UHASSELT

KNOWLEDGE IN ACTION

# Faculteit Bedrijfseconomische Wetenschappen

master handelsingenieur in de beleidsinformatica

***Masterthesis***

## *A comparison of rule mining algorithms*

**Maarten Dupont**

Scriptie ingediend tot het behalen van de graad van master handelsingenieur in de beleidsinformatica

**PROMOTOR :**

Prof. dr. Mieke JANS

**BEGELEIDER :**

Mevrouw Manal LAGHMOUCH

**2022**
**2023**

UHASSELT

KNOWLEDGE IN ACTION

**www.uhasselt.be**
Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek


# Faculteit Bedrijfseconomische Wetenschappen

master handelsingenieur in de beleidsinformatica

***Masterthesis***

***A comparison of rule mining algorithms***

**Maarten Dupont**

Scriptie ingediend tot het behalen van de graad van master handelsingenieur in de beleidsinformatica

**PROMOTOR :**

Prof. dr. Mieke JANS

**BEGELEIDER :**

Mevrouw Manal LAGHMOUCH

# Een vergelijking van rule mining algoritmes

Maarten Dupont

Universiteit Hasselt, 3590 Diepenbeek, België

**Rule mining is een belangrijke techniek binnen conformance checking voor het identificeren van de onderliggende relaties, patronen en regels tussen de variabelen in de data. Echter is het moeilijk om de verschillende technieken te vinden en te bepalen welke techniek het beste resultaat zal leveren. Hierdoor wordt er vaak geen rekening gehouden met andere technieken die andere, maar misschien betere resultaten opleveren. In deze studie zullen vijf rule mining technieken met elkaar vergeleken worden en zal hun capaciteit om regels te halen uit procesafwijkingen bekeken worden. Hiervoor zullen de technieken individueel getest worden op dezelfde artificiële maar realistische dataset. De resultaten tonen grote verschillen in de hoeveelheid regels die gevonden worden, de kwaliteit van deze regels en in de mogelijk om te werken met ongelabelde afwijkingen.**

***Keywords:*** *Association rule mining, Sequential rule mining, Clustering-based rule mining, Decision tree rule mining, Declarative process mining*

# 1 Introductie

Process mining is een familie van data-analysetechniek die gebruikt wordt om processen te ontdekken, te monitoren en te verbeteren door kennis en inzichten te verkrijgen uit data. Event logs worden gegenereerd door informatiesystemen en leggen hierin de acties van gebruikers en systemen vast wanneer deze een interactie hebben met bedrijfsprocessen (Setiawan & Yahya, 2018). Deze event logs worden gebruikt door de process mining algoritmes om procesmodellen te ontdekken, processtromen te visualiseren, procesprestaties te meten en procesknelpunten te identificeren. Process mining is hierdoor een veelgebruikte techniek voor organisaties om de efficiëntie en effectiviteit van bedrijfsprocessen te verbeteren. (van der Aalst, 2016)

Een belangrijk onderdeel binnen process mining is conformance checking. Conformance checking is een proces waarbij een normatief procesmodel vergeleken wordt met de waargenomen gebeurtenissen uit een event log. Hierdoor kan men controleren of de daadwerkelijke uitvoering van het proces overeenkomt met het procesmodel (Carmona et al., 2018). Het doel van conformance checking is om afwijkingen in het proces te identificeren. De gevonden afwijkingen moeten steeds van twee kanten bekeken worden: (1) is het ontdekte model fout en weerspiegelt het dus niet de realiteit en (2) de gevallen wijken af van het ontdekte model en corrigerende maatregelen zijn nodig (van der Aalst, 2016). Om een onderscheid te kunnen maken tussen deze twee aspecten is de kennis van een domeinexpert vereist. Door de groei in complexiteit van procesmodellen en hierdoor dus ook de toename van afwijkingen, wordt het manueel verwerken van de afwijkingen steeds moeilijker. Hierdoor is er dus nood aan automatisering in het identificeren en aanpakken van procesafwijkingen. Om dit proces te automatiseren kan rule mining toegepast worden om regels over het proces te genereren, die ons meer kunnen vertellen over afwijkingen in het proces (Rozinat & van der Aalst, 2008).

Rule mining, ook wel rule learning genoemd, is een techniek die gebruikt wordt om onderliggende relaties, patronen en regels tussen de variabelen in de data te ontdekken. Deze techniek wordt toegepast in verschillende domeinen zoals het analyseren van consumentengedrag, fraude detectie, financiën en de gezondheidszorg om eerder zeer moeilijk waarneembare kennis te achterhalen. Aan de hand van deze relaties kunnen er echter ook business regels opgesteld worden die men vervolgens kan gebruiken voor het modelleren en verbeteren van processen (Carmona et al., 2018). Er bestaan verschillende soorten rule mining technieken, waaronder association rule mining, decision tree mining, sequential rule mining en andere technieken. Toch valt op dat association rule mining de meest populaire techniek is. Dit wordt mogelijk verklaard door de eenvoudige en intuïtieve methode die deze techniek toepast om patronen en relaties te ontdekken in data. Verder kan het gebruikt worden voor een reeks aan toepassingen in verschillende domeinen. Daarnaast is deze techniek ook zeer schaalbaar en kan het door gebruik te maken van optimalisatietechnieken ook grote hoeveelheden data verwerken. Dit maakt association rule mining dus een zeer aantrekkelijke keuze, maar daarom niet altijd de beste. Zo hebben de andere

technieken specifiekere toepassingen waarvoor zij beter geschikt kunnen zijn (Karthikeyan & Ravikumar, 2014). Het is dus steeds van belang om verschillende soorten algoritmes te bekijken bij het bepalen van een geschikt algoritme. Echter valt op dat hier in slechts weinig onderzoeken rekening mee wordt gehouden. Er wordt ook weinig vergeleken tussen de verschillende technieken. De meeste onderzoeken vergelijken enkel de prestatie van de verschillende algoritmes binnen een bepaalde techniek zoals bijvoorbeeld het werk van Prithiviraj & Porkodi (2015). In deze paper worden de verschillende algoritmes binnen de association rule mining techniek vergeleken met elkaar. Dit soort studies geven een goed overzicht van de verschillende algoritmes binnen een techniek, maar er ontbreekt een duidelijk overzicht over de verschillende technieken heen in de huidige literatuur. Het vergelijken van de verschillende technieken zorgt voor een overzicht van de prestaties van deze technieken.

In deze studie worden verschillende rule mining technieken vergeleken met elkaar en wordt hun geschiktheid voor het achterhalen van business regels uit procesdata bestaande uit afwijkingen onderzocht. Dit wordt gedaan door elke techniek individueel te testen op een reeks van procesafwijkingen en hun resultaten op deze test met elkaar te vergelijken. Er zijn vijf rule mining technieken geanalyseerd, namelijk: association rule mining, sequential rule mining, clustering-based rule mining, decision tree mining en declarative process mining. Het resultaat van deze testen geeft een duidelijk overzicht van de prestaties van deze technieken bij het analyseren van procesafwijkingen. Het begrijpen en vergelijken van deze rule mining-technieken is namelijk essentieel voor het selecteren van de meest geschikte aanpak voor een bepaalde dataset en mining-doelstelling.

De rest van deze studie is als volgt ingedeeld. In sectie 2 wordt het gerelateerde werk bekeken. De methodologie van de studie wordt geïntroduceerd in sectie 3. Hierna volgt het uitvoeren van een experiment in sectie 4. Tot slot worden in sectie 5 de resultaten van het experiment besproken en de verschillende algoritmes met elkaar vergeleken.

## 2 Gerelateerd werk

Rule mining speelt een cruciale rol bij het achterhalen van onderliggende patronen, relaties en regels die waardevolle inzichten kunnen geven en keuzes kunnen ondersteunen. Rule mining kan voor twee doelen toegepast worden binnen process mining. Zo kan rule mining process mining ondersteunen in het ontdekken van processen door procesregels te genereren. Hiernaast kan rule mining gebruikt worden voor conformance checking. Hier wordt rule mining gebruikt in combinatie met process mining om patronen en relaties in event logs te identificeren die kunnen wijzen op procesafwijkingen of inefficiënties.

## 2.1 Rule mining technieken

Association rule mining werd voor het eerst gebruikt door Agrawal et al. (1993) en is momenteel één van de meest gebruikte rule mining algoritmes. Het werd origineel gebruikt om correlaties te vinden in het consumentengedrag van klanten. Doorheen de jaren is association rule mining geëvolueerd en uitgebreid en wordt het in veel domeinen toegepast. Ook wordt het steeds meer gebruikt bij het analyseren van event logs. Zo gebruikte Chen & Wu, (2005) association rule mining om het orderpickingproces in hun warenhuis te verbeteren door regels te creëren over het consumentengedrag van de klant. De gevonden regels toonde aan welke artikelen vaak samen besteld werden en welke artikelen dus best dicht bij elkaar moest liggen in het warenhuis om het orderpickingproces efficiënter te maken. Hiernaast werd door Khan & Parkinson (2018, 2019) association rule mining gebruikt om conformance checking toe te passen door regels te halen uit beveiliging event logs. De gevonden regels zorgden ervoor dat geen domeinexpert meer nodig was om de beveiliging van systemen te monitoren. Hierdoor konden de gevonden patronen gebruikt worden in het automatiseren van het beveiligingssysteem. Verder gebruikte Shrivastava et al. (2011) het association rule mining algoritme Apriori voor het analyseren van error logs. Deze regels gaven gebruikers een duidelijker beeld over wat de oorzaak was van de error en hoe deze eventueel vermeden kan worden in de toekomst.

Sequential rule mining is nog een veel gebruikte techniek voor het analyseren van event logs. Deze techniek werd ontwikkeld door Agrawal & Srikant (1995) en houdt rekening met de volgorde waarin activiteiten plaatsvinden. Het werk van Abdelwahab et al. (2022) vergelijkt de verschillende sequential rule mining algoritmes met elkaar. Deze studie biedt een goed overzicht over de verschillende sequential rule mining technieken en toont aan dat de keuze van het sequential rule mining algoritme neer komt op een afweging tussen efficiëntie en effectiviteit. Setiawan & Yahya (2018) paste sequential rule mining toe voor het identificeren van sequentiële regels voor het productieproces. Vervolgens werd met deze regels conformance checking gedaan en werden de regels gebruikt om het bestaande procesmodel te verbeteren. Ook Husák et al. (2020) maakte gebruik van sequeltial rule mining om patronen te vinden die kunnen wijzen op cyberaanvallen. Door deze patronen te herkennen kunnen dit soort aanvallen eerder gedetecteerd en bestreden worden.

Door de enorme hoeveelheid data die bedrijven bijhouden ligt de focus steeds meer op het versnellen van rule mining. Eén manier om het genereren van regels te versnellen is door gebruik te maken van clustering-based rule mining. Hierbij zal de data eerst gegroepeerd worden in clusters vooraleer er regels worden gegenereerd. Dit zorgt ervoor dat het proces niet alleen sneller verloopt, maar dat er ook gerichtere regels gevonden kunnen worden binnen elke cluster die misschien niet gevonden kunnen worden bij het analyseren van alle data. Mirebrahim et al. (2017) gebruikte een clustering-based rule mining algoritme voor het monitoren van energieverbruik in een universiteit. Deze aanpak zorgde ervoor dat afwijkingen in het energieverbruik snel en efficiënt gedetecteerd konden worden. Ook Öztayşi et al. (2022) behaalde betere resultaten door

eerst fuzzy clustering toe te passen op e-commerce data en vervolgens association rule mining toe te passen op elke cluster. De resultaten laten zien dat de voorgestelde methode in staat is om patronen in de gegevens te identificeren die niet worden vastgelegd door traditionele methoden voor het zoeken naar associatieregels. Deze nieuwe patronen geven ondernemingen een beter zicht op het consumentengedrag van klanten. Hiernaast gebruikte Riaz et al. (2014) ook clustering based association rule mining om productaanbevelingen in online winkels te optimaliseren. De resultaten laten zien dat door eerst te clusteren vooraleer association rule mining toegepast werd, de gevonden regels beter in staat zijn om productaanbevelingen te identificeren die relevanter en persoonlijker zijn voor elke klant, wat leidt tot meer verkopen en klanttevredenheid. Ook werden de regels gebruikt in het beter identificeren van promotiecampagnes gericht op het individuele gedrag van de klant.

Regels kunnen ook gegenereerd worden door gebruik te maken van beslissingsbomen. Deze regels worden weergegeven in de vorm van een boomstructuur. Het creëren van de beslissingsboom hangt af van de gekozen split criteria (waar de takken van de boom splitsen), de snoeiregels (hoe de grootte van de boom beperkt wordt) en de stopregels (waar de takken stoppen). Imai et al. (2019) maakte gebruik van een decision tree voor het creëren van een model dat de bijwerkingen en de ernst van de bijwerking van een nieuw medicijn voorspelt. Verder gebruikte Jeihouni et al. (2020) een decision tree voor het genereren van regels die helpen bij het identificeren van grondwaterzones van hoge kwaliteit. Hiernaast kon er door naar de beslissingspunten te kijken achterhaald worden welke attributen het belangrijkste waren bij het bepalen van de kwaliteit van grondwater. Hiernaast maakte Elacio et al. (2020) gebruik van een decision tree voor het detecteren van afwijkingen in het gedrag van personeel tijdens het productieproces. Deze afwijkingen werden vervolgens gebruikt bij het identificeren van de belangrijkste factoren die van invloed zijn op het verloop van werknemers alsook het identificeren van werknemers met een grote kans op verloop. Dit geeft managers meer informatie over werknemers en maakt het mogelijk om gerichte interventies toe te passen om de werknemer te behouden. Ook Tayefi et al. (2017) koos voor het werken met een decision tree om afwijkingen op te merken in patiënten met coronaire hartziekte. De regels gevonden door de decision tree kunnen gebruikt worden om coronaire hartziekte sneller te identificeren bij patiënten en geeft de artsen meer inzicht in de grootste risicofactoren.

Tot slot kan er ook nog gebruik gemaakt worden van declarative process mining. Dit is een variant van process mining die zich meer focust op het analyseren van de declaratieve aspecten van het proces zoals de beperkingen, doelen en de business regels. Door zich te focussen op de beperkingen en de hoofddoelen van het proces, kan het een beter beeld van het proces aanbieden en geeft het inzicht op het gebied van inefficiënties, knelpunten en compliance. Zo gebruikte Rovani et al. (2015) deze methode om inzicht te krijgen in de processen van een ziekenhuis. De resultaten gaven verschillende inefficiënties weer binnen het ziekenhuis die verbeterd kunnen worden om de zorg voor de patiënten te verbeteren. Ook Mertens et al. (2022)

gebruikte declarative process mining om het proces van een spoedafdeling te analyseren. De resultaten laten zien dat de geïntegreerde methode zowel declaratieve als procedurele kennis in het spoedeisende zorgproces kan identificeren, inclusief beslismomenten, beslissingsregels en de uiteindelijke beslissing. Hiernaast paste Ardimento et al. (2020) deze methode toe voor het detecteren van malware. Zo konden de gevonden patronen gebruikt worden om afwijkingen in event logs te detecteren die indicatief zijn voor malware.

## 2.2 Rule mining algoritmes

Na het bekijken van de gerelateerde werken is het ook essentieel om dieper in te gaan op de specifieke algoritmes binnen elke techniek. Dit geeft een duidelijker beeld van hoe elke techniek werkt en welke algoritmes binnen elke techniek gebruikt worden.

### 2.2.1 Association rule mining

Association rule mining is een machine learning algoritme ontworpen om de verborgen relaties tussen variabelen in data te ontdekken. De probleemomschrijving van een association rule mining algoritme zoals beschreven door Boutorh & Guessoum (2016) ziet er uit als volgt: stel I= {i1 , i2 , ... , im} is een set van items en T=t1, t2 , ... , tn is een set van transacties. Elke transactie is een set van items zodat ti $\subseteq I$. Associatieregels geven een implicatie weer in de vorm: X$\Rightarrow Y$, waarbij X $\subset I$, Y $\subset I$ en X$\cap Y = \emptyset$. X is dan een set van items genaamd een itemset. In de regel X $\Rightarrow Y$ wordt X het antecedent en Y consequente van de regel genoemd. Deze techniek gaat dus op zoek naar welke items samen voorkomen in dezelfde transactie. De regels impliceren dat bij het voorkomen van het antecedent, de consequent ook zal voorkomen in dezelfde transactie met een zekere betrouwbaarheid. Support en betrouwbaarheid zijn de twee meest belangrijke kwaliteitsmaatstaven die gebruikt worden om te bepalen of een regel interessant is of niet. De support van een associatieregel wordt gedefinieerd als het percentage van alle transacties die zowel X als Y bevatten. De support van een regel wordt berekend aan de hand van de volgende formule

$$\text{support}(\text{X}\Rightarrow Y)=\frac{\sigma(X \cup Y)}{n}$$

$\sigma(X \cup Y)$ is het aantal transacties dat alle items van de regel bevatten en n is het totaal aantal transacties. De betrouwbaarheid van een regel is het percentage van transacties die zowel X en Y bevatten over het aantal transacties die X bevatten ($\sigma(X)$). De betrouwbaarheid van een regel kan berekend worden aan de hand van de volgende formule

$$\text{betrouwbaarheid}(\text{X}\Rightarrow Y)=\frac{\sigma(X \cup Y)}{\sigma(X)}$$

De keuze van algoritme speelt ook een grote rol in hoe de implementatie verloopt. Het algoritme dat het meeste gebruikt wordt is het Apriori algoritme. Dit algoritme werkt in twee delen. Eerst zal het op zoek gaan naar alle frequente item sets waarvan het support hoger is dan

de vooropgestelde minimum support. Daarna zal het hieruit alle regels halen die ook voldoen aan de vooraf opgelegde minimum betrouwbaarheid. Alhoewel het Apriori algoritme het eenvoudigste is om te gebruiken is het niet efficiënt en zal in de problemen lopen bij het toepassen op grote hoeveelheden data.

Een tweede veelvoorkomend association rule mining algoritme is het Eclat algoritme. Dit algoritme gebruikt backtracking om de frequente itemsets te vinden. Hierdoor zal het algoritme eerste kijken naar de grotere itemsets waardoor er bespaard kan worden op de tijd die nodig is om de kleinere itemsets te bekijken. Dit zorgt ervoor dat dit algoritme minder geheugen nodig heeft om uitgevoerd te worden en hierdoor ook sneller is dan het Apriori algoritme.

Een derde veelgebruikt algoritme is het frequent pattern (FP) growth algoritme. Dit algoritme zal in zijn eerst pas door de data alle items die voorkomen tellen. Aan de hand hiervan zal het algoritme in zijn tweede pas door de data een FP-tree maken die de structuur van elke transactie weergeeft. Hieruit kunnen dan de associatieregels gehaald worden.

Association rule mining is één van de meest gebruikte technieken binnen rule mining vanwege het vermogen om patronen en relaties tussen items in een dataset te ontdekken. Verder is association rule mining eenvoudig te implementeren en interpreteren en kan het toegepast worden in verschillende domeinen. (Jain et al., 2013)

### 2.2.2 Sequential rule mining

Sequential rule mining is een rule mining algoritme dat op zoek gaat naar frequente patronen in sequentiële data. Het focust zich vooral op de volgorde waarmee de gebeurtenissen in het proces plaatsvinden, in tegenstelling tot alleen zoeken naar de frequentie en het gelijktijdig voorkomen van gebeurtenissen zoals gedaan wordt bij association rule mining.

De werking van het algoritme is echter wel zeer gelijkend met dat van association rule mining met het grote verschil tussen de twee dat er bij sequential rule mining rekening wordt gehouden met de volgorde waarin activiteiten plaatsvinden. Zo betekent de associatieregel A $\Rightarrow B$ äls A plaatsvindt, dan vindt ook B plaats", maar hiertussen kunnen nog C en D plaatsvinden. De sequentiële regels impliceren dat äls A plaatsvindt, dan wordt dat gevolgd door B". Dit betekent dat A en B elkaar opvolgen terwijl dit bij associatieregels niet altijd het geval is.

Om te beginnen moet er eerst een waarde gekozen worden voor de minimum support (minSupp) en de minimum betrouwbaarheid (minConf). Deze parameters geven aan hoe het algoritme de gevonden regels moet beoordelen. Hiermee kan het algoritme aan de slag en zal het algoritme frequente patronen zoeken tot alle frequente (k-1) sequentiële patronen gevonden zijn. Voor de patronen die voldoen aan de minSupp zal de betrouwbaarheid berekend worden. Hierdoor zullen enkel nog de patronen overgehouden worden die voldoen aan de minConf.

Het resultaat van het algoritme is dus een reeks van frequente sequentiële patronen (Agrawal & Srikant, 1995).

Er zijn verschillende sequential rule mining algoritmes beschikbaar. Elk algoritme gebruikt dezelfde input en resulteert in dezelfde output maar verschillen in prestatie door het toepassen van verschillende optimalisaties, zoekstrategiën en datastructuren. Prefixspan is een algoritme dat gebouwd is op het idee om een prefix-boomstructuur te gebruiken om de sequentiële patronen weer te geven. De prefixen gebruikt om de structuur van de boom te bepalen zijn gedeeltelijke sequentiële patronen die als startpunt gebruikt worden bij het ontdekken van de volledige sequentiële patronen. De frequente sequentiële patronen worden door de boom op verschillende prefixen geprojecteerd en worden vervolgens recursief ontdekt door een devide-and-conquer framework toe te passen. Dit houdt in dat de gebruikte prefix geleidelijk wordt uitgebreid om alle sequentiële patronen te ontdekken. Een ander veelgebruikt sequential rule mining algoritme is het SPADE algoritme. SPADE maakt gebruik van een verticale weergave van de data en past hier een depth-first aanpak op toe om de frequente reeksen te genereren. Dit betekent dat dit algoritme begint met een leeg patroon en dit geleidelijk uitbreidt door items toe te voegen om zo langere patronen te ontdekken. Verder is GSP één van de eerste sequential rule mining algoritmes. Dit Apriori gebaseerde algoritme gebruikt een sliding window aanpak om de frequente sequentiële patronen te vinden. De sliding window aanpak gebruikt de tijd om sequentiële patronen te vinden door rekening te houden met een vast aantal recente gebeurtenissen tegelijk. Hierdoor is GSP zeer geschikt voor het identificeren van patronen in time-series data (Mabroukeh & Ezeife, 2010).

Sequentiële rule mining is een nuttige techniek voor het ontdekken van patronen en relaties tussen gebeurtenissen die in een specifieke volgorde of sequence plaatsvinden. Het is met name handig in toepassingen waarbij timing en ordening belangrijk zijn, zoals bij process mining of event log analyse.

### 2.2.3 Clustering-based rule mining

Clustering-based rule mining heeft als doel het rule mining proces te versnellen en interessantere regels te vinden door eerst de data te clusteren. Deze methode zal dus eerst de data clusteren volgens een bepaalde techniek en zal vervolgens op elke individuele cluster het rule mining algoritme toepassen. Door eerst te clusteren zal het rule mining algoritme niet alleen sneller kunnen werken, maar zal het ook gerichter zoeken naar regels binnen elke cluster. Zo kan het dus regels achterhalen die relevant zijn binnen een bepaalde cluster, maar daarom niet over de hele data heen. Hierdoor biedt deze methode meer inzicht op specifieke aspecten binnen de data. Echter zorgt dit er ook voor dat er meer algemene regels gemist kunnen worden omdat ze niet relevant zijn binnen de clusters zelf, maar wel over het geheel van de data. Hierdoor is het kiezen van een geschikte clusteringtechniek van groot belang (Riaz et al., 2014).

De algoritmes die het vaakst voorkomen om clustering-based rule mining toe te passen zijn K-means, Hierarchical, Density-Based Spatial Clustering of Applications with Noise en Expectation-Maximization. Deze clustering technieken worden meestal gebruikt in combinatie met association rule mining, maar kunnen ook toegepast worden in combinatie met andere rule mining algoritmes. Het grote verschil tussen de verschillende algoritmes is hoe de algoritmes hun clusters en welke elementen tot die clusters behoren bepalen. K-Means doet dit door willekeurige clusters te plaatsen en deze aan te passen aan de hand van de gemiddelde afstand tussen het centrum van de cluster en de elementen in de cluster. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) groepeert gegevenspunten die zich in dichtbevolkte gebieden bevinden en negeert uitschieters als ruis. Het definieert clusters als gebieden met een hoge dichtheid en breidt ze uit totdat de dichtheid onder een bepaalde drempel zakt. EM (Expectation-Maximization) berekent de kans dat elk element behoort tot een bepaalde cluster en verdeelt aan de hand van deze kans de elementen over de verschillende clusters (Mirebrahim et al., 2017).

### 2.2.4 Decision tree rule mining

Decision tree rule mining is een rule mining techniek die een hiërarchische boomstructuur aanneemt. Deze beslissingsboom bestaat steeds uit drie soorten nodes namelijk: root nodes, internal nodes and leaf nodes. De root node is het beginpunt van de boom en bevindt zich dan ook bovenaan de boomstructuur. De internal nodes vormen de verbindingen tussen de root node bovenaan en de de leaf nodes onderaan de boom. Elke internal node stelt een test op een bepaald attribuut voor waar er zich een splitsing van de takken zal plaatsvinden aan de hand van de resultaten op die test. De leaf nodes zijn de eindpunten van de boom en geven het resultaat van die tak weer. Het bouwen van een decision tree gebeurt volgens een recursief partitieproces dat op basis van splitsingscriteria de data herhaaldelijk verdeelt. Als de boom gemaakt is kunnen er regels gegenereerd worden door de verschillende paden van root node tot leaf node te volgen (Gama et al., 2006).

Er zijn drie veelvoorkomende decision tree rule mining algoritmes, namelijk CHAID (Chi-squared Automatic Interaction Detection) (Kass, 1980), CART (Classification and Regression Trees)(Breiman, 1984) en C4.5 (Quinlan, 1993). Het verschil tussen de verschillende algoritmes is hoe het algoritme de splitsingscriteria in de internal nodes bepaalt. Zo gebruikt CHAID de chi-square test voor discrete variabelen en de f-test voor continue variabelen om de splitsingen in de boom te bepalen. CART gebruikt de GINI index om te bepalen waar de splitsing zullen plaatsvinden, terwijl C4.5 dit doet aan de hand van de entropy index.

Decision tree rule mining is een handig hulpmiddel voor het identificeren van belangrijke attributen of kenmerken die bijdragen aan een specifieke uitkomst of afwijking. Verder zijn decision trees snel, produceren ze gemakkelijk interpreteerbare modellen en kunnen ze grote datasets met veel attributen aan.

### 2.2.5 Declarative process mining

Declarative process mining richt zich op het ontdekken van de declaratieve procesmodellen die verborgen zijn in de event logs van een informatiesysteem. Het doel van declaratieve process mining is om informatie uit event logs te halen die ons in staat stelt het gedrag van het systeem te begrijpen in termen van de doelen die het probeert te bereiken, de beperkingen die het moet respecteren en de voorkeuren die het heeft. Deze informatie kan vervolgens worden gebruikt om declaratieve procesmodellen te bouwen die het gedrag van het systeem op een meer abstracte en begrijpelijke manier beschrijven dan het onbewerkte event log. Dit gebeurt door het algoritme een reeks beperkingen, bedrijfsregels, gebeurtenisvoorwaarden of andere (logische) uitdrukkingen, die eigenschappen van en afhankelijkheden tussen activiteiten in een bedrijfsproces weergeven, te definiëren. Vervolgens worden op een impliciete manier alle alternatieve paden gespecificeerd en gedefinieerd als de paden die niet in strijd zijn met de bedrijfsregels. Met deze informatie zal het algoritme een decalaratief procesmodel proberen op te stellen (Goedertier et al., 2015).

Het meest gebruikte declarative process mining algoritme is de Declare miner. Dit algoritme werkt op basis van een reeks beperkingen. Het gebruikt een combinatie van declaratieve beperkingen en technieken voor het ontdekken van processen om een procesmodel te genereren. Ook het Alpha Algorithm en Inductive Miner zijn nog twee veelvoorkomende declarative process mining algoritmes. Alpha-algoritme ontdekt procesmodellen door beperkingen op het gedrag van het systeem te definiëren. Het gebruikt een combinatie van een event log en declaratieve beperkingen om een procesmodel te genereren. Inductive Miner doet dit op basis van observaties van gedrag in de event logs. Het gebruikt een reeks procesboomstructuren om een procesmodel te genereren (Polyvyanyy et al., 2021).

Declarative process mining wordt gebruikt voor het identificeren van patronen en regels in event logs, met name in toepassingen waarbij een hoog niveau van inzicht in het procesgedrag vereist is. Ook kan het complexe, grootschalige procesmodellen aan en kan het patronen en regels identificeren die moeilijk of onmogelijk te identificeren zijn met andere rule mining algoritmes.

# 3 Methodologie

Om te achterhalen welke rule mining algoritmes het meest geschikt zijn voor het analyseren van procesafwijkingen zullen verschillende algoritmes individueel geëvalueerd worden op de hoeveelheid regels die ze creëren en de kwaliteit van deze regels. Dit creëert een overzicht van de effectiviteit van deze algoritmes. De algoritmes zullen getest worden op een realistische, maar artificieel gegenereerde, dataset en beoordeeld worden op hun capaciteit om relevante regels te genereren met als doel de afwijkingen sneller te identificeren en de oorzaak te begrijpen.

## 3.1 Artificiële dataset

Om de verschillende algoritmes te testen is er gebruik gemaakt van een artificiële datasets uit het werk van Laghmouch et al. (2020). Deze datasets bestaat uit synthetische logs gebaseerd op declaratieve procesmodellen van een procure-to-pay proces.

De datasets die gebruikt worden in deze studie bestaan uit 1000 cases die een uitzondering zijn en een tweede dataset met daarin 1000 cases die een anomalie zijn in het procure-to-pay proces. Uitzonderingen zijn cases die afwijken van het normatief model, maar door bepaalde condities wel acceptabel zijn. Anomalieën weerspiegelen mogelijke problemen in het proces die opgevolgd moeten worden. Deze afwijkingen zijn dus al gelabeld, maar dit is gedaan door een domeinexpert die moet bepalen of een case een uitzondering of een anomalie is. Om te testen of de verschillende rule mining algoritmes in deze paper dit ook kunnen, werden de twee datasets gecombineerd. De prestatie van de verschillende algoritmes op de gecombineerde dataset zullen dan vergeleken kunnen worden met de prestatie op de individuele datasets. Als de resultaten overeenkomen toont dit aan dat er weinig verschil is tussen het analyseren van gelabelde data en niet gelabelde data. In dat geval is er dus geen domeinexpert meer nodig om het onderscheid tussen de twee te maken.

## 3.2 Studieopzet

In deze studie zullen vijf rule mining technieken met elkaar vergeleken worden, namelijk: association rule mining, sequential rule mining, clustering-based rule mining, decision tree rule mining and declarative process mining. Deze technieken werden geselecteerd op hun vermogen om business regels te genereren. Om dit te bepalen is er gekeken naar andere papers die rule mining in deze context gebruiken. Deze papers zijn al eerder aangehaald in het gerelateerde werk.

Om de prestatie van de technieken te beoordelen zal er gekeken worden naar een aantal criteria. Elke techniek genereert zijn eigen soort regels met als gevolg dat we geen individuele regels kunnen vergelijken over de technieken heen. Hierdoor zal er gekeken worden naar de hoeveelheid regels die gegenereerd worden. De hoeveelheid regels vertelt ons meer over hoe eenvoudig het is om de regels hierna te verwerken. Hoe meer regels er gegenereerd worden, hoe meer werk het is om deze regels te beoordelen en te implementeren. Ook zal de kwaliteit van de regels beoordeeld worden door te kijken naar de gemiddelde support en betrouwbaarheid van alle gevonden regels. Dit geeft aan in hoeveel procent van de cases de gevonden regels van toepassing zijn en hoe betrouwbaar ze zijn. Tot slot zal er nog gekeken worden naar de prestatie van de technieken op de verschillende datasets. Door de prestatie op de gecombineerde dataset te vergelijken met die van de individuele datasets kunnen we zien of dezelfde regels gevonden worden. Om dit te bepalen zal er gekeken worden naar hoeveel procent van de regels gevonden bij de individuele datasets, ook terugkomen in de combinatie van de twee. Hoe hoger dit percentage, hoe minder het nodig is om op voorhand onderscheid te maken tussen de twee soorten afwijkingen.

# 4 Resultaten

Om de effectiviteit van de verschillende algoritmes in het analyseren van procesafwijkingen te tonen, zullen de verschillende algoritmes getest worden op de artificiële datasets. De verschillende algoritmes zullen getest worden op de dataset met enkel de uitzonderingen, enkel de anomalieën en een combinatie van de twee. Dit zorgt ervoor dat ook binnen elk algoritme gekeken kan worden of er een onderscheid te vinden is tussen de resultaten bij het toepassen op de verschillende datasets. Deze testen zullen voornamelijk uitgevoerd worden in Python (versie 3.6), R (versie 4.3) en in RuM.

## 4.1 Association rule mining

### 4.1.1 Data preprocessing

Het eerste algoritme dat getest zal worden is association rule mining (ARM). Hiervoor zal het Python pakket mlxtend gebruikt worden. Om ARM toe te passen op de data zal de data eerst gegroepeerd moeten worden per case en zal hierna one-hot encoding toegepast worden. One-hot encoding zorgt ervoor dat de activiteiten van een categorische variabele aangepast worden naar binaire variabele. Dit is nodig omdat ARM enkel binaire variabelen als input gebruikt.

### 4.1.2 Prestatie

Ten eerste zal er gekeken worden naar de combinatie van de twee datasets. Om de regels te genereren is er een minimum support van 20 procent en een minimum betrouwbaarheid van 60 procent gekozen. Met deze parameters genereert het Apriori algoritme 1763 regels. Deze regels hebben een gemiddelde support van 60 procent en een gemiddelde betrouwbaarheid van 80 procent. Zoals te zien in tabel 1 bestaan de regels steeds uit een antecedent en een consequent. Deze regels vertellen ons dat bijvoorbeeld de activiteiten 'Approve PR' en 'Create PO' in 71,5 procent van de cases samen voorkomen en dat als 'Approve PR' voorkomt in een case in 85,4 procent van de gevallen ook 'Create PO' voorkomt.

| Antecedent | Consequent | Support | Betrouwbaarheid |
|---|---|---|---|
| Approve PR | Create PO | 0,715 | 0,854 |
| Create PR | Approve PR | 0,713 | 0,852 |
| Pay | Approve PR | 0,805 | 0,832 |
| Approve PR | Receive goods or services | 0,71 | 0,848 |
| Receive invoice | Approve PR | 0,702 | 0,812 |

Tabel 1: Voorbeeld regels voor de combinatie van de twee datasets

Vervolgens kijken we naar de 1000 cases die anomalieën in het proces zijn. Deze cases worden op dezelfde manier als hierboven geanalyseerd. Deze data levert 1721 regels op met een gemiddelde support van 61 procent en een gemiddelde betrouwbaarheid van 82 procent. Verder valt op dat er 180 regels zijn met een zeer hoge support en betrouwbaarheid van meer dan 80 procent. Dit toont aan dat deze regels in bijna elke case voorkomen en dus van groot belang zijn. Om te valideren of de combinatie van de twee datasets gelijkaardige regels kan vinden, zullen de gevonden regels in de combinatie vergeleken worden met de regels uit de individuele datasets. Dit gebeurt door te kijken welke regels in beide gevallen voorkomen. Van de 1721 regels komen er 221 regels ook voor in de combinatie van de twee datasets. Verder zien we dat de combinatie 122 van de 180 regels met een zeer hoge support en betrouwbaarheid ook gevonden heeft. Dit toont aan dat de combinatie slechts 13 procent van de regels kon vinden, maar de combinatie vond wel 68 procent van de belangrijkste regels.

Ook de dataset met 1000 cases die een uitzondering zijn in het proces worden op dezelfde manier geanalyseerd. Het algoritme genereerde hier 1817 regels met een gemiddelde support van 59 procent en een gemiddelde betrouwbaarheid van 82 procent. Van de 1817 regels zijn er 50 regels met een zeer hoge support en betrouwbaarheid van meer dan 80 procent. Van de 1817 regels werden er 112 ook gevonden door de combinatie van de twee waarvan 20 regels tot de belangrijkste 50 regels hoorden. De combinatie kon dus slechts 6 procent van regels vinden en vond 40 procent van de belangrijkste regels. Hiernaast is er nog gekeken hoeveel regels de twee individuele datasets delen met elkaar. Er blijken 263 regels voor te komen in beide individuele datasets waarvan er 54 tot de belangrijkste regels voor de anomalie dataset behoren en 14 tot de belangrijkste regels voor de uitzondering dataset behoren. Van de 263 regels werden er 62 regels gevonden door de combinatie van de twee. Hierbij zaten 26 van de belangrijkste regels voor anomalieën en 12 van de belangrijkste regels voor de uitzonderingen.

| | Aantal regels | Gemiddelde support | Gemiddelde betrouwbaarheid | Gevonden door de combinatie |
|---|---|---|---|---|
| Combinatie | 1763 | 60% | 80% | 8% |
| Anomalieën | 1721 | 61% | 82% | 13% |
| Uitzonderingen | 1817 | 59% | 82% | 6% |

Tabel 2: Prestatie ARM op de verschillende datasets

In totaal is het de combinatie van de twee datasets dus gelukt om 271, oftewel 8 procent van de regels te vinden die ook gevonden werden door het analyseren van de individuele datasets. Hiernaast slaagde de combinatie er wel in om de belangrijkste 62 procent van de regels te vinden.

## 4.2 Sequential rule mining

### 4.2.1 Data preprocessing

De volgende methode die getest wordt, is sequential rule mining. Het Python pakket prefixspan is hiervoor gebruikt om het prefixspan algoritme toe te passen op de data. De data moet eerst aangepast worden vooraleer deze geanalyseerd kan worden door het algoritme. Zo moeten de activiteiten eerst weer gegroepeerd worden per case. Vervolgens moeten deze cases omgezet worden naar sequenties. Dit wordt gedaan door te kijken welke activiteiten dicht bij elkaar gebeuren en dus samen horen. Voor deze dataset is er voor gekozen om alle activiteiten in dezelfde case die binnen de acht uur van elkaar voorkomen samen te voegen. Dit zorgt ervoor dat er sequenties gecreëerd worden die deze vorm aannemen: ['Approve PR', 'Create PR', 'Receive invoice', 'Receive goods or services', ('Sign','Pay')]. In het geval van deze case zien we dat Sign en Pay extra tussen haakjes staan. Dit wil zeggen dat deze twee activiteiten dus gecombineerd werden omdat ze binnen de acht uur van elkaar plaatsvonden. Aan de hand van dit soort sequenties kan het algoritme nu achterhalen welke sequenties en activiteiten vaak samen voorkomen.

### 4.2.2 Prestatie

De eerste dataset die geanalyseerd wordt, is de combinatie van de twee datasets. Hiervoor is er gekeken naar sequenties met een minimum support van 5 procent oftewel sequenties die minstens 100 keer voorkomen. Het algoritme genereerde 247 sequenties die voldoen aan de minimum support en hebben gemiddeld een support van 12 procent. Van de 247 sequenties bevatten er 25 een combinatie van activiteiten.

| Support | Sequenctie |
|---|---|
| 708 | ['Sign', 'Receive goods or services'] |
| 507 | ['Receive goods or services', 'Receive invoice'] |
| 438 | ['Create PO', 'Receive goods or services'] |
| 265 | ['Sign', 'Receive goods or services', 'Pay'] |
| 187 | ['Sign', ('Create PO', 'Sign')] |

Tabel 3: Vaak voorkomende sequenties bij de gecombineerde dataset

De volgende dataset is de dataset met 1000 cases die anomalieën zijn in het proces. Net zoals hierboven zullen deze geanalyseerd worden met een minimum support van 5 procent oftewel sequenties die minstens 50 keer voorkomen. Het algoritme produceerde hier 191 sequenties die voldoen aan de minimum support. Deze sequenties hebben een gemiddelde support van 13 procent. Ook deze resultaten bevatten combinaties van activiteiten. Zo hebben 24 van de 191 sequenties een combinatie van activiteiten. Om de accuraatheid van de combinatie te testen is ook hier gekeken of de combinatie dezelfde sequenties kan vinden als de testen op de individuele datasets. Voor deze dataset kwamen 154 regels oftewel 81 procent van de regels ook terug in de combinatie. Ook 16 van de 24 sequenties met een combinatie van activiteiten kwamen in beide voor.

Ook de laatste dataset met 1000 cases die een uitzondering vormen in het proces werden op dezelfde manier getest. Dit zorgde voor de creatie van 390 sequenties met een minimum support van vijf procent. Deze sequenties hebben een gemiddelde support van 11 procent en er zijn 47 sequenties met een combinatie van activiteiten. Als we deze resultaten vergelijken met die van de combinatie komen 214 regels in beide voor waarvan er 23 bestaan uit een combinatie van de twee. Dit betekent dat 55 procent van de regels ook voorkomen in de combinatie van de twee. Hiernaast is er ook nog gekeken naar de overlap tussen de twee individuele datasets. Zo valt op dat 121 sequenties in alle drie de testen voorkomen, waarvan er 14 bestaan uit een combinatie van activiteiten.

| | Aantal regels | Gemiddelde support | Gevonden door de combinatie |
|---|---|---|---|
| Combinatie | 247 | 12% | 54% |
| Anomalieën | 191 | 13% | 81% |
| Uitzonderingen | 390 | 11% | 55% |

Tabel 4: Prestatie sequential rule mining op de verschillende datasets

Het sequential rule mining algoritme slaagde er dus in om 247 oftewel 54 procent van de sequenties gevonden door de individuele testen terug te vinden. Hiernaast vond het 25 oftewel 44 procent van de sequenties met een combinatie van activiteiten.

## 4.3 Clustering-based rule mining

### 4.3.1 Data preprocessing

De clustering-based rule mining methode zal getest worden door gebruik te maken van het sklearn pakket in Python om k-nearest-neighbor(KNN) toe te passen en het mlxtend pakket om hierna association rule mining toe te passen op elke cluster. Ook hier zal de data eerst gegroepeerd worden per case en zal er zoals bij association rule mining one-hot encoding toegepast worden om de activiteiten om te vormen naar een binaire variabele.

### 4.3.2 Prestatie

Vervolgens kan KNN toegepast worden op de data om de clusters te genereren. Echter moet bij het gebruik van KNN het aantal clusters op voorhand gedefinieerd worden. Dit wordt normaal gedaan door te kijken naar statistieken zoals de within-cluster sum of squares (WSS) die aangeeft hoe dicht de punten in de clusters bij elkaar liggen. Echter is onze enige variabele de activiteiten in elke case wat het dus onmogelijk maakt om afstand tussen punten in een cluster te berekenen en dus om WSS toe te passen. Hierdoor is er voor gekozen om te kijken naar het aantal unieke regels die er gegeneerd worden om te bepalen hoeveel clusters optimaal is voor deze dataset. De regels worden gegenereerd door association rule mining toe te passen op elke cluster.

Dezelfde methode als bij association rule mining is ook hier toegepast. Ook de parameters zijn gelijk gebleven. Zo is er steeds gezocht naar regels met een minimum support van 20 procent en een minimum betrouwbaarheid van 60 procent. We zien dat er bij twee clusters in totaal 2233 regels gegenereerd worden waarvan er 1850 uniek zijn. Bij drie clusters worden er meer regels aangemaakt namelijk 2905 en stijgt het aantal unieke regels ook tot 1932. Hierna zien we dat indien we het aantal clusters verhogen er in totaal meer regels gemaakt worden, maar dat het aantal unieke regels gelijk blijft. Hierdoor is er voor gekozen om te werken met drie clusters.

| Clusters | 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| Aantal regels | 2233 | 2905 | 3562 | 4125 |
| Unieke regels | 1850 | 1932 | 1932 | 1932 |

Tabel 5: Regels per aantal clusters

Om na te gaan of deze methode andere resultaten oplevert, zullen de resultaten van elke cluster vergeleken worden met de regels gevonden door enkel association rule mining toe te passen. De eerste cluster bevat 371 regels met een gemiddelde support van 43 procent en een gemiddelde betrouwbaarheid van 88 procent. Indien we deze regels vergelijken met degene gevonden door enkel association rule mining toe te passen, vinden we dat 154 van de 371 regels al eerder gevonden werden. Dit betekent dat deze cluster voor 58 procent bestaat uit unieke regels.

Cluster twee bestaat uit 602 regels met een gemiddelde support van 83 procent en een gemiddelde betrouwbaarheid van 92 procent. De cases in deze cluster zijn dus meer gelijkend op elkaar dan in de vorige cluster. Ook hier is er gekeken of deze regels al eerder voorkwamen bij association rule mining. Zo zien we dat 182 regels al eerder gevonden werden. Deze cluster produceerde dus ook meer unieke regels. Zo waren 70 procent van de regels uniek.

De derde en grootse cluster bevat 1932 regels met een gemiddelde support van 76 procent en een gemiddelde betrouwbaarheid van 87 procent. Als we ook hier kijken naar het aantal unieke regels zien we dat 450 regels al eerder gevonden werden. 1482 regels waren dus uniek wat overeenkomt met 77 procent van de gevonden regels.

| | Aantal regels | Gemiddelde support | Gemiddelde betrouwbaarheid | Gevonden door ARM |
|---|---|---|---|---|
| Cluster 1 | 371 | 43% | 88% | 58% |
| Cluster 2 | 602 | 83% | 92% | 70% |
| Cluster 3 | 1932 | 76% | 87% | 77% |

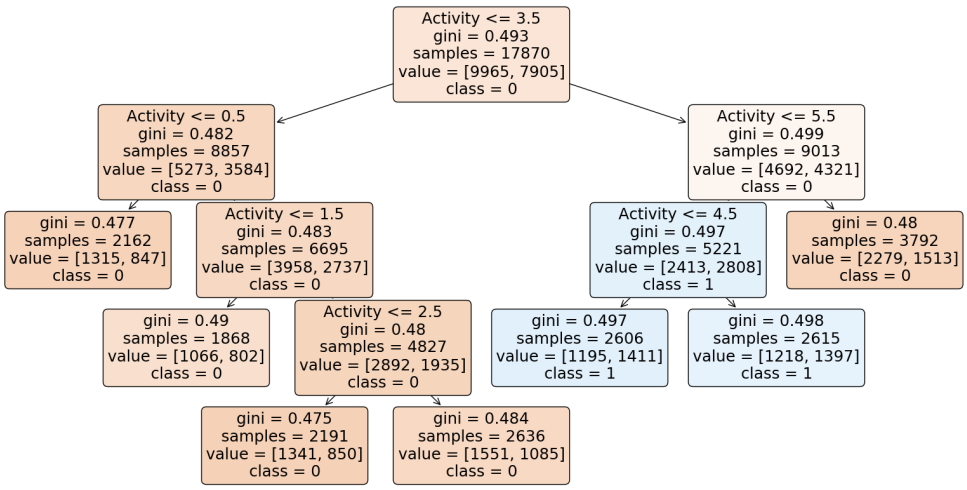
Tabel 6: Prestatie individuele clusters op de gecombineerde dataset

Door eerst te clusteren zijn er dus meer unieke regels gecreëerd door op zoek te gaan naar regels binnen elke cluster. Zo worden er 1482 nieuwe regels gevonden die niet gevonden werden door enkel association rule mining toe te passen. Dit komt overeen met 77 procent van de regels die dus uniek zijn.

## 4.4 Decision tree rule mining

### 4.4.1 Data preprocessing

Decision tree rule mining is getest door het algoritme de taak te geven een model te creëren dat probeert te voorspellen of een case een anomalie of een uitzondering is uit de combinatie van de twee. Vervolgens kunnen de internal nodes uit de boom omgezet worden naar regels die de keuzemogelijkheden verklaren. In eerste instantie is dit getest met het Python pakket sklearn. Echter bleek er al snel een probleem met dit pakket, namelijk dat het enkel met numerieke variabelen kan werken. Dit vormt een groot probleem voor onze case want de enige variabele die gebruikt wordt, is een categorische. De enige optie is om de categorische variabele om te zetten naar een numerieke variabele door elke activiteiten te nummeren. Hiermee kan het decision tree algoritme van sklearn aan de slag, maar valt het al snel op dat de verkregen uitkomst geen relevante informatie weergeeft. Zo splitst de boom op vlak van het nummer van de activiteiten, bijvoorbeeld splitst de boom in de eerste node volgens de variabele activity $\leq 3,5$. Dit betekent dat de activiteiten met een lager nummer naar links gaan en die met een hoger nummer naar rechts gaan in de boom. Deze manier van splitsen is echter niet correct voor onze data.




Figuur 1: Decision tree in Python die verkeerd splitst

Omdat het probleem met de categorische variabele onvermijdbaar is, is er ook gekeken naar het toepassen van decision tree rule mining in R. Hiervoor is het pakket rpart gebruikt. Dit pakket kan zowel gebruik maken van numerieke als categorische variabelen en lost het probleem

bij Python dus al op. Om de analyse in R uit te voeren is er voor gekozen om de cases te groeperen en de verschillende activiteiten in elke case te verdelen in verschillende variabelen naargelang hun volgorde in de case. De data gebruikt om de boom mee te trainen ziet er dus uit als volgt:

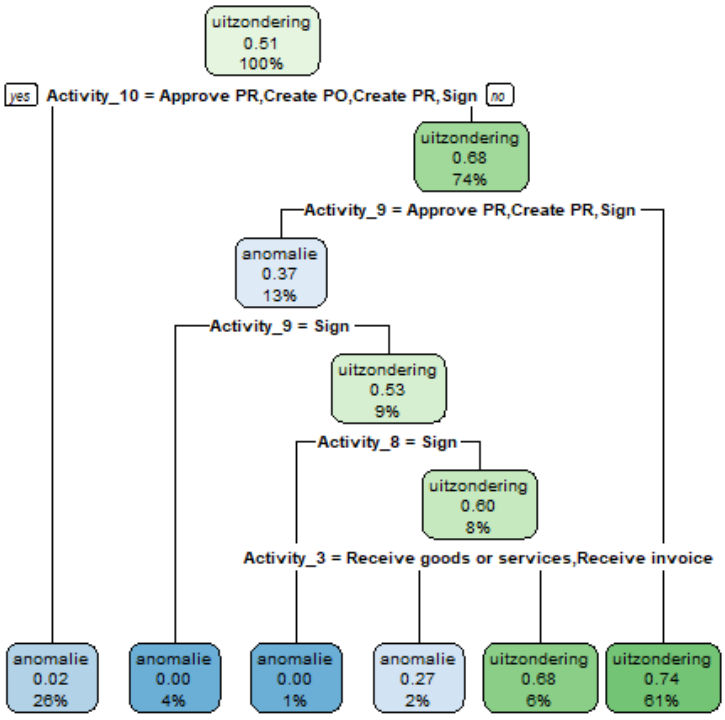
| Case ID | Anomalie | Activiteit 1 | Activiteit 2 | Activiteit 3 | ... |
|---|---|---|---|---|---|
| 1 | Anomalie | Receive goods or services | Create PO | Sign | ... |
| 2 | Anomalie | Pay | Approve PR | Receive goods or services | ... |
| 3 | Anomalie | Create PR | Pay | Receive goods or services | ... |
| 4 | Uitzondering | Sign | Sign | Approve PR | ... |
| 5 | Uitzondering | Receive goods or services | Receive goods or services | Approve PR | ... |

Tabel 7: Voorbeeld van de data gebruikt voor de decision tree in R

### 4.4.2 Prestatie

De data werd verdeeld in 80 procent van de cases voor de training set en 20 procent voor de test set. Het gevonden model kan vervolgens gegenereerd worden met de training set en hierna gevalideerd worden met de test set en is te zien in figuur 2.

Het gevonden model behaalt een accuraatheid van 81,34 procent bij het toepassen van het model op de test set. Aan de hand van dit model kunnen er nu regels gemaakt worden door te kijken naar de decision nodes. Zo zien we bijvoorbeeld dat de eerste node splitst door te kijken welke activiteit als 10de plaatsvindt. Als deze 10de activiteit Approve PR, Create PO, Create PR of Sign is, kunnen we met 98 procent zekerheid zeggen dat deze case een anomalie is. Naast deze regel werden er nog 5 andere regels gevonden door de verschillende takken van de boom te volgen.




Figuur 2: Decision tree in R die voorspelt of een case een anomalie of een uitzondering is

| Anomalie | Anomalie/ Uitzondering | Regel |
|---|---|---|
| Anomalie | [1;0] | when Activity 10 is Pay or Receive goods or services or Receive invoice and Activity 9 is Sign |
| Anomalie | [1;0] | when Activity 10 is Pay or Receive goods or services or Receive invoice and Activity 9 is Approve PR or Create PR and Activity 8 is Sign |
| Anomalie | [.98;.02] | when Activity 10 is Approve PR or Create PO or Create PR or Sign |
| Anomalie | [.73;.27] | when Activity 10 is Pay or Receive goods or services or Receive invoice and Activity 9 is Approve PR or Create PR and Activity 8 is Approve PR or Create PR or Pay or Receive goods or services or Receive invoice and Activity 3 is Receive goods or services or Receive invoice |
| Uitzondering | [[.32;.68]] | when Activity 10 is Pay or Receive goods or services or Receive invoice and Activity 9 is Approve PR or Create PR and Activity 8 is Approve PR or Create PR or Pay or Receive goods or services or Receive invoice and Activity 3 is Approve PR or Create PO or Create PR or Pay or Sign |
| Uitzondering | [1;0] | when Activity 10 is Pay or Receive goods or services or Receive invoice and Activity 9 is Pay or Receive goods or services or Receive invoice |

Tabel 8: Regels uit de decision tree in R

## 4.5 Declarative process mining

### 4.5.1 Data preprocessing

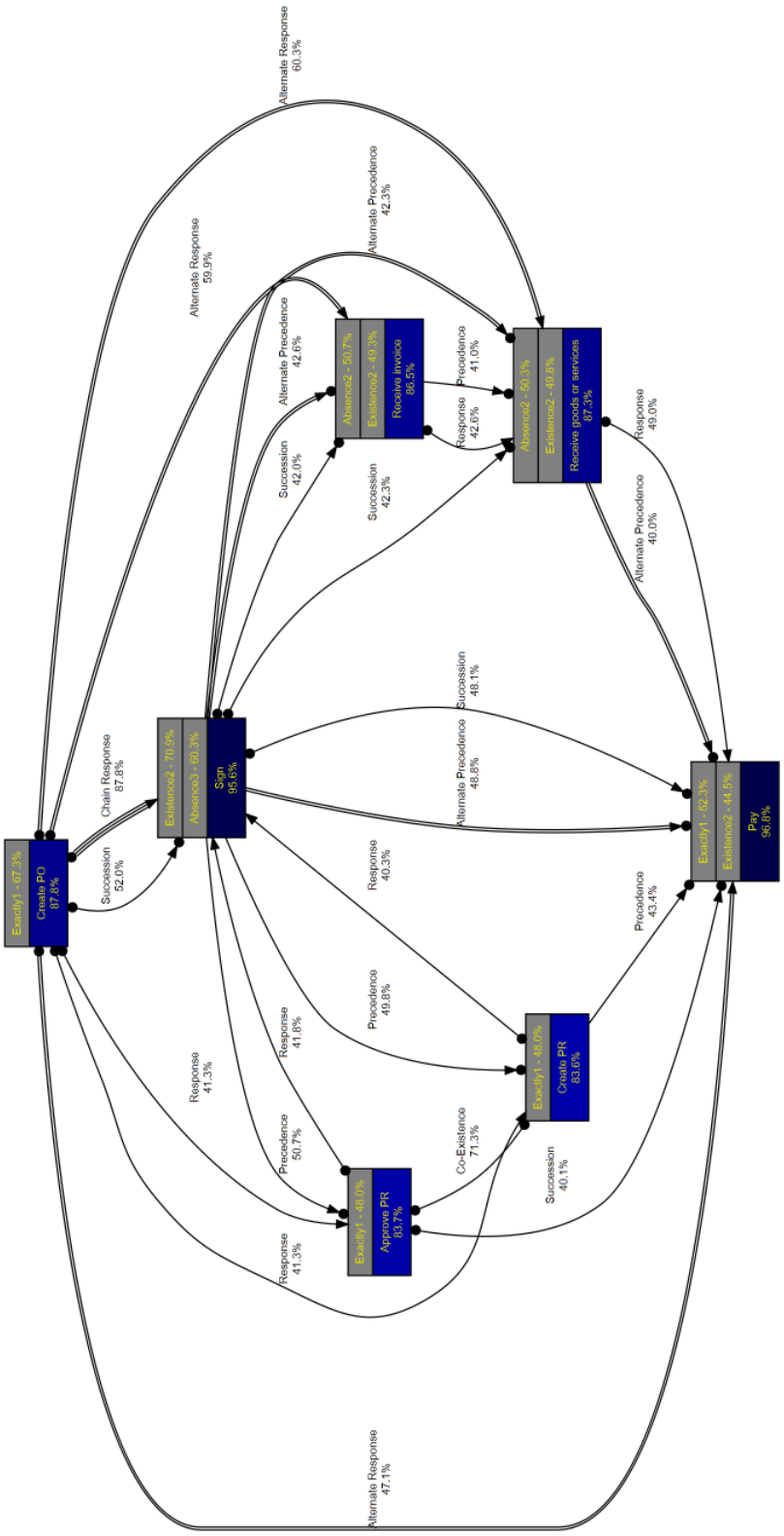
De laatste methode die getest zal worden is declarative process mining. Dit zal gedaan worden door gebruik te maken van de tool RuM. RuM is een gespecialiseerde tool voor het ontdekken en analyseren van declaratieve procesmodellen. RuM maakt gebruik van de declare miner om het model te genereren. De eerste dataset die getest werd is de combinatie van de twee datasets. Dit model in figuur 3 werd gecreëerd met een minimum support van 40 procent.

### 4.5.2 Prestatie

Hieruit kunnen we dan de beperkingen halen die gebruikt worden om het model te maken. Deze beperkingen vormen de regels die ons meer kunnen vertellen over het proces. Dit model levert ons 35 beperkingen op die een minimum support van 40 procent hebben. Als we de minimum support verlagen naar 20 procent zodat het een gelijke minimum support heeft als in vorige testen zien we dat er 84 beperkingen zijn. Deze beperkingen hebben een gemiddelde support van 30 procent.

| Beperkingen: |
|---|
| In 67.30% of traces in the log, Create PO occurs exactly once |
| In 52.30% of traces in the log, Pay occurs exactly once |
| In 48.80% of traces in the log, Each time Pay occurs, it is preceded by Sign and no other Pay can recur in between |
| In 47.95% of traces in the log, Approve PR occurs exactly once |
| In 47.95% of traces in the log, Create PR occurs exactly once |
| In 42.65% of traces in the log, Each time Receive invoice occurs, it is preceded by Sign and no other Receive invoice can recur in between |
| In 42.25% of traces in the log, Each time Receive goods or services occurs, it is preceded by Sign and no other Receive goods or services can recur in between |
| In 40.15% of traces in the log, Approve PR occurs if and only if it is followed by Pay |
| In 40.00% of traces in the log, Each time Pay occurs, it is preceded by Receive goods or services and no other Pay can recur in between |
| In 39.75% of traces in the log, Sign occurs at least three times |

Tabel 9: Set van beperkingen waarop het declaratief model gebaseerd is




Figuur 3: Declaratief model van de combinatie van de twee datasets met een minimum support van 40%

Ook de individuele datasets zullen geanalyseerd worden in RuM om te kijken of de gevonden beperkingen in de combinatie ook hier terugkomen. De eerste dataset is de data met 1000 cases die anomalieën zijn in het proces. Als we ook hier een minimum support van 20 procent toepassen krijgen we een model met 68 beperkingen met een gemiddelde support van 32 procent. Van de 68 beperkingen komen er 48 oftewel 71 procent terug in de beperkingen van de combinatie. Ook als we kijken naar de dataset met 1000 cases die een uitzondering zijn in het proces, zien we dat er 75 beperkingen gegenereerd worden met een gemiddelde support van 31 procent. Hiervan komen er 47 beperkingen oftewel 63 procent ook terug in de combinatie van de twee. In totaal slaagt de combinatie er dus in om 66 procent van de uitzonderingen gevonden in de individuele testen ook te vinden.

| | Aantal uitzonderingen | Gemiddelde support | Gevonden door de combinatie |
|---|---|---|---|
| Combinatie | 84 | 30% | 66% |
| Anomalieën | 68 | 32% | 71% |
| Uitzonderingen | 75 | 31% | 63% |

Tabel 10: Prestatie declarative process mining op de verschillende datasets

# 5 Discussie

## 5.1 Prestatie

Nu alle methodes getest zijn, zal hun geschiktheid voor het analyseren van procesafwijkingen in dit onderdeel besproken worden. Zo zien we dat bij het gebruiken van association rule mining een overvloed aan regels gegenereerd wordt. Zo worden uit de 2000 cases geanalyseerd in de combinatie van de twee datasets 1763 regels gehaald alsook 1721 en 1817 regels bij het analyseren van de individuele datasets. De grote hoeveelheid regels maakt het moeilijk om de regels te interpreteren en implementeren. Er kan wel gekeken worden naar regels met een zeer hoge support en confidentie die samen de belangrijkste regels vormen. Ook zagen we dat de combinatie van de twee datasets een heel andere reeks aan regels vond dan de regels gevonden bij het analyseren van de individuele datasets. Zo kwamen slechts 8 procent van de regels gevonden bij de individuele datasets ook terug in de combinatie van de twee. Echter viel wel op dat de combinatie 68 procent van de belangrijkste regels wel kon vinden. Hierdoor is het combineren van de data meer geschikt voor het vinden van de belangrijkste regels, maar kan de combinatie niet vertrouwd worden om alle regels te vinden en is het dus ook noodzakelijk om te kijken naar de individuele datasets om alle regels te verkrijgen. Hiernaast zijn de gevonden regels ook van een lagere kwaliteit omdat de gevonden regels enkel aangeven dat bepaalde activiteiten samen voorkomen in de vorm van een 'als-dan' regel. Een limitatie van ARM is ook dat deze regels niks over de volgorde waarin de activiteiten plaatsvinden zeggen.

De regels gevonden door sequential rule mining vertellen ons al iets meer over het proces. Deze regels houden namelijk wel rekening met de volgorde van de activiteiten. Ook houden deze regels rekening met activiteiten die kort na elkaar gebeuren en vormen ze connecties tussen deze activiteiten die aantonen dat deze activiteiten samen horen. Met sequential rule mining werden hier 247 sequenties gegenereerd bij de combinatie van de twee datasets. Ook bij de individuele testen vond het algoritme 191 sequenties voor de cases die een anomalie zijn en 390 sequenties voor de cases die een uitzondering zijn. Van deze sequenties kwamen 54 procent ook terug in de combinatie van de twee datasets. Dit toont aan dat sequential rule mining al beter dan association rule mining in staat is om alle regels terug te vinden in de combinatie, maar zeker nog niet alle. De lagere hoeveelheid regels maakt de analyse en implementatie van de regels wel eenvoudiger. Ook vertelt elke regel ons meer over het proces dan de regels gevonden bij association rule mining.

Clustering-based rule mining is een techniek die ook gebruik maakt van association rule mining, maar de data eerst clustert voor association rule mining toegepast wordt. Zo zien we dat in deze studie drie clusters het beste resultaat opleverde. Deze drie clusters genereerden elke hun regels en vonden respectievelijk 371, 602 en 1932 regels waarvan in totaal 1932 uniek waren. Hiernaast zagen we bij de tweede en derde cluster een zeer grote gemiddelde support en betrouwbaarheid van boven de 75 procent. Dit toont aan dat de gevonden regels zeer relevant zijn binnen hun cluster. Als we deze regels vergelijken met de regels gevonden bij het enkel toepassen van association rule mining zien we dat 77 procent van de regels niet gevonden werden door enkel association rule mining toe te passen. Hieruit kunnen we concluderen dat door eerst te clusteren er nieuwe regels gevonden worden die ook relevant kunnen zijn voor het proces. Deze resultaten komen ook overeen met onze bevindingen bij association rule mining, waar we kijken naar de individuele datasets. Dit is ook een soort clustering dat toegepast wordt en toont ook aan dat door de data te splitsen nieuwe regels gevonden kunnen worden. Echter zijn de nadelen van het gebruiken van association rule mining ook hier aanwezig. Zo genereerde deze methode nog meer regels dan enkel association rule mining en zeggen de regels ons enkel dat deze activiteiten samen voorkomen.

De vierde techniek die getest werd, is decision tree rule mining. Dit bleek een techniek te zijn die beter te implementeren is in R in plaats van Python wat gebruikt werd om de vorige methodes te testen. Tijdens deze studie werd er aan de hand van de gecombineerde data geprobeerd te voorspellen of een case een anomalie of een uitzondering was. Hierin slaagde het gevonden model met een accuraatheid van 81.34 procent. Dit model kan dus gebruikt worden om toekomstige afwijking automatisch te classificeren waardoor er geen domeinexpert meer nodig is om het onderscheid tussen de twee te maken. Hiernaast vormen de decision nodes van de boom ook regels die ons meer kunnen vertellen over het proces. Zo ontstonden er 6 regels uit het model die bepaalden of een case een anomalie of een uitzondering is en de procentuele kans dat een case volgens die regel tot één van de twee groepen behoorde. Deze regels zijn dus veel

waardevoller dan regels gevonden bij de eerdere methodes en zijn ook beperkter in hoeveelheid wat het analyseren en implementeren van de regels eenvoudiger maakt.

Decalarative process mining is de laatste techniek getest in dit onderzoek. Deze techniek maakte gebruik van RuM om een declaratief model te creëren. Dit model werd opgesteld aan de hand van beperkingen die gezien kunnen worden als regels. Voor de combinatie van de twee datasets ontstonden er 82 beperkingen. Ook bij de datasets voor de anomalieën en de uitzonderingen ontstonden er respectievelijk 67 en 76 beperkingen. Hiervan kwam 66 procent ook terug in de combinatie van de twee. Dit toont aan dat de combinatie de meerderheid van de regels kon terugvinden en dus een goede representatie is van de twee individuele datasets. Hiernaast zijn de gevonden beperkingen ook veel gedetailleerder. Zo vertellen deze beperkingen ons meer dan de regels gevonden door alle andere methodes. Dit komt door de gedetailleerde beschrijving van de beperkingen en de verschillende soorten beperkingen die gebruikt worden om het model te beschrijven. Deze methode geeft ons dus het meeste inzicht in de werking van het proces. Ook zijn de regels beperkt in hoeveelheid wat het implementeren en interpreteren van de regels ook makkelijker maakt.

Als we de vijf technieken vergelijken met elkaar zien we dat er een groot verschil is in de prestatie van elke techniek. Zo zien we dat het gebruiken van ARM zorgt voor een grote hoeveelheid regels. Dit is te zien aan de 1763 regels gevonden bij ARM, maar ook bij de 1932 regels gevonden bij clustering-based rule mining waar er ook gebruik gemaakt is van ARM. Deze grote hoeveelheid regels maakt de manuele implementatie en interpretatie zeer tijdrovend. De andere technieken doen het op dit vlak veel beter met nog 247 regels bij sequential rule mining, 84 regels bij declarative process mining en slechts 6 regels bij decision tree rule mining. Deze technieken scoren beter in de hoeveelheid regels die ze genereren, maar er valt ook op dat bij sequential rule mining en declarative process mining de gemiddelde support van de regels een stuk lager ligt dan bij de andere technieken. Dit betekent dat de regels die ze gevonden hebben dus minder relevant zijn omdat deze minder vaak voorkomen. Bij sequental rule mining is dit wel deels te verklaren door te werken met een minimum support van 5 procent. Het gebruiken van ARM zorgt misschien voor een grote hoeveelheid regels maar deze regels zijn wel een stuk relevanter met een gemiddelde support van 60 procent en van 76 procent bij clustering-based rule mining. Tot slot is er nog gekeken of de verschillende technieken in staat zijn om regels te halen uit een ongelabelde dataset. Hier zien we dat ARM op dit vlak zeer zwak scoort met slechts 8 procent van de regels uit de individuele datasets die het algoritme kan terugvinden in de gecombineerde dataset. Dit toont aan dat ARM niet geschikt is voor het werken met dit soort ongelabelde data en het niet in staat is alle regels te vinden. Zowel sequential rule mining als declarative process mining scoren hier beter op met 54 procent en 66 procent respectievelijk. Hier kan geconcludeerd worden dat deze algoritmes dus al meer in staat zijn te werken met ongelabelde data en meer dan de helft van de regels terug kunnen vinden. Om dit te verbeteren kan er gebruik gemaakt worden van clustering-based rule mining die door eerst

te clusteren 77 procent nieuwe regels vindt in vergelijking met enkel ARM toe te passen. Dit is in deze studie enkel getest in combinatie met ARM, maar is ook mogelijk met andere rule mining technieken. Ook decision tree rule mining kan helpen in het labelen van de data. Het gevonden model kan 81 procent van de cases correct labelen. Deze gelabelde data kan dan gebruikt worden in combinatie met een andere rule mining techniek om gerichtere regels te vinden.

| | Aantal regels | Gemiddelde support | Regels die terugkomen in de individuele datasets |
|---|---|---|---|
| Association rule mining | 1763 | 60% | 8% |
| Sequential rule mining | 247 | 12% | 54% |
| Clustering-based rule mining | 1932 | 76% | / |
| Decision tree rule mining | 6 | / | / |
| Declarative process mining | 84 | 30% | 66% |

Tabel 11: Prestatie rule mining technieken op de gecombineerde dataset

## 5.2 Implicaties

Deze studie biedt voor het eerst een vergelijking aan van verschillende rule mining technieken. Bij eerder onderzoek lag de focus steeds op het vergelijken van de verschillende algoritmes binnen een bepaalde techniek. De bevindingen van dit onderzoek tonen voor het eerst aan hoe de prestaties van rule mining technieken verschillen bij het toepassen van deze technieken op dezelfde data. De resultaten bieden een duidelijk overzicht aan van de verschillende rule mining technieken bij het analyseren van procesafwijkingen. Deze studie draagt bij aan het beter begrijpen van de rule mining technieken wat essentieel is voor het bepalen van de geschikte techniek in toekomstig onderzoek.

## 5.3 Limitaties

Deze studie heeft ook enkele limitaties. Ten eerste is de gebruikte data een artificiële dataset gebaseerd op een procure-to-pay proces. Ook al is dit een realistische representatie van dit proces, moet toekomstig onderzoek ook aantonen dat deze resultaten ook vergelijkbaar zijn met resultaten uit andere data. Ten tweede is in deze studie enkel gewerkt met procesafwijkingen. Onderzoek naar de prestatie van de verschillende rule mining technieken op andere soorten data is zeker waardevol. Ten derde is er voor elke techniek slechts gebruik gemaakt van één techniek. Er bestaan al studies die de verschillende algoritmes binnen elke techniek vergelijken, maar toekomstig onderzoek kan dit uitbreiden naar het testen van verschillende technieken en meerdere algoritmes per techniek op dezelfde data.

# 6 Conclusie

Deze studie toont voor het eerst de verschillen tussen de rule mining technieken aan. Dit is gedaan door de prestatie van de verschillende technieken op dezelfde data te vergelijken. Deze testen werden uitgevoerd op een realistische maar artificiële dataset met event logs van procesafwijkingen. De resultaten tonen aan dat het gebruiken van association rule mining zorgt voor zeer veel regels, de regels gevonden door sequential rule mining en declarative process mining een lage support hebben en dat sequential rule mining en declarative process mining beter in staat zijn te werken met ongelabelde data dan association rule mining. Hiernaast zien we dat clustering-based rule mining voor 77 procent nieuwe regels oplevert en dat decision tree rule mining er in slaagt om een model te creëren dat kan voorspellen of een case een anomalie of een uitzondering is met een accuraatheid van 81 procent.

# 7 Bronnenlijst


Abdelwahab, A., Link to external site, this link will open in a new window, & Youssef, N. (2022). Performance Evaluation of Sequential Rule Mining Algorithms. Applied Sciences, 12(10), 5230. https://doi.org/10.3390/app12105230

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. Proceedings of the 1993 ACM SIGMOD international conference on Management of data, 207-216. https://doi.org/ 10.1145/170035.170072

Agrawal, R., & Srikant, R. (1995). Mining sequential patterns. Proceedings of the Eleventh International Conference on Data Engineering, 3-14. https: //doi.org/10.1109/ICDE.1995.380415

Ardimento, P., Aversano, L., Bernardi, M. L., & Cimitile, M. (2020). Data-Aware Declarative Process Mining for Malware Detection. 2020 International Joint Conference on Neural Networks (IJCNN), 1-8. https://doi.org/10.1109/ IJCNN48605.2020.9206902

Boutorh, A., & Guessoum, A. (2016). Complex diseases SNP selection and classification by hybrid Association Rule Mining and Artificial Neural Network—Based Evolutionary Algorithms. Engineering Applications of Artificial Intelligence, 51, 58-70. https://doi.org/10.1016/ j.engappai.2016.01.004

Breiman, L. (2017). Classification and Regression Trees. Routledge. https://doi.org/ 10.1201/9781315139470

Carmona, J., van Dongen, B., Solti, A., & Weidlich, M. (2018). Conformance Checking: Relating Processes and Models. Springer International Publishing. https://doi.org/10.1007/978-3-319-99414-7

Chen, M.-C., & Wu, H.-P. (2005). An association-based clustering approach to order batching considering customer demand patterns. Omega, 33(4), 333-343. https://doi.org/10.1016/j.omega.2004.05.003

Elacio, A., Balazon, F., & Lacatan, L. (2020). Digital Transformation in Managing Employee Retention using Agile and C4.5 Algorithm.

Gama, J., Fernandes, R., & Rocha, R. (2006). Decision trees for mining data streams. Intelligent Data Analysis, 10(1), 23-45. https://doi.org/10.3233/IDA-2006-10103

Goedertier, S., Vanthienen, J., & Caron, F. (2015). Declarative business process modelling: Principles and modelling languages. Enterprise Information Systems, 9(2), 161-185. https://doi.org/10.1080/17517575.2013.830340

Husák, M., Bajtoš, T., Kašpar, J., Bou-Harb, E., & Čeleda, P. (2020). Predictive Cyber Situational Awareness and Personalized Blacklisting: A Sequential Rule Mining Approach. ACM Transactions on Management Information Systems, 11(4), 19:1-19:16. https://doi.org/10.1145/3386250

Imai, S., Yamada, T., Kasashi, K., Ishiguro, N., Kobayashi, M., & Iseki, K. (2019). Construction of a flow chart–like risk prediction model of ganciclovir-induced neutropaenia including severity grade: A data mining approach using decision tree. Journal of Clinical Pharmacy and Therapeutics, 44(5), 726-734. https://doi.org/10.1111/jcpt.12852

Jain, J. K., Tiwari, N., & Ramaiya, M. (2013). A Survey: On Association Rule Mining. International Journal of Engineering, 3(1).

Jeihouni, M., Toomanian, A., & Mansourian, A. (2020). Decision Tree-Based Data Mining and Rule Induction for Identifying High Quality Groundwater Zones to Water Supply Management: A Novel Hybrid Use of Data Mining and GIS. Water Resources Management, 34(1), 139-154. https://doi.org/10.1007/ s11269-019-02447-w

Karthikeyan, T., & Ravikumar, N. (2014). A Survey on Association Rule Mining. 3(1).

Kass, G. V. (1980). An Exploratory Technique for Investigating Large Quantities of Categorical Data.

Khan, S., & Parkinson, S. (2018). Eliciting and utilising knowledge for security event log analysis: An association rule mining and automated planning approach. Expert Systems with

Applications, 113, 116-127. https://doi.org/ 10.1016/j.eswa.2018.07.006

Khan, S., & Parkinson, S. (2019). Discovering and utilising expert knowledge from security event logs. Journal of Information Security and Applications, 48, 102375. https://doi.org/10.1016/j.jisa.2019.102375

Laghmouch, M., Jans, M., & Depaire, B. (2020). Classifying process deviations with weak supervision. 2020 2nd International Conference on Process Mining (ICPM), 89-96. https://doi.org/10.1109/ICPM49681.2020.00023

Mabroukeh, N. R., & Ezeife, C. I. (2010). A taxonomy of sequential pattern mining algorithms. ACM Computing Surveys, 43(1), 3:1-3:41. https://doi.org/ 10.1145/1824795.1824798

Mertens, S., Gailly, F., Van Sassenbroeck, D., & Poels, G. (2022). Integrated Declarative Process and Decision Discovery of the Emergency Care Process. Information Systems Frontiers, 24(1), 305-327. https://doi.org/10.1007/s10796-020-10078-5

Mirebrahim, S. H., Shokoohi-Yekta, M., Kurup, U., Welfonder, T., & Shah, M. (2017). A clustering-based rule-mining approach for monitoring long-term energy use and understanding system behavior. Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments, 1-9. https://doi.org/10.1145/3137133.3137144

Öztayşi, B., Yurdadön, P., & Onar, S. Ç. (2022). Fuzzy Clustering Based Association Rule Mining: A Case Study on Ecommerce. In C. Kahraman, A. C. Tolga, S. Cevik Onar, S. Cebi, B. Oztaysi, & I. U. Sari (Red.), Intelligent and Fuzzy Systems (pp. 112-118). Springer International Publishing. https://doi.org/10.1007/ 978-3-031-09173-5-15

Polyvyanyy, A., Wynn, M. T., Van Looy, A., & Reichert, M. (Red.). (2021). Business Process Management: 19th International Conference, BPM 2021, Rome, Italy, September 06–10, 2021, Proceedings (Vol. 12875). Springer International Publishing. https://doi.org/10.1007/978-3-030-85469-0

Prithiviraj, P., & Porkodi, R. (2015). A Comparative Analysis of Association Rule Mining Algorithms in Data Mining: A Study.

Quinlan, J. R. (1993). C4.5: Programs for Machine Learning. Morgan Kaufmann.

Riaz, M., Arooj, A., Hassan, M. T., & Kim, J.-B. (2014). Clustering based association rule mining on online stores for optimized cross product recommendation. The 2014 International Conference on Control, Automation and Information Sciences (ICCAIS 2014), 176-181.

https://doi.org/10.1109/ICCAIS.2014.7020553

Rovani, M., Maggi, F. M., de Leoni, M., & van der Aalst, W. M. P. (2015). Declarative process mining in healthcare. Expert Systems with Applications, 42(23), 9236-9251. https://doi.org/ 10.1016/j.eswa.2015.07.040

Rozinat, A., & van der Aalst, W. M. P. (2008). Conformance checking of processes based on monitoring real behavior. Information Systems, 33(1), 64-95. https://doi.org/10.1016/j.is.2007.07.001

Setiawan, F., & Yahya, B. N. (2018). Improved behavior model based on sequential rule mining. Applied Soft Computing, 68, 944-960. https://doi.org/ 10.1016/j.asoc.2018.01.035

Shrivastava, K. K., Mishra, R., & Dubey, S. M. (2011). Analysis of System Error Log Using Association Mining. 39-44. https://www.seekdl.org/conferences/ paper/details/2199.html

Tayefi, M., Tajfard, M., Saffar, S., Hanachi, P., Amirabadizadeh, A. R., Esmaeily, H., Taghipour, A., Ferns, G. A., Moohebati, M., & Ghayour-Mobarhan, M. (2017). hs-CRP is strongly associated with coronary heart disease (CHD): A data mining approach using decision tree algorithm. Computer Methods and Programs in Biomedicine, 141, 105-109. https://doi.org/10.1016/j.cmpb.2017.02.001

van der Aalst, W. M. P. (2016). Process Mining: Data Science in Action. Springer Berlin / Heidelberg. http://ebookcentral.proquest.com/lib/ubhasselt/ detail.action?docID=4505537