this thesis proposed, developed, and conceptualized the research idea, which she also implemented, designed, and evaluated. All co-authors discussed the results periodically. The author of this theses co-wrote the manuscript in close collaboration with Christian M. M. Frey.

Thesis Reference

[92], [MFT+24]

E iGEDI: interactive Generating Event Data with Intentional Features

Authors

Andrea Maldonado, Sai Anirudh Aryasomayajula, Christian M. M. Frey, and Thomas Seidl

Venue

Proceedings of the ICPM 2024 Tool Demonstration Track, co-located with the 6th International Conference on Process Mining (ICPM 2024), Copenhagen, Denmark, October 14-18, 2024. CEUR Workshop Proceedings, Volume 3783, pages 349.

URL

https://ceur-ws.org/Vol-3783/paper_349.pdf

Code

https://github.com/lmu-dbs/gedi/tree/demo-icpm24

Declaration of Authorship

This work was a joint effort with former colleagues from LMU Munich and the University of Technology Nuremberg. The author of this thesis proposed and developed the idea. She performed the implementation, which was later refined by S. Anirudh Aryasomayajula. She also wrote the manuscript, with all authors assisting in its finalization.

Thesis Reference

[90], [MAF+24]

F Know Your Streams: On the Conceptualization, Characterization, and Generation of Intentional Event Streams

Authors

Andrea Maldonado*, Christian Imenkamp*, Hendrik Reiter, Thomas Seidl, Wilhelm Hasselbring, Martin Werner, and Agnes Koschmider
* *First authors with equal contribution*

Venue

Proceedings of the International Conference on Process Mining Workshops, ICPM'25 Workshops [93]. Venue details: Workshop co-located with ICPM 2025.

DOI

Publicaiton pending

Code

https://github.com/andreamalhera/gedi_streams

Declaration of Authorship

This work was a joint effort with colleagues from LMU Munich, University of Bayreuth, Christian-Albrechts-University Kiel, and others. The author of this thesis proposed and developed the core idea. She implemented the first version of the approach. In close collaboration with Christian Imenkamp, she designed the evaluation and experiments and co-wrote the manuscript.

Thesis Reference

[93], [MIR+25]

G SHAining on Process Mining: Explaining Event Log Characteristics Impact on Algorithms

Authors

Andrea Maldonado, Christian M. M. Frey, Sai Anirudh Aryasomayajula, Stephan A. Fahrenkrog-Petersen, Ludwig Zellner, and Thomas Seidl [91]

Venue

Proceedings of International Conference on Process Mining, Montevideo, Uruguay, ICPM'25 [91].
CORE ranking: B, a top conference in process mining
Average acceptance rate: 24%

DOI

Publication pending. https://doi.org/10.48550/arXiv.2509.08482

Code

https://github.com/andreamalhera/SHAining

Declaration of Authorship

This work was a joint effort with former colleagues at LMU Munich, colleagues from the Machine Learning Lab at University of Technology Nuremberg, and University of Liechtenstein. The author of this thesis proposed, developed and implemented the theoretical idea. She performed the evaluation in close collaboration with Christian M. M. Frey and wrote the manuscript. All authors assisted in its finalization.

Thesis Reference

[91], [MFA+25]

References

[1] Gdpr. https://eur-lex.europa.eu/eli/reg/2016/679/oj. accessed: 24.09.2025.

[2] Health insurance portability and accountability act of 1996. ((https://www.govinfo.gov/content/pkg/PLAW-104publ191/pdf/PLAW-104publ191.pdf)), 1996. public Law 104-191.

[3] Ieee standard for extensible event stream (xes) for achieving interoperability in event logs and event streams. *IEEE Std 1849-2016*, pages 1–50, 2016.

[4] J. N. Adams, C. Pitsch, T. Brockhoff, and W. M. Van Der Aalst. An experimental evaluation of process concept drift detection. *Proceedings of the VLDB Endowment*, 16(8):1856–1869, 2023.

[5] R. Agrawal and R. Srikant. Mining sequential patterns. *Proceedings of the 11th IEEE International Conference on Data Engineering (ICDE 1995)*, pages 3–14, 1995.

[6] L. Alam and S. Mueller. Examining the effect of explanation on satisfaction and trust in ai diagnostic systems. *BMC medical informatics and decision making*, 21(1):178, 2021.

[7] A. Alves de Medeiros, A. Weijters, and W. van der Aalst. Using genetic algorithms to mine process models. In *Applications and Theory of Petri Nets 2007*, pages 48–69. Springer, 2007.

[8] K. Andree, M. Hoang, F. Dannenberg, I. Weber, and L. Pufahl. Discovery of workflow patterns-a comparison of process discovery algorithms. In *International Conference on Cooperative Information Systems*. Springer, 2023.

[9] R. Andrews, F. Emamjome, A. H. ter Hofstede, and H. A. Reijers. An expert lens on data quality in process mining. In *2020 2nd International Conference on Process Mining (ICPM)*, pages 49–56, 2020.

[10] E. Angriman, A. van der Grinten, M. von Looz, H. Meyerhenke, M. Nöllenburg, M. Predari, and C. Tzovas. Guidelines for experimental algorithmics: A case study in network analysis. *Algorithms*, 12(7):127, 2019.

[11] S. Appleby, G. Bergami, and G. Morgan. Enhancing declarative temporal model mining in relational databases: A preliminary study. In *Proceedings of the 27th International Database Engineered Applications Symposium*, pages 34–42, New York, NY, USA, 2023. Association for Computing Machinery.

[12] A. Augusto, R. Conforti, A. Armas-Cervantes, M. Dumas, and M. L. Rosa. Measuring fitness and precision of automatically discovered process models: A principled and scalable approach. *IEEE Transactions on Knowledge and Data Engineering*, 34(4):1870–1888, 2022.

[13] A. Augusto, R. Conforti, M. Dumas, M. La Rosa, F. M. Maggi, A. Marrella, M. Mecella, and A. Soo. Automated discovery of process models from event logs: Review and benchmark. *IEEE transactions on knowledge and data engineering*, 2018.

[14] A. Augusto, R. Conforti, M. Dumas, M. La Rosa, and A. Polyvyanyy. Split miner: automated discovery of accurate and simple business process models from event logs. *Knowledge and Information Systems*, 59(2):251–284, May 2019.

[15] A. Augusto, J. Mendling, M. Vidgof, and B. Wurm. The connection between process complexity of event sequences and models discovered by process mining. *Information Sciences*, 598:196–215, 2022.

[16] C. O. Back, S. Debois, and T. Slaats. Entropy as a measure of log variability. *Journal on Data Semantics*, 8(2):129–156, 2019.

[17] S. Barbon Junior, P. Ceravolo, E. Damiani, and G. Marques Tavares. Evaluating trace encoding methods in process mining. In *International Symposium: From Data to Models and Back*, pages 174–189. Springer, 2020.

[18] A. Beer, O. Palotás, A. Maldonado, A. Draganov, and I. Assent. DROPP: Structure-Aware PCA for Ordered Data: A General Method and its Applications in Climate Research and Molecular Dynamics. In *IEEE International Conference on Data Engineering (ICDE 2024)*, pages 1143–1156, 2024.

[19] O. A. Bello, A. Folorunso, O. E. Ejiofor, F. Z. Budale, K. Adebayo, and O. A. Babatunde. Machine learning approaches for enhancing fraud prevention in financial transactions. *International Journal of Management Technology*, 10(1):85–108, 2023.

[20] D. Benvenuti, L. Falleroni, A. Marrella, and F. Perales. An interactive approach to support event log generation for data pipeline discovery. In *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1172–1177, Jun 2022.

[21] G. Bergami. Fast synthetic data-aware log generation for temporal declarative models. In *Proceedings of the 6th Joint Workshop on Graph Data Management Experiences Systems (GRADES) and Network Data Analytics (NDA)*, pages 1–9, Seattle WA USA, Jun 2023. ACM.

[22] Y. Bertrand, C. Imenkamp, L. Malburg, M. Ehrendorfer, M. Franceschetti, J. Grüger, F. Leotta, J. Mangler, R. Seiger, A. Koschmider, S. Rinderle-Ma, B. Weber, and E. Serral. An object-centric core metamodel for iot-enhanced event logs, 2025.

[23] E. Bogdanov, I. Cohen, and A. Gal. Conformance checking over stochastically known logs. In *International Conference on Business Process Management*, pages 105–119. Springer, 2022.

[24] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos. A review of feature selection methods on synthetic data. *Knowledge and information systems*, 34(3):483–519, 2013.

[25] A. Burattin. Plg2: Multiperspective process randomization with online and offline simulations. *Proceedings of the BPM Demo Track 2016, co-located with the 14th International Conference on Business Process Management (BPM 2016)*, 1789:1–6, 2016.

[26] A. Burattin. Streaming process mining. *Process Mining Handbook*, 349:3–10, 2022.

[27] A. Burattin, M. Cimitile, F. M. Maggi, and A. Sperduti. Online discovery of declarative process models from event streams. *IEEE Transactions on services computing*, 8(6):833–846, 2015.

[28] A. Burattin, B. Re, L. Rossi, and F. Tiezzi. A purpose-guided log generation framework. In *International Conference on Business Process Management (BPM 2022)*, pages 181–198. Springer, 2022.

[29] A. Burattin and A. Sperduti. Plg: a framework for the generation of business process models and their execution logs. In *BPM 2010 workshops*, volume 66, pages 214–219, 2010.

[30] J. Bye, K. Fraedrich, E. Kirk, S. Schubert, and X. Zhu. Random walk lengths of about 30 years in global climate. *Geophysical Research Letters*, 38(5), 2011.

[31] J. Carmona, M. de Leoni, B. Depaire, and T. Jouck. Summary of the process discovery contest 2016. In *Lecture Notes in Business Information Processing*. Springer, 2017.

[32] M. Chaudhari and C. Mehta. A survey on algorithms for sequential pattern mining. *International Journal of Engineering Development and Research*, 3(4):1312–1319, 2015.

[33] C. Ciccio, M. Bernardi, M. Cimitile, and F. Maggi. Generating event logs through the simulation of declare models. In *Workshop on Enterprise and Organizational Modeling and Simulation*, pages 20–36. Springer, 2015.

[34] C. D. Ciccio and M. Montali. Declarative process specifications: Reasoning, discovery, monitoring. In W. van der Aalst et al., editors, *Handbook of Process Mining*, pages 45–72. Springer, 2023.

[35] T. Colburn. Methodology of computer science. *The Blackwell guide to the philosophy of computing and information*, pages 318–326, 2004.

[36] T. D. Cook, D. T. Campbell, and W. Shadish. *Experimental and quasi-experimental designs for generalized causal inference*, volume 1195. Houghton Mifflin Boston, MA, 2002.

[37] A. Costa, S. Y. Eroglu, K. Andree, and L. Pufahl. A collection of publicly available event logs enhanced by metadata for process mining research. In H. Reijers, A. Marrella, A. del Río Ortega, S. Rinderle-Ma, B. Depaire, J.-R. Rehse, F. Santoro, F. Zerbato, A. E. Marquez-Chamorro, I. Beerepoot, S. Agostinelli, and J. D. Smedt, editors, *Proceedings of the Best Dissertation Award, Doctoral Consortium, and Demonstration & Resources Forum at the 23rd International Conference on Business Process Management (BPM 2025)*, CEUR Workshop Proceedings, Seville, Spain, Aug. 2025.

[38] J. Cuomo, H. Homayouni, I. Ray, and S. Ghosh. Detecting temporal dependencies in data. *Proceedings of the British International Conference on Databases*, 2022.

[39] A. A. B. da Costa and P. Dasgupta. Learning temporal causal sequence relationships from real-time time-series. *Journal of Artificial Intelligence Research*, 70:205–243, 2021.

[40] P. De Koninck, S. Vanden Broucke, and J. De Weerdt. act2vec, trace2vec, log2vec, and model2vec: representation learning for business processes. In *16th International Conference on Business Process Management (BPM 2018)*, pages 305–321. Springer, 2018.

[41] M. de Leoni, W. M. van der Aalst, and C. Di Francescomarino. A general process mining framework for correlating, predicting and analyzing process characteristics. *Information Systems*, 54:235–257, 2015.

[42] J. De Weerdt and M. T. Wynn. *Foundations of Process Event Data*, pages 193–211. Springer International Publishing, Cham, 2022.

[43] F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint:1702.08608*, 2017.

[44] G. H. Dunteman. *Principal Components Analysis*, volume 69 of *Quantitative Applications in the Social Sciences*. SAGE Publications, Inc., Newbury Park, CA, 1989.

[45] V. Durand-Guerrier. Truth versus validity in mathematical proof. *ZDM*, 40(3):373–384, 2008.

[46] G. Engelberg, M. Hadad, M. Pegoraro, P. Soffer, E. Hadar, and W. M. van der Aalst. An uncertainty-aware event log of network traffic. In *BPM (Demos/Resources Forum)*, pages 67–71, 2023.

[47] M. Fani Sani, S. J. van Zelst, and W. M. van der Aalst. The impact of biased sampling of event logs on the performance of process discovery. *Computing*, 103(6):1085–1104, 2021.

[48] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37–37, 1996.

[49] J. Ferina and V. Daggett. Visualizing protein folding and unfolding. *Journal of Molecular Biology*, 431(8):1540–1564, 2019.

[50] I. Fontana, M. Langheinrich, and M. Gjoreski. Gans for privacy aware mobility modeling. *IEEE Access*, 11:29250–29262, 2023.

[51] B. Gebru, L. Zeleke, D. Blankson, M. Nabil, S. Nateghi, A. Homaifar, and E. Tunstel. A review on human–machine trust evaluation: Human-centric and machine-centric perspectives. *IEEE Transactions on Human-Machine Systems*, 52(5):952–962, 2022.

[52] K. Goel, S. J. Leemans, N. Martin, and M. T. Wynn. Quality-informed process mining: A case for standardised data quality annotations. *ACM Transactions on Knowledge Discovery from Data*, 16(3):1–47, 2022.

[53] H. Goosse. *Climate System Dynamics and Modelling*. Cambridge University Press, Cambridge, 2024.

[54] A. H. Gordon. Global warming as a manifestation of a random walk. *Journal of Climate*, 4(6):589–597, 1991.

[55] Grand View Research. Intelligent process automation market size, share trends analysis report by component, by technology, by deployment, by organization size, by application, by end-use, by region, and segment forecasts, 2025 - 2030, 2023. Accessed Sep 27th, 2025, Report: GVR-4-68039-270-1.

[56] J. Grüger, T. Geyer, D. Jilg, and R. Bergmann. Sample: A semantic approach for multi-perspective event log generation. In M. Montali, A. Senderovich, and M. Weidlich, editors, *Process Mining Workshops*, pages 328–340, Cham, 2023. Springer Nature Switzerland.

[57] J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, Boston, 3rd edition, 2011.

[58] D. J. Haraway. *Modest_Witness@Second_Millennium. FemaleMan_Meets_OncoMouse™: Feminism and Technoscience*. Routledge, New York, 1997.

[59] F. U. Hartl. Protein misfolding diseases. *Annual review of biochemistry*, 86:21–26, 2017.

[60] J. J. Heckman. Sample selection bias as a specification error. *Econometrica: Journal of the econometric society*, pages 153–161, 1979.

[61] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[62] P. Imbach, S. C. Chou, A. Lyra, D. Rodrigues, D. Rodriguez, D. Latinovic, G. Siqueira, A. Silva, L. Garofolo, and S. Georgiou. Future climate change scenarios in central america at high spatial resolution. *PLoS One*, 13(4):e0193570, 2018.

[63] C. Imenkamp, M. Kabierski, H. Reiter, M. Weidlich, W. Hasselbring, and A. Koschmider. Determining window sizes using species estimation for accurate process mining over streams. In *International Conference on Advanced Information Systems Engineering*, pages 109–124. Springer, 2025.

[64] C. Imenkamp, A. Maldonado, H. Reiter, M. Werner, W. Hasselbring, A. Koschmider, and A. Burattin. Avocado: The streaming process mining challenge. *arXiv preprint:2510.17089*, 2025. Accepted at conference: Modellierung 2026.

[65] G. Janssenswillen, N. Donders, T. Jouck, and B. Depaire. A comparative study of existing quality measures for process discovery. *Information Systems*, 71:1–15, 2017.

[66] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New York, 2002.

[67] D. T. Jones. t42morph: A protein structure prediction resource. `http://www0.cs.ucl.ac.uk/staff/d.jones/t42morph.html`, 2025. Accessed: 2025-10-01.

[68] T. Jouck, A. Bolt, B. Depaire, M. de Leoni, and W. M. P. van der Aalst. An integrated framework for process discovery algorithm evaluation. *IEEE Transactions on Knowledge and Data Engineering*, 33(6):2370–2385, 2021.

[69] T. Jouck and B. Depaire. Generating artificial data for empirical analysis of control-flow discovery algorithms. *Business & Information Systems Engineering*, 61(6):695–712, Dec 2019.

[70] M. Kabierski, M. Richter, and M. Weidlich. Addressing the log representativeness problem using species discovery. In *2023 5th International Conference on Process Mining (ICPM)*, pages 65–72. IEEE, 2023.

[71] M. Kabierski, M. Richter, and M. Weidlich. Quantifying and relating the completeness and diversity of process representations using species estimation. *Information Systems*, 130:102512, 2025.

[72] A. A. Kalenkova, I. A. Lomazova, and W. M. P. van der Aalst. Process model discovery: A method based on transition system decomposition. In *Application and Theory of Petri Nets and Concurrency*, volume 8489 of *Lecture Notes in Computer Science*, pages 96–115. Springer, 2014.

[73] A. A. Kalenkova, W. M. van der Aalst, I. A. Lomazova, and V. A. Rubin. Process mining using bpmn: relating event logs and process models. In *Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems*, pages 123–123, 2016.

[74] Y. S. Kim, M. K. Kim, N. Fu, J. Liu, J. Wang, and J. Srebric. Investigating the impact of data normalization methods on predicting electricity consumption in a building using different artificial neural network models. *Sustainable Cities and Society*, 118:105570, 2025.

[75] A. Koschmider, D. Janssen, and F. Mannhardt. Framework for process discovery from sensor data. In *EMISA Workshop 2020 : Enterprise Modeling and Information Systems Architectures 2020: 10th International Workshop on Enterprise Modeling and Information Systems Architectures*, pages 32–38. CEUR-WS.org, 2020.

[76] J. K. Lannin. Generalization and justification: The challenge of introducing algebraic reasoning through patterning activities. *Mathematical Thinking and learning*, 7(3):231–258, 2005.

[77] S. J. Leemans, D. Fahland, and W. M. Van Der Aalst. Discovering block-structured process models from event logs containing infrequent behaviour. In *Business Process Management Workshops: BPM 2013 International Workshops, Beijing, China, August 26, 2013, Revised Papers 11*, pages 66–78. Springer, 2014.

[78] S. J. J. Leemans, K. Goel, and S. J. van Zelst. Using multi-level information in hierarchical process mining. In *International Conference on Process Mining (ICPM)*, 2020.

[79] S. J. J. Leemans, N. Tax, and A. H. M. ter Hofstede. Indulpet miner: Combining discovery algorithms. In *International Conference on Cooperative Information Systems*, pages 149–167. Springer, 2018.

[80] S. J. J. Leemans, W. M. P. van der Aalst, and U. Čubrić. Directly-follows-based process mining: Exploration & abstraction. In *Proceedings of the 11th International Conference on Process Mining (ICPM)*, pages 1–12, 2019.

[81] A. Lepsien. Quantifying uncertainty for explainable process mining (phd proposal). In *Proceedings of the ICPM Doctoral Consortium 2023*, volume 3502 of *CEUR Workshop Proceedings*, pages 21–24, 2023.

[82] Z. C. Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[83] X. Liu et al. A correlation-matrix-based hierarchical clustering method for functional brain connectivity analysis. *Advances in Cognitive Neurodynamics*, 2012:1–10, 2012.

[84] D. Loreti, F. Chesani, A. Ciampolini, and P. Mello. Generating synthetic positive and negative business process traces through abduction. *Knowledge and Information Systems*, 62:813–839, Feb 2020.

[85] G. Lucas et al. Calibrating workers' trust in intelligent automated systems. *Frontiers in Psychology*, 15:11573890, 2024.

[86] E. Lundstad et al. The global historical climate database hclim. *Scientific Data*, 10(1):1–19, 2023.

[87] A. Maaradji, M. Dumas, M. La Rosa, and A. Ostovar. Detecting sudden and gradual drifts in business processes from execution traces. *IEEE Transactions on Knowledge and Data Engineering*, 29(10):2140–2154, 2017.

[88] N. K. Maharana and G. Saritha. Data pre-processing and data augmentation techniques: A review. *Materials Today: Proceedings*, 70:173–181, 2022.

[89] G. G. Maisuradze, A. Liwo, and H. A. Scheraga. Principal component analysis for protein folding dynamics. *Journal of Molecular Biology*, 385(1):312–329, 2009.

[90] A. Maldonado, S. A. Aryasomayajula, C. M. M. Frey, and T. Seidl. iGEDI: interactive Generating Event Data with Intentional Features. In *International Conference on Process Mining (ICPM) Doctoral Consortium / Demo 2024*, 2024.

[91] A. Maldonado, C. M. M. Frey, S. A. Aryasomayajula, S. A. Fahrenkrog-Petersen, L. Zellner, and T. Seidl. SHAining on Process Mining: Explaining Event Log Characteristics Impact on Algorithms. In *Proceedings of International Conference on Process Mining (ICPM'25)*, Montevideo, Uruguay, 2025.

[92] A. Maldonado, C. M. M. Frey, G. M. Tavares, N. Rehwald, and T. Seidl. GEDI: Generating Event Data with Intentional Features for Benchmarking Process Mining. In *Proceedings of the 22nd International Conference on Business Process Management (BPM 2024)*, pages 221–237, 2024.

[93] A. Maldonado, C. Imenkamp, H. Reiter, T. Seidl, W. Hasselbring, M. Werner, and A. Koschmider. Know your streams: On the conceptualization, characterization, and generation of intentional event streams. In *International Conference on Process Mining (ICPM'25)*, 2025. Workshops.

[94] A. Maldonado, G. M. Tavares, R. S. Oyamada, P. Ceravolo, and T. Seidl. FEEED: feature extraction from event data. In *Tool Demonstration Track at the 5th International Conference on Process Mining (ICPM 2023)*, volume 3648 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2023.

[95] S. Marder. *Research Methods for Science*. Cambridge University Press, 2011.

[96] J. Mendling, H. Leopold, H. Meyerhenke, and B. Depaire. Methodology of algorithm engineering. *arXiv preprint arXiv:2310.18979*, 2023.

[97] M. Mirzaie, B. Behkamal, M. Allahbakhsh, S. Paydar, and E. Bertino. State of the art on quality control for data streams: A systematic literature review. *Computer Science Review*, 48:100554, May 2023.

[98] A. A. Mitsyuk, I. S. Shugurov, A. A. Kalenkova, and W. M. van der Aalst. Generating event logs for high-level process models. *Simulation Modelling Practice and Theory*, 74:1–16, 2017. Publisher: Elsevier.

[99] L. Mosquera, K. El Emam, L. Ding, V. Sharma, X. Zhang, S. Kababji, C. Carvalho, B. Hamilton, D. Palfrey, L. Kong, B. Jiang, and D. Eurich. A method for generating synthetic longitudinal health data. *BMC Medical Research Methodology*, 23:67, Mar 2023.

[100] Y. Naritomi and S. Fuchigami. Slow dynamics in protein fluctuations revealed by time-structure based independent component analysis: the case of domain motions. *The Journal of Chemical Physics*, 134(6):065101, 2011.

[101] G. Nikolentzos, M. Vazirgiannis, C. Xypolopoulos, M. Lingman, and E. Brandt. Synthetic electronic health records generated with variational graph autoencoders. *npj Digital Medicine*, 6:1–12, Apr 2023.

[102] T. R. S. A. of Sciences. The nobel prize in chemistry 2024 - computational protein design and structure prediction. `https://www.nobelprize.org/prizes/chemistry/2024/press-release/`, 2024. Awarded to David Baker for computational protein design and to Demis Hassabis and John Jumper for protein structure prediction.

[103] M. Pishgar, S. Harford, J. Theis, W. Galanter, J. M. Rodríguez-Fernández, L. Chaisson, Y. Zhang, A. Trotter, K. M. Kochendorfer, et al. A process mining-deep learning approach to predict survival in a cohort of hospitalized covid-19 patients. *BMC Medical Informatics and Decision Making*, page 194, 2022.

[104] S. K. Pradhan, M. Jans, and N. Martin. Getting the data in shape for your process mining analysis: An in-depth analysis of the pre-analysis stage. *ACM Computing Surveys*, 2025.

[105] F. Qayyum, S. Anwar, M. Sajid, and et al. A survey of datasets, preprocessing, modeling and usability in big data analytics. *IEEE Access*, 10:62655–62697, 2022.

[106] M. Rafiei, F. Wangelik, M. Pourbafrani, and W. van der Aalst. Travag: Differentially private trace variant generation using gans. In S. Nurcan, A. Opdahl, H. Mouratidis, and A. Tsohou, editors, *Research Challenges in Information Science: Information Science and the Connected World*, Lecture Notes in Business Information Processing, pages 415–431, Cham, 2023. Springer Nature Switzerland.

[107] J.-R. Rehse and P. Fettke. Process mining crimes – a threat to the validity of process discovery evaluations. In *BPM Forum in International Conference on Business Process Management*, pages 3–19. Springer, 2018.

[108] J.-R. Rehse, S. J. Leemans, P. Fettke, and J. M. E. van der Werf. On process discovery experimentation: addressing the need for research methodology in process discovery. *ACM Transactions on Software Engineering and Methodology*, 34(1):1–29, 2024.

[109] J.-R. Rehse, S. J. J. Leemans, and J. M. E. van der Werf. Process miner, are you sure? conducting valid and reliable experiments in process mining. *Tutorial in International Conference on Business Process Management*, 2024.

[110] H. A. Reijers, T. Slaats, and C. Stahl. Declarative modeling–an academic dream or the future for bpm? In F. Daniel, J. Wang, and B. Weber, editors, *Business Process Management*, pages 307–322, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[111] H. A. Reijers, W. M. van der Aalst, et al. Unlocking non-block-structured decisions: Inductive mining for complex processes. In *International Conference on Business Process Management*, pages 1–16, 2007.

[112] A. Rozinat, A. K. A. de Medeiros, C. W. Günther, A. Weijters, and W. M. van der Aalst. The need for a process mining evaluation framework in research and practice: position paper. In *International Conference on Business Process Management*, pages 84–89. Springer, 2007.

[113] A. Rozinat, A. K. A. de Medeiros, C. W. Günther, A. Weijters, and W. M. van der Aalst. The need for a process mining evaluation framework in research and practice: position paper. In *International Conference on Business Process Management*, pages 84–89. Springer, 2007.

[114] A. Rullo, F. Alam, and E. Serra. Trace encoding techniques for multi-perspective process mining: A comparative study. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 15(1):e1573, 2025.

[115] T. J. Santner, B. J. Williams, W. I. Notz, and B. J. Williams. *The design and analysis of computer experiments*, volume 1. Springer, 2003.

[116] M. Savino, C. Di Nunzio, A. Piemonte, F. Greco, G. Maccioni, and A. Shaban-Nejad. A process mining approach for clinical guidelines compliance. *Journal of Biomedical Informatics*, 137:104220, 2023.

[117] M. Schilling et al. A literature review on process mining. Technical report, FH Wedel, 2022.

[118] S. Schultze and H. Grubmüller. Time-lagged independent component analysis of random walks and protein dynamics. *Journal of Chemical Theory and Computation*, 17(9):5766–5776, 2021.

[119] W. R. Shadish, T. D. Cook, and D. T. Campbell. *Experimental and quasi-experimental designs for generalized causal inference.* Houghton, Mifflin and Company, 2002.

[120] S. E. Sim, S. Easterbrook, and R. C. Holt. Using benchmarking to advance research: A challenge to software engineering. In *25th International Conference on Software Engineering, 2003. Proceedings.*, pages 74–83. IEEE, 2003.

[121] M. Song, G. Janssenswillen, W. M. van der Aalst, and R. Conforti. Supporting process mining by showing events at a glance. *IEEE Transactions on Knowledge and Data Engineering*, 31(2):267–280, 2019.

[122] Y. Song, J. Lu, H. Lu, and G. Zhang. Learning data streams with changing distributions and temporal dependency. *IEEE Transactions on Neural Networks and Learning Systems*, 34(8):3952–3965, 2023.

[123] D. Suriadi, J. Munoz-Gama, et al. Explainable ai in process mining: A systematic literature review. *ACM Computing Surveys*, 58(3):1–36, 2025.

[124] P. G. Swanborn. A common base for quality control criteria in quantitative and qualitative research. *Quality and quantity*, 30(1):19–35, 1996.

[125] S. J. Teague. Implications of protein flexibility for drug discovery. *Nature reviews Drug discovery*, 2(7):527–541, 2003.

[126] R. L. Thorndike. Who belongs in the family? *Psychometrika*, 18(4):267–276, 1953.

[127] W. M. van der Aalst. On the representational bias in process mining. In *2011 IEEE 20th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pages 2–7. IEEE, 2011.

[128] W. M. van der Aalst. Foundations of process discovery. In *Process Mining Handbook*. Springer, 2022.

[129] W. M. Van Der Aalst and B. F. Van Dongen. Discovering petri nets from event logs. In *Transactions on Petri nets and other models of concurrency vii*, pages 372–422. Springer, 2013.

[130] W. M. P. van der Aalst. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer Berlin Heidelberg, 2011.

[131] W. M. P. van der Aalst. *Process Mining: A 360 Degree Overview*, pages 3–34. Springer International Publishing, Cham, 2022.

[132] W. M. P. van der Aalst. Object-centric process mining: An introduction. *Information Systems*, 110:101810, 2024.

[133] B. van Dongen. BPI Challenge 2020. 4TU.ResearchData, 2020. Dataset.

[134] S. J. van Zelst, F. Mannhardt, M. de Leoni, and A. Koschmider. Event abstraction in process mining: literature review and taxonomy. *Granular Computing*, 6(3):719–736, 2021.

[135] S. J. van Zelst, B. F. van Dongen, W. M. van der Aalst, and H. Verbeek. Discovering workflow nets using integer linear programming. *Computing*, 100:529–556, 2018.

[136] S. J. van Zelst, B. F. van Dongen, and W. M. P. van der Aalst. Event stream-based process discovery using abstract representations. *Knowledge and Information Systems*, 54(2):407–435, feb 2018.

[137] S. Vanden Broucke, J. Vanthienen, and B. Baesens. Straightforward petri net-based event log generation in prom. *SSRN Electronic Journal*, 2014.

[138] S. K. vanden Broucke, C. Delvaux, J. Freitas, T. Rogova, J. Vanthienen, and B. Baesens. Uncovering the relationship between event log characteristics and process discovery techniques. In *BPM 2013 International Workshops, Beijing, China, August 26, 2013, Revised Papers 11*, pages 41–53. Springer, 2014.

[139] S. K. vanden Broucke, C. Delvaux, J. Freitas, T. Rogova, J. Vanthienen, and B. Baesens. Uncovering the relationship between event log characteristics and process discovery techniques. In *BPM 2013 International Workshops, Beijing, China, August 26, 2013, Revised Papers 11*, pages 41–53. Springer, 2014.

[140] L. Wang, W. Zhang, and X. He. Continuous patient-centric sequence generation via sequentially coupled adversarial learning. In *Database Systems for Advanced Applications*, volume 11447, pages 36–52, 2019.

[141] Y. Warnecke, M. Kuhn, F. Diederichs, T. J. Brix, L. Clever, R. Bergmann, D. Heider, and M. Storck. Towards fairness in synthetic healthcare data: A framework for the evaluation of synthetization algorithms. In *German Medical Data Sciences 2025: GMDS Illuminates Health*, pages 25–34. IOS Press, 2025.

[142] S. Yang, Y. Zhou, Y. Guo, R. Farneth, I. Marsic, and B. Randall. Semi-synthetic trauma resuscitation process data generator. In *2017 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 573–573, Aug 2017.

[143] F. Zandkarimi, P. Decker, and J.-R. Rehse. Fig4pm: a library for calculating event log measures. *ICPM Doctoral Consortium and Demo Track*, pages 27–28, 2021.

[144] L. Zellner. *Fusing Rule and Process Mining*. PhD thesis, Ludwig-Maximilians-Universität München, 2025.

[145] L. Zellner, F. Richter, J. Sontheim, Andrea Maldonado, and T. Seidl. Concept Drift Detection on Streaming Data with Dynamic Outlier Aggregation. In *Revised Selected Papers of the 2nd International Conference on Process Mining (ICPM 2020) Workshops*, pages 206–217, 2020.

[146] M. Zhang, H. Lin, S. Takagi, Y. Cao, C. Shahabi, and L. Xiong. Csgan: Modality aware trajectory generation via clustering-based sequence gan. In *2023 24th IEEE International Conference on Mobile Data Management (MDM)*, pages 148–157, Jul 2023.

[147] F. Zhou, R. Yin, G. Trajcevski, K. Zhang, J. Wu, and A. Khokhar. Improving human mobility identification with trajectory augmentation. *GeoInformatica*, 25:453–483, Jul 2021.

[148] Y. Zisgen, D. Janssen, and A. Koschmider. Generating synthetic sensor event logs for process mining. In J. De Weerdt and A. Polyvyanyy, editors, *Intelligent Information Systems*, Lecture Notes in Business Information Processing, pages 130–137, Cham, 2022. Springer International Publishing.

[149] I. Žliobaitė, A. Bifet, J. Read, B. Pfahringer, and G. Holmes. Evaluation methods and decision theory for classification of streaming data with temporal dependence. *Machine Learning*, 98(3):455–482, 2015.

List of Figures

2.1 Hierarchical specialization of data types, from ordered data to event data representations (logs and streams). 8
2.2 Folding trajectory for a small alpha-helical protein [67]. 9
2.3 Temperature and precipitation time series [62]. 9
2.4 Request for reimbursement at TU/e as dotted chart [133]. 10
2.5 Inductive Miner [77] example in BPMN notation in 5 steps 15
2.6 ILP Miner [135] example in BPMN in 5 steps 17
2.7 Split Miner [14] example in BPMN notation in 5 steps 19
2.8 In [96], advances in the refinement of knowledge in empirical research come with increasing technical sophistication, data requirements, and scientific rigor. 21
2.9 Adapted from Rehse et al. [108], conceptual structure of reliability in experimental research. 22
2.10 Algorithm research pipeline and associated validity concerns. . . . 24
2.11 Local, global, and universal representational bias in process mining research. 25

3.1 Conceptual workflow of empirical steps from algorithm engineering [96] 34
3.2 Conceptual workflow of the dissertation's approach, showing cumulative empirical steps (x-axis) and paper-specific components (y-axis, cf. Section 1.1). 35
3.3 Sequential processes in healthcare and baking 37
3.4 AND-branching processes in healthcare and baking 37
3.5 Clustered correlation matrix using hierarchical clustering with single linkage and Pearson correlation as distance [83] between event data features [94] 39
3.6 Correlation plot for selected features from Figure 3.5 40
3.7 Visual representation of our experimental trial set-up. Annotations in black indicate how many combinations result, and in green indicate how many of these were feasible. 42
3.8 Our 22,000 generated and tested event logs cover and enrich the feature space. 42

3.9 Evaluation metrics of process discovery algorithm results from our 22,000 generated (orange) vs. real event logs (blue). 43

4.1 Extended clustered correlation matrix generated using full hierarchical clustering with single linkage and Pearson correlation distance [83]. 53

List of Tables

2.1 Illustrative overview of main process mining tasks. 13
2.2 Knowledge extension categories and corresponding research methods in algorithm engineering. 20
2.3 Process mining crimes' connection to validity concerns and representational bias. 30