UNIVERZITA PAVLA JOZEFA ŠAFÁRIKA V KOŠICIACH

PRÍRODOVEDECKÁ FAKULTA

IDENTIFIKÁCIA PODOZRIVÝCH FORENZNÝCH ARTEFAKTOV

2023 **Boris HAMADEJ**

UNIVERZITA PAVLA JOZEFA ŠAFÁRIKA V KOŠICIACH

PRÍRODOVEDECKÁ FAKULTA

IDENTIFIKÁCIA PODOZRIVÝCH FORENZNÝCH ARTEFAKTOV

BAKALÁRSKA PRÁCA

Študijný program:	Aplikovaná informatika
Pracovisko (katedra/ústav):	Ústav informatiky
Vedúci bakalárskej práce:	Mgr. Eva Marková
Konzultant bakalárskej práce:	doc. RNDr. JUDr. Pavol Sokol, PhD.

Košice 2023 **Boris HAMADEJ**

Univerzita P. J. Šafárika v Košiciach
Prírodovedecká fakulta

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Boris Hamadej
Študijný program: aplikovaná informatika (jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: Informatika
Typ záverečnej práce: Bakalárska práca
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Identifikácia podozrivých forenzných artefaktov

Názov EN: Identification of suspicious forensic artifacts

Cieľ: (1) Analýza forenzných artefaktov vo vybranom operačnom systéme
(2) Porovnanie existujúcich prístupov k identifikácii anomálií pri forenznom vyšetrovaní
(3) Návrh nástroja pre identifikáciu podozrivých forenzných artefaktov vo vybranom operačnom systéme, otestovanie nástroja a zhodnotenie výsledkov

Literatúra: (1) Baddar, S. A. H., Merlo, A., & Migliardi, M. (2019). Behavioral-anomaly detection in forensics analysis. IEEE Security & Privacy, 17(1), 55-62.
(2) Pang, G., Shen, C., Cao, L., & Hengel, A. V. D. (2021). Deep learning for anomaly detection: A review. ACM Computing Surveys (CSUR), 54(2), 1-38.
(3) Pourhabibi, T., Ong, K. L., Kam, B. H., & Boo, Y. L. (2020). Fraud detection: A systematic literature review of graph-based anomaly detection approaches. Decision Support Systems, 133, 113303.

Vedúci: Mgr. Eva Marková
Konzultant: doc. RNDr. JUDr. Pavol Sokol, PhD.
Oponent: RNDr. Tomáš Bajtoš
Ústav : ÚINF - Ústav informatiky
Riaditeľ ústavu: doc. RNDr. Ondrej Krídlo, PhD.
Spôsob sprístupnenia elektronickej verzie práce: bez obmedzenia

Dátum schválenia: 15.05.2023

Poďakovanie

Týmto sa chcem poďakovať vedúcemu svojej práce Mgr. Eve Markovej a konzultantovi doc. RNDr. JUDr. Pavlovi Sokolovi, PhD. za odborné vedenie, cenné rady a veľkú pomoc počas tvorby práce.

Abstrakt v štátnom jazyku

Digitálna forenzná analýza sa stala nevyhnutnou súčasťou reakcie na počítačové bezpečnostné incidenty ako aj súčasťou vyšetrovania kybernetickej kriminality. Dôležitými krokmi forenzného vyšetrovania sú identifikácia digitálnych stôp potenciálnych útočníkov, ich zber, analýza a ich následné zdokumentovanie. V našej práci sa venujeme metódam a postupom na čo najpresnejšie identifikovanie podozrivých forenzných artefaktov v operačnom systéme Windows a ich efektívnemu využitiu pri analýze a detekcii anomálií. Ako náš modelový prípad používame „Prípad ukradnutej sečuánskej omáčky“ z portálu DFIR Madness. Tieto dáta sa v predošlom výskume predspracovali a na tomto upravenom datasete sme otestovali niekoľko existujúcich metód na detekciu anomálií bez učiteľa, ako napríklad ECOD, IForest či PCA. Analyzovali sme výsledky a úspešnosť jednotlivých metód pri detekcii anomálií, čím sme získali lepší prehľad o možnostiach ich uplatnenia pri digitálnej forenznej analýze. Na základe našej analýzy sme vybrali najlepšie metódy a implementovali ich do jednoduchého nástroja, ktorý užívateľom poskytne možnosť vybrať si metódy, ktoré chcú použiť. Tento nástroj následne porovnáva čas behu jednotlivých metód a ich výsledky, čo užívateľom umožní lepšie porozumieť výhodám a nevýhodám jednotlivých metód a vybrať si z nich tie najvhodnejšie pre ich konkrétny prípad.

Kľúčové slová: detekcia anomálií, digitálna forenzná analýza

Abstrakt v cudzom jazyku

Digital forensics has become an essential part of computer security incident response as well as cybercrime investigations. Important steps of a forensic investigation are the identification of digital traces of potential attackers, their collection, analysis, and their subsequent documentation. In our work, we focus on methods and procedures for the most accurate identification of suspicious forensic artifacts in the Windows operating system and their effective use in the analysis and detection of anomalies. We use "The Case of the Stolen Szechuan Sauce" from DFIR Madness as our model case. These data were preprocessed in the previous research. We tested several existing unsupervised methods for detecting anomalies, such as ECOD, IForest or PCA, on the edited dataset. We analyzed the results and success of individual methods in detecting anomalies, which gave us a better overview of the possibilities of their application in digital forensic analysis. Based on our analysis, we have selected the best methods and implemented them into a simple tool that will give users the opportunity to choose the methods they want to use. This tool then compares the running time of individual methods and their results, which will allow users to better understand the advantages and disadvantages of individual methods and choose the most suitable one for their case.

Keywords: anomaly detection, digital forensics

Obsah

Obsah **6**

Zoznam ilustrácií **8**

Zoznam skratiek a značiek **9**

Úvod **10**

1 Digitálna forenzná analýza **12**

1.1 Proces digitálnej forenznej analýzy 12

1.1.1 Identifikácia 12

1.1.2 Zber a uchovanie dát 12

1.1.3 Analýza 13

1.1.4 Dokumentácia 13

1.2 Techniky digitálnej forenznej analýzy 13

1.3 Forenzný artefakt 14

2 Anomália **17**

2.1 Typy anomálií 18

2.2 Súvisiace práce 18

2.3 Metódy detekcie anomálií 19

2.3.1 Štatistické metódy 20

2.3.2 Metódy založené na blízkosti 20

2.3.3 Súbory outlierov (outlier ensembles) 21

2.3.4 Lineárne modely 22

2.3.5 Neurónové siete 22

3 Dataset a metodológia **23**

3.1 Prípad ... 23
3.2 Dataset ... 24
3.3 Porovnanie otestovaných metód ... 25
3.4 Vybrané metódy ... 28
3.4.1 Empirical-Cumulative-distribution-based Outlier Detection (ECOD) ... 28
3.4.2 Isolation Forest (IForest) ... 28
3.4.3 Principal Component Analysis (PCA) ... 29
3.4.4 One-Class Support Vector Machine (OCSVM) ... 30
4 Návrh nástroja a vyhodnotenie ... 32
4.1 Nástroj ... 32
4.2 Výsledky ... 33
Záver ... 40
Zoznam použitej literatúry ... 42
Prílohy ... 46

Zoznam ilustrácií

Obr. 1 Príklad anomálií v 2D datasete [12] 17

Obr. 2 Rozdelenie otestovaných metód ADBench [15] 19

Obr. 3 Proces ensemble-based metód [31] 21

Obr. 4 Rozdelenie anomálie a bežného údajového bodu [38] 29

Obr. 5 Detekcia anomálií pomocou PCA [39] 30

Obr. 6 Rozdelenie dvoch tried pomocou OCSVM [40] 31

Obr. 7 Ukážka menu 32

Obr. 8 Schéma boxplotu [41] 33

Obr. 9 Doba trvania metód 34

Obr. 10 Graf počtu správne detegovaných anomálií 35

Obr. 11 Boxplot rozpätia správne detegovaných anomálií 35

Obr. 12 Heatmapa metódy PCA 36

Obr. 13 Heatmapa metódy ECOD 36

Obr. 14 Heatmapa metódy IForest 37

Obr. 15 Heatmapa metódy OCSVM 37

Zoznam skratiek a značiek

SVM **S**upport Vector Machines, metódy podporných vektorov

IoT Internet of Things, internet vecí

ID Identity Document, identifikačné číslo

PyOD Python Outlier Detection, Python detekcia anomálií

HBOS Histogram-based Outlier Score, anomálne skóre založené na histograme

ECOD Empirical-Cumulative-distribution-based Outlier Detection, detekcia anomálií založená na empiricko-kumulatívnej distribúcii

COPOD Copula-Based Outlier Detection, detekcia anomálií založená na kopule

CBLOF Clustering-Based Local Outlier Factor, lokálny odľahlý faktor založený na klastrovaní

kNN k Nearest Neighbors, k najbližší susedia

SOD Subspace Outlier Detection, podpriestorová detekcia anomálií

ROD Rotation-based Outlier Detection, detekcia anomálií založená na rotácii

LOF Local Outlier Factor, lokálny odľahlý faktor

iNNE Clustering-Based Local Outlier Factor, lokálny odľahlý faktor založený na klastrovaní

Loda Lightweight On-line Detector of Anomalies, ľahký priamy detektor anomálií

PCA Principal Component Analysis, analýza hlavných komponentov

OCSVM One-Class Support Vector Machines, metódy podporných vektorov jednou triedou

SVDD Support Vector Data Description, metóda popisu údajov podpornými vektormi

DFIR Digital Forensics and Incident Response, digitálna forenzná analýza a reakcia na incidenty

Úvod

Digitálna forenzná analýza je kľúčovým nástrojom v boji proti kybernetickej kriminalite a zabezpečeniu ochrany údajov v digitálnom svete. S narastajúcim objemom údajov a zložitosťou súčasných informačných systémov sa manuálna analýza stáva časovo náročnou a zložitou úlohou, ktorá môže vyžadovať veľké množstvo odborných znalostí a zdrojov. V tejto súvislosti sa detekcia anomálií javí ako efektívny a rýchlejší prístup, ktorý by mohol pomôcť pri identifikácii podozrivých forenzných artefaktov.

Detekcia anomálií je proces identifikácie vzorov alebo pozorovaní v súbore údajov, ktoré nezodpovedajú očakávanému správaniu. Ide o kľúčovú úlohu v rôznych oblastiach, ako je finančníctvo, priemysel, zdravotníctvo a kybernetická bezpečnosť, kde identifikácia neobvyklých vzorov môže pomôcť odhaliť podvody, diagnostikovať choroby alebo identifikovať narušenia bezpečnosti. Vo financiách sa detekcia anomálií môže použiť na identifikáciu nezvyčajných transakcií kreditnými kartami, ktoré môžu naznačovať podvod. V zdravotníctve sa môže použiť na diagnostikovanie chorôb u pacientov. V kybernetickej bezpečnosti možno detekciu anomálií použiť na identifikáciu nezvyčajnej sieťovej prevádzky, ktorá môže naznačovať potenciálne narušenie bezpečnosti.

Existuje veľa metód na detekciu anomálií vrátane štatistických metód, lineárnych modelov, techník strojového učenia a neurónových sietí. Štatistické metódy na detekciu anomálií zahŕňajú použitie funkcií hustoty pravdepodobnosti a štatistických modelov. Tieto metódy sa opierajú o predpoklad, že údaje sledujú určité rozdelenie, napríklad normálové rozdelenie. Techniky strojového učenia na detekciu anomálií zahŕňajú použitie zhlukovacích algoritmov, ako sú k-means a zhlukovanie založené na hustote, a použitie klasifikačných algoritmov, ako metóda podporných vektorov (SVM). Každá metóda má svoje výhody a nevýhody a výber metódy závisí mimo iného od daného datasetu, jeho veľkosti, typu údajov a konkrétneho problému, ktorý je potrebné vyriešiť. V tejto práci poskytujeme prehľad o niekoľkých metódach detekcie anomálií. Porovnávame úspešnosť detekcie podozrivých artefaktov, kombinácie parametrov daných metód, ale aj čas behu. Napriek mnohým dostupným metódam sa stretávame s pár výzvami pri našom výskume:

- Nedostatok univerzálne použiteľných metód. Využívanie jednej metódy pre jednu oblasť neznamená jej funkčnosť aj v iných oblastiach.

- Dáta obsahujú šum, ktorý sa môže falošne tváriť ako anomália.
- Použiteľnosť detekčných metód v budúcnosti. Prispôsobenie útočníkov, aby ich útoky boli ťažšie detekovateľné.
- Nedostatok verejne dostupných roztriedených datasetov.

Kvôli týmto výzvam nie je ľahké vyriešiť problém detekcie anomálií vo svojej najvšeobecnejšej forme. V skutočnosti väčšina existujúcich metód detekcie anomálií rieši špecifickú formuláciu problému. Táto formulácia je vyvolaná rôznymi faktormi, ako je povaha údajov, dostupnosť označených údajov, typ anomálií, ktoré sa majú zistiť a podobne. Často sú tieto faktory určené aplikačnou doménou, v ktorej je potrebné anomálie odhaliť.

Táto práca je rozdelená do štyroch kapitol. V prvej kapitole našej práce sa venujeme opisu procesu digitálnej forenznej analýzy, rôznych techník, ktoré sa môžu na skúmanie dôkazov použiť. Ďalej vysvetľujeme, čo je to forenzný artefakt a aké artefakty poznáme v súborovom systéme operačného systému Windows. V druhej kapitole sa zameriavame na definíciu anomálií a ich typy. Opisujeme súvisiace práce, ktoré sa taktiež zameriavali na detekciu anomálií v rôznych odvetviach. Ďalej sa zameriavame na typy metód podľa toho, aké techniky používajú na detekciu. V tretej kapitole opisujeme náš modelový prípad, vybraný dataset a ako bol predspracovaný v predošlom výskume. Porovnávame 12 vybraných metód, z ich výsledkov vyberieme zopár, ktoré bližšie analyzujeme a porovnávame detailnejšie pomocou nášho nástroja. Následne bližšie skúmame nami vybrané metódy. Posledná kapitola je zameraná na nástroj, pomocou ktorého si užívateľ vyberie, aké metódy chce použiť na vybranom datasete. Následne mu tento nástroj vypíše výsledky vo forme rôznych grafov a štatistických metód na porovnanie. Pomocou tohto nástroja porovnáme metódy a z výsledkov vyvodzujeme konkrétne závery, ktorá metóda je v našom prípade najlepšia.

1 Digitálna forenzná analýza

Digitálna forenzná analýza je proces skúmania digitálnych stôp s cieľom odhaliť relevantné informácie, identifikovať postup páchateľa a zrekonštruovať udalosti súvisiace s vyšetrovaním. Tento proces zohráva kľúčovú úlohu pri porozumení kontextu a dôsledkov digitálnych stôp. Dôležitou súčasťou digitálnej forenznej analýzy je analýza podozrivých kybernetických útokov s cieľom identifikovať, zmierniť a odstrániť kybernetické hrozby. Vďaka tomu je digitálna forenzná analýza kritickou súčasťou procesu reakcie na incidenty. Digitálna forenzná analýza je užitočná aj v prípade následkov útoku na poskytovanie informácií požadovaných orgánmi činnými v trestnom konaní. Digitálne stopy možno zbierať z rôznych zdrojov vrátane počítačov, mobilných zariadení, zariadení internetu vecí (IoT) a prakticky akéhokoľvek iného počítačového systému [1].

1.1 Proces digitálnej forenznej analýzy

Proces digitálnej forenznej analýzy sa môže meniť od jedného prípadu k druhému, ale zvyčajne pozostáva z týchto základných krokov – identifikácia, zber a uchovanie dát, analýza a dokumentácia [1].

1.1.1 Identifikácia

Identifikácia je prvý krok vo forenznom procese. Vyšetrovateľ musí identifikovať, aké stopy sa nachádzajú na zariadení, kde sú uložené a v akom formáte sú uložené. Digitálne stopy môžu mať rôzne formáty (textové správy, e-maily, obrázky alebo videá, história vyhľadávania, dokumenty, transakcie atď.) a na rôznych zariadeniach vrátane počítačov, smartfónov, tabletov a ďalších. Forenzných vyšetrovateľov tiež obzvlášť zaujímajú artefakty zariadenia, napríklad údaje operačného systému, súbory registra a podobne [2].

1.1.2 Zber a uchovanie dát

Fáza zberu dát zahŕňa získavanie digitálnych stôp z rôznych zariadení. Potom, čo boli zariadenia zaistené a uložené na bezpečnom mieste, digitálny forenzný vyšetrovateľ

alebo forenzný analytik využije rôzne forenzné techniky na extrakciu akýchkoľvek údajov, ktoré môžu byť relevantné pre vyšetrovanie, a bezpečne ich uloží. Táto fáza môže zahŕňať vytvorenie digitálnej kópie príslušných údajov. Je dôležité zabezpečiť, aby sa údaje počas procesu zberu nestratili alebo nepoškodili. Strate údajov môžete zabrániť skopírovaním pamäťového média alebo vytvorením klonu disku [3].

1.1.3 Analýza

Keď sú príslušné zariadenia identifikované a izolované a údaje sú duplikované a bezpečne uložené, digitálni forenzní vyšetrovatelia použijú techniky na extrakciu relevantných údajov a ich preskúmanie, pričom hľadajú stopy, ktoré poukazujú na nelegálnu činnosť. To často zahŕňa napríklad obnovenie a preskúmanie odstránených, poškodených alebo zašifrovaných súborov [3].

1.1.4 Dokumentácia

Zistenia sú zdokumentované a prezentované jasným, stručným a objektívnym spôsobom na použitie v ďalšom vyšetrovaní. Cieľom je dostať tieto zistenia do formátu, ktorému budú rozumieť aj nezainteresovaní ľudia [4]. Správna dokumentácia pomáha formulovať časovú os činností, ktoré sú súčasťou protiprávneho konania [3].

1.2 Techniky digitálnej forenznej analýzy

Digitálna forenzná analýza využíva celý rad techník na skúmanie digitálnych stôp a získavanie zmysluplných poznatkov. Niektoré techniky zahŕňajú:

- **Analýza súborového systému** – Skúmanie štruktúry metadát, obsahu súborov a adresárov na digitálnom pamäťovom zariadení s cieľom identifikovať relevantné stopy, ako sú vymazané súbory alebo skryté údaje.
- **Analýza logov** – Prehľadávanie záznamov (logov) z rôznych zdrojov (napr. systémové a sieťové logy, logy aplikácií) na sledovanie aktivít používateľov, zisťovanie bezpečnostných incidentov alebo odhaľovanie dôkazov o manipulácii.

- **Analýza časovej osi** – Zostavenie chronologickej postupnosti udalostí na základe časových pečiatok z rôznych zdrojov, ako sú metadáta súborov alebo zápis v logoch, aby sme pochopili priebeh aktivít a identifikovali nezrovnalosti alebo anomálie [5].

1.3 Forenzný artefakt

V tejto podkapitole si bližšie vysvetlíme, čo je to forenzný artefakt a aké artefakty poznáme v operačnom systéme Windows. Forenzný artefakt môže byť stopa, ktorá vzniká v dôsledku ľudskej alebo systémovej činnosti. Tieto artefakty môžu byť fyzické alebo digitálne a môžu poskytnúť dôležité informácie pri forenzných vyšetrovaniach. Digitálne forenzné artefakty môžu zahŕňať súbory, e-maily, obrázky alebo iné typy údajov uložených v elektronických zariadeniach, pričom môžu pomôcť vyšetrovateľom pochopiť, čo sa stalo, kto bol zapojený a ako sa udalosti odohrali [6].

V tejto časti sa budeme zaoberať niektorými dôležitými forenznými artefaktmi vo Windowse, ktoré súvisia so súborovým systémom, ich funkciou a kde ich môžeme nájsť:

- **Thumbcache** – Funkcia, ktorá je v operačných systémoch Windows dostupná od Windows Vista. Používa sa na ukladanie miniatúr súborov do vyrovnávacej pamäte pre zobrazenie Windows Prieskumníka. Keď otvoríte Prieskumníka v zobrazení miniatúr, súbory v priečinku sa zobrazia ako malé obrázky, ktoré predstavujú obsah súborov. Tieto obrázky sú uložené v centralizovanom súbore vyrovnávacej pamäte miniatúr. Keď používateľ vymaže súbor, jeho miniatúra zostane v súbore vo vyrovnávacej pamäti. Analýza súboru ThumbCache poskytuje informácie, ako sú metadáta pôvodného súboru, jeho ID vyrovnávacej pamäte, kontrolný súčet hlavičky, posun údajov, typ údajov a veľkosť údajov [7].
  Lokácia:
  C:\Users\%USERNAME%\AppData\Local\Microsoft\Windows\Explorer

- **Kôš** – Kôš systému Windows obsahuje súbory, ktoré používateľ odstránil, ale ešte neboli vymazané zo systému. Aj keď používatelia môžu kôš

vyprázdniť pomerne ľahko, pre vyšetrovateľa je stále cenným zdrojom stôp. Keď je nejaký súbor vymazaný, v Koši sa vytvoria dva súbory. Prvý súbor začína hodnotou „$R“, za ktorou nasleduje náhodný reťazec – tento súbor obsahuje skutočný obsah vymazaného súboru. Druhý súbor začína „$I“ a končí rovnakým reťazcom ako súbor „$R“ – tento súbor obsahuje metadáta pre tento konkrétny súbor [8].

Lokácia:

C:\$Recycle.Bin\SID*\$Ixxxxxx

C:\$Recycle.Bin\SID*\$Rxxxxxx

- **OpenSaveMRU** – Tento kľúč obsahuje úplnú cestu k súboru, ku ktorému pristupovala ľubovoľná aplikácia prostredníctvom dialógového okna Otvoriť/Uložiť ako. Tento typ informácií je dôležitý počas procesu forenznej analýzy, pretože môže odhaliť podrobnosti týkajúce sa stiahnutých súborov a posledných súborov, ku ktorým mal používateľ prístup [9].

  Lokácia:

  NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePIDlMRU

- **Jump Lists** – Sú to funkcie systému Windows zavedené v systéme Windows 7. Obsahujú informácie o nedávno použitých aplikáciách a súboroch. Analýza Jump List súborov môže poskytnúť cenné informácie o aktivite používateľov v systéme, ako je vytváranie súborov, prístup a úpravy. Dajú sa vytvoriť dva typy týchto súborov „AutomaticDestinations” a „CustomDestinations” [10].

  Lokácia:

  C:\Users\%USERNAME%\AppData\Roaming\Microsoft\Windows\Recent\AutomaticDestinations*.automaticDestinations-ms

  C:\Users\%USERNAME%\AppData\Roaming\Microsoft\Windows\Recent\CustomDestinations*.customDestinations-ms

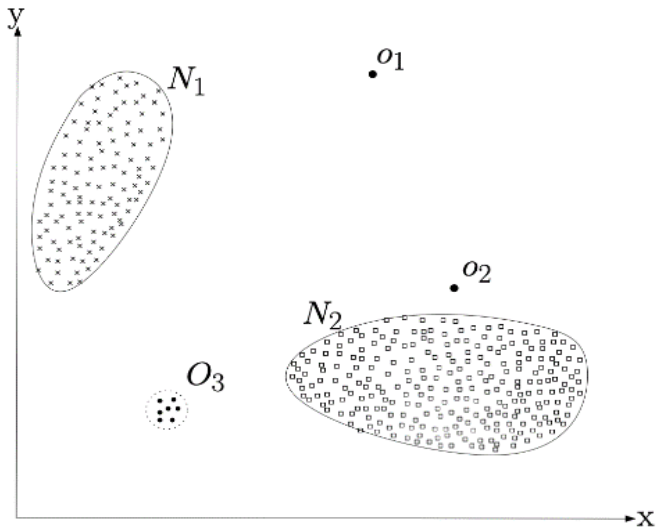
- **Prefetch súbory** – Tieto súbory urýchľujú načítanie konkrétneho zdroja aplikácie, čo umožňuje rýchlejšie otvárať najpoužívanejšiu aplikáciu. Predbežné načítanie umožňuje prehliadaču načítať zdroje potrebné na zobrazenie obsahu, ku ktorému bude možné pristupovať neskôr. Prefetch súbory prezradia, či jednotlivec nainštaloval a spustil konkrétny program [11].

  Lokácia:

  C:\Windows\Prefetch

2 Anomália

Anomália je odchýlka od očakávaného alebo normálneho správania. Môže sa vzťahovať na odchýlku od normy v akomkoľvek systéme alebo súbore údajov, napríklad v počítačovej sieti, finančných transakciách alebo dokonca v ľudskom správaní. Anomálie môžu naznačovať problémy, chyby alebo potenciálne škodlivé aktivity. Obrázok 1 znázorňuje anomálie v jednoduchom dvojrozmernom súbore údajov. Dáta majú dve normálové oblasti, N1 a N2, keďže väčšina pozorovaní leží v týchto dvoch oblastiach. Body, ktoré sú dostatočne vzdialené od týchto regiónov, napríklad body o1 a o2, a body v regióne O3, sú anomálie [12].



Obr. 1 Príklad anomálií v 2D datasete [12]

Anomálie v kontexte digitálnej forenznej analýzy označujú nezrovnalosti alebo odchýlky od očakávaných vzorcov, správania alebo údajov v digitálnom systéme. Tieto anomálie môžu byť indikátormi neoprávneného prístupu alebo škodlivých aktivít. Identifikácia a analýza anomálií je pre odborníkov v digitálnej forenznej oblasti kľúčová, aby odhalili potenciálne narušenia bezpečnosti, stopy o trestnej činnosti alebo jednoducho pochopili príčinu neočakávanej udalosti.

2.1 Typy anomálií

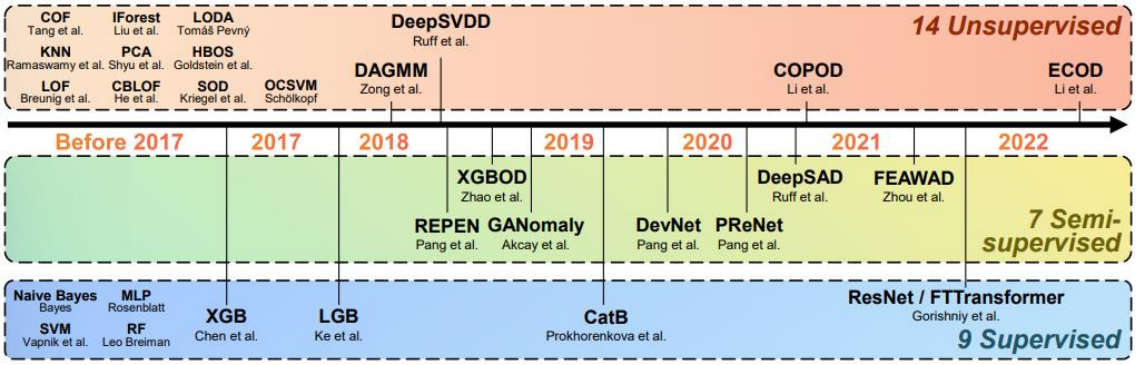
Anomálie možno klasifikovať do rôznych typov na základe ich charakteristík a kontextu, v ktorom sa nachádzajú:

- **Bodové anomálie** – Najjednoduchší typ, kde sa jednotlivé dátové body alebo udalosti výrazne odchyľujú od normy alebo očakávaného správania. Napríklad náhly nárast sieťovej prevádzky alebo neoprávnený pokus o prihlásenie [12].
- **Kontextové anomálie** – Tieto anomálie sa vyskytujú, keď sa dátový bod alebo udalosť považuje za nezvyčajnú v konkrétnom kontexte, ako napríklad prihlásenie používateľa do systému v nezvyčajnom čase alebo z neočakávaného miesta [12].
- **Kolektívne anomálie** – Tieto anomálie zahŕňajú súbor súvisiacich dátových bodov alebo udalostí, ktoré, keď sa zvažujú spoločne, vykazujú abnormálne správanie. Príkladom kolektívnej anomálie môže byť séria neúspešných pokusov o prihlásenie, po ktorých nasleduje úspešné prihlásenie a odcudzenie údajov [12].

2.2 Súvisiace práce

Detekcia anomálií je témou veľkého množstva vedeckých výskumov, článkov a kníh. Za zmienku stojí obsiahly článok napísaný Chandolim a kol. [12], ktorý opisuje rôzne typy techník použitých v rôznych oblastiach, avšak nie pri digitálnej forenznej analýze. Článok od Ahmeda a kol. [13] poskytol porovnanie metód pri detekcii neoprávneného prístupu do siete, čo ale pre náš výskum nie je veľmi relevantné. Rozsiahla kniha o analýze anomálií od Charu C. Aggarwala [14] poskytuje veľké množstvo informácii o typoch metód na detekciu anomálií, detailne opisujú ako fungujú, či aké sú medzi jednotlivými typmi metód rozdiely. Veľké porovnanie metód zrealizovali Han a kol. [15] pomocou knižníc PyOD [16] a scikit-learn [17], ktoré sme využili aj my. Ich benchmark pozostával z 30 metód, ktoré sú rozdelené podľa typu a zoradené podľa roku implementácie na Obrázku 2. Týchto 30 metód bolo otestovaných na 57 vybraných datasetoch. Avšak tieto datasety boli z oblasti medicíny, finančníctva a pod. Detekcia anomálií z oblasti digitálnej forenznej analýzy bola obsiahnutá v iných prácach.

Studiawan a kol. [18] bližšie porovnali niekoľko metód s učiteľom a bez učiteľa. Tieto metódy použili pri detekcii anomálií v nevyvážených autentifikačných záznamoch. Xu a kol. [19] analyzovali konzolové záznamy pomocou metódy PCA, ktorá im poskytla sľubné výsledky.



Obr. 2 Rozdelenie otestovaných metód ADBench [15]

2.3 Metódy detekcie anomálií

Ako sme v úvode naznačili, existuje veľké množstvo metód na detekciu anomálneho správania, z ktorých každá má svoje silné stránky a limitácie. Poznáme 3 základné typy metód na detekciu anomálií podľa úrovne dozoru. Metódy s učiteľom (supervised), ktoré vyžadujú dataset, kde sú označkované (labeled) dáta ako „normálne“ a „abnormálne“. Pomocou týchto dát sa metóda učí. Tento prístup sa však pri detekcii anomálií používa zriedkavo kvôli všeobecnej nedostupnosti označkovaných dát. Druhým typom sú semi-supervised metódy, ktoré potrebujú čiastočne označkované dáta. Môže to byť akákoľvek kombinácia normálnych alebo anomálnych údajov. Metódy detekcie anomálií bez učiteľa (unsupervised) predpokladajú, že údaje sú neoznačené a sú najčastejšie používané kvôli ich širšiemu a relevantnému použitiu. My sme sa však zaoberali len metódami bez učiteľa, pretože sú najrozšírenejším typom metód kvôli už spomínanému nedostatku verejne dostupných roztriedených datasetov. Na implementáciu metód sme použili dve Python knižnice: **PyOD** a **scikit-learn**.

V tejto podkapitole si rozoberieme niekoľko typov metód na detekciu anomálií. Klasifikácia týchto metód do rôznych typov je založená na ich základných princípoch alebo technikách používaných na detekciu anomálií.

2.3.1 Štatistické metódy

Z hľadiska detekcie anomálií možno metódy založené na štatistickom modeli rozdeliť do dvoch skupín: parametrické a neparametrické metódy. Parametrické metódy predpokladajú, že údaje sledujú špecifické rozdelenie a ich úlohou je naučiť sa parametre tohto rozdelenia. Príklady parametrických metód zahŕňajú Gaussove modely zmesí (GMM) a lineárnu regresiu [20]. Akonáhle je model prispôsobený, parametrické metódy sú zvyčajne rýchle pri detekcii anomálií. Na druhej strane neparametrické metódy nepredpokladajú žiadne špecifické rozdelenie údajov [21]. Do tejto kategórie patria napríklad metódy založené na histograme (HBOS), metóda **ECOD** [22] alebo **COPOD**, ktorá je založená na empirických modeloch kopule [23]. V porovnaní s parametrickými modelmi môžu byť neparametrické metódy výpočtovo nákladnejšie.

2.3.2 Metódy založené na blízkosti

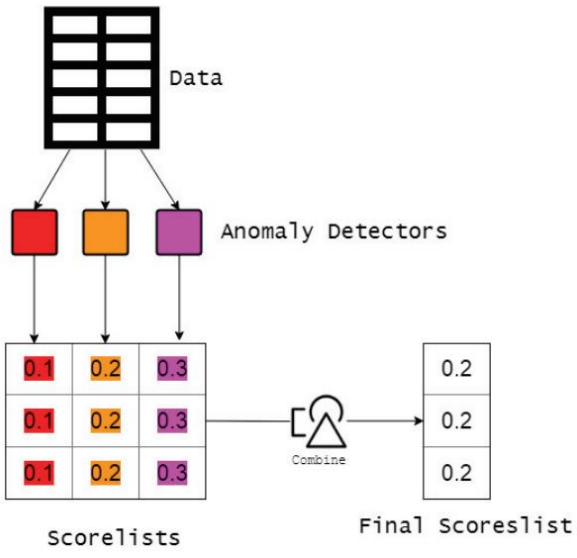
Techniky založené na blízkosti (proximity-based) definujú údajový bod ako anomáliu, keď jeho lokalita (alebo blízkosť) je riedko osídlená. Blízkosť dátového bodu môže byť definovaná rôznymi spôsobmi, ktoré sa od seba jemne líšia. Najbežnejšie spôsoby sú nasledujúce [14]:

- **Metódy založené na zhlukovaní (cluster-based)** – Tieto metódy zahŕňajú analýzu dátových bodov vo vzťahu ku zhlukom. Anomálne skóre pre dátový bod je určené faktormi, ako je jeho neprísluśnosť k akémukoľvek zhluku, jeho vzdialenosť od iných zhlukov, veľkosť najbližšieho klastra alebo kombinácia týchto faktorov. Body v podstate buď patria do zhlukov, alebo sa považujú za anomálie [14]. K takýmto metódam patrí K-means a **CBLOF** [24].
- **Metódy založené na vzdialenosti (distance-based)** – Vzdialenosť dátového bodu od jeho k-najbližšieho suseda sa používa na definovanie blízkosti. Dátové body s veľkými vzdialenosťami k najbližšiemu susedovi sú definované ako anomálie [14]. Tento prístup predpokladá, že normálne dátové objekty majú určitú hustotu susedstva [20]. Tento spôsob používa napríklad metóda kNN alebo **SOD** [25].

- **Metódy založené na hustote (density-based)** – Základným princípom týchto metód je, že anomáliu možno nájsť v oblasti s nízkou hustotou, zatiaľ čo normálne body sa pravdepodobne vyskytujú v zahustených oblastiach. **LOF** [26] je veľmi populárna metóda, ktorá počíta pomer medzi lokálnou hustotou bodu a lokálnou hustotou jeho najbližších susedov. Bod sa považuje za odľahlý, ak je jeho hodnota LOF vysoká. Tento spôsob je účinnejší ako metódy založené na vzdialenosti, ale nie je veľmi účinný vo vysoko dimenzionálnych dátach [27].

2.3.3 Súbory outlierov (outlier ensembles)

Ensemble-based prístup je založený na používaní metódy (alebo súboru metód) viackrát na rôznych nastaveniach datasetu (rôzne podpriestory alebo podmnožiny). Následne sa agreguje skóre, aby sa získalo konečné skóre anomálie. Tento prístup predpokladá, že rôzne modely robia rôzne chyby v úsudku, ktoré by sa dali zmierniť kombináciou výsledkov. Tieto metódy často používajú existujúce metódy, ako je LOF, ktorých výsledky potom spriemerujú [28]. Tento proces možno vidieť na Obrázku 3. Niekoľko vytvorených metód demonštruje, že súbor veľmi slabých detektorov anomálií môže viesť k silnému detektoru anomálií. Takouto metódou je napríklad **IForest** [29], jej vylepšený variant **iNNE** [28] alebo metóda **Loda** [30].



Obr. 3 Proces ensemble-based metód [31]

2.3.4 Lineárne modely

Tieto metódy sú založené na predpoklade, že dáta môžu byť reprezentované lineárnou kombináciou ich vlastností. Anomálie sa detegujú identifikáciou údajových bodov, ktoré sa výrazne odchyľujú od predikcie lineárneho modelu. Lineárne modely pri detekcii anomálií sú zvyčajne jednoduchšie a rýchlejšie ako iné metódy, ale môžu byť menej účinné pri identifikácii zložitých alebo nelineárnych vzťahov v dátach. Medzi metódy, ktoré používajú lineárny model patrí **PCA** [32], **OC-SVM** [33], lineárna alebo robustná regresia. Robustná regresia je odolnejšia voči anomáliám a šumom v dátach ako tradičná lineárna regresia. Táto metóda minimalizuje vplyv extrémnych hodnôt na odhad parametrov modelu. Pri detekcii anomálií sú odchýlky od robustného modelu považované za anomálie [14].

2.3.5 Neurónové siete

Konkrétnejšie hĺbkovo učiace (deep learning) neurónové siete sú obzvlášť vhodné na učenie sa reprezentácií údajov, ktoré majú hierarchický charakter, ako sú obrázky alebo text. Metódy, ktoré využívajú tieto siete môžeme kategorizovať na „zmiešané“ a „úplne hlboké“. V zmiešaných sieťach sa reprezentácie naučia oddelene v predchádzajúcom kroku predtým, ako sa tieto reprezentácie potom vložia do klasických (plytkých) metód detekcie anomálií ako je OCSVM. Naproti tomu úplne hlboké neurónové siete využívajú cieľ učenia sa reprezentácie priamo na zisťovanie anomálií. Metódou, ktorá využíva práve takýto prístup je napríklad **Deep SVDD** [34].

3 Dataset a metodológia

V tejto kapitole sa bližšie pozrieme na vybraný modelový prípad, predspracovanie dát a metodológiu nášho spracovania, a napokon bližší pohľad na hlavné metódy, ktoré sme použili v našom programe.

3.1 Prípad

Ako náš modelový prípad sme si vybrali Case001 – Ukradnutá sečuánská omáčka zo stránky DFIR Madness. Tento prípad sa zaoberá ukradnutím tajného receptu na sečuánsku omáčku spoločnosti CITADEL, ktorý bol uverejnený na dark webe. Úlohou je zodpovedať na určené otázky ako napríklad, aký bol počiatočný vstupný vektor (ako sa tam útočník dostal), aký malvér bol použitý, či útočník pristúpil k iným systémom a či útočník ukradol aj iné citlivé súbory. Na výber je niekoľko úrovní obtiažnosti podľa toho, aké dôkazy si vyberieme analyzovať. DFIR Madness poskytuje nasledujúce digitálne stopy na účely forenznej analýzy [35]:

- DC01 Disk Image (E01)
- DC01 Memory and PageFile
- DC01 Autoruns
- DC01 Protected Files
- Case001 PCAP
- Desktop Disk Image (E01)
- Desktop Memory and PageFile
- Desktop Autoruns
- Desktop Protected Files

V našej práci sme bližšie pracovali len s klonom disku DC01 Disk Image (E01).

3.2 Dataset

Dataset bol predpripravený v predošlom výskume. Z obrazu disku servera, odkiaľ sa ukradol tajný recept, sa vytvorila časová os pomocou nástroja Plaso (log2timeline) a následne sa pomocou parseru psort.py prekonvertovala do formátu l2tcsv. Týmto procesom sa získal CSV súbor so 17 poliami a 1 256 180 záznamami. V Tabuľke 3 môžeme vidieť rozdelenie do 11 rámcov podľa poľa „Zdroj“. Ďalej sa pracovalo len so záznamami zo zdroja „FILE“, ktorých je 843 863 [36].

Zdroj	Počet záznamov
AMCACHE	136
AMCACHEPROGRAM	3
EVT	86 180
FILE	843 863
LNK	45
LOG	194
OLECF	253
PE	18 115
RECBIN	1
REG	307 315
WEBHIST	75
Celkovo	1 256 180

Tab. 1 Rozdelenie záznamov podľa zdroju

V ďalšom kroku sa konvertovali kategorické atribúty každého záznamu na binárne a rozdelili sa do 7 kategórii:

- timestamp = ’M’, ’A’, ’C’, ’B’
- source_type = ’file_stat’, ’NTFS _file_stat’, ’file _entry_shell_item’, ’NTFS _USN_change’
- file_type = ’filef’, ’directory’, ’link’
- dir_type = ’dir _appdata’, ’dir _win’, ’dir _user’, ’dir _other’
- file_type2 = ’file _executable’, ’file _graphic’, ’file _documents’, ’file _ps’, ’file _other’

- file_format = ’mft’, ’lnk_shell_items’, ’olecf_olecf_automatic_destinations/lnk/shell_items’, ’winreg _bagmru/shell_items’, ’usnjrnl’
- file_size = ’size _none’, ’size _Q1’, ’size _Q2’, ’size _Q3’, ’size _Q4’

Dataset sa vymedzil na čas, kedy útok prebiehal. Ten trval od 22:24:50 dňa 18.9.2020 do 4:52:45 dňa 19.9.2020. Manuálnou forenznou analýzou sa identifikovalo 15 relevantných inodov: 84630, 84880, 84987, 86966, 86967, 86968, 86970, 86971, 86975, 87059, 87060, 87064, 87111, 87112 a 87137. Záznamy s inodmi 0 a 84656 sa vynechali, pretože sa nachádzali v datasete najčastejšie. Následne sa tieto dáta agregovali funkciou *sum* podľa atribútu „inode“, čo vytvorilo finálny počet dát, a to 487. Na analýzu sa použilo celkovo 126 kombinácií podľa atribútu „inode“ so siedmimi kategóriami atribútov (ako je popísané vyššie), od použitia jednej kategórie až po všetkých sedem kategórií atribútov [36].

3.3 Porovnanie otestovaných metód

V Tabuľke 1 môžeme vidieť zoznam nami otestovaných metód. Zahrnutie týchto informácii do nášho pozorovania nám umožnilo zvážiť celý rad faktorov pri výbere najlepších metód pre náš výskum. Typ metódy nás oboznámil o základnom prístupe a technikách použitých v každej metóde. S typom súvisí aj časová zložitosť. Tá bola obzvlášť užitočná, pretože nám pomohla určiť praktickosť každej metódy z hľadiska výpočtových zdrojov a času potrebného na vykonanie našej analýzy. Pri vzorci časovej zložitosti n je počet údajových bodov a d je počet dimenzii.

Metóda	Rok impl.	Typ	Časová zložitosť
ECOD	2022	Statistical	O(n.d)
COPOD	2020	Statistical	
LOF	2000	Proximity-Based	$O(n^2)$
CBLOF	2003	Proximity-Based	O(n)
SOD	2009	Proximity-Based	$O(n^2.d)$
ROD	2020	Proximity-Based	$O(n.\frac{d(d-1)(d-2)}{6})$
IForest	2008	Outlier Ensembles	O(n)
iNNE	2018	Outlier Ensembles	O(n)
Loda	2016	Outlier Ensembles	$O(n.k.d^{-1/2})$
PCA	2003	Linear Model	$O(d^2.n+d^3)$
OCSVM	2001	Linear Model	$O(n^3)$
DeepSVDD	2018	Neural Networks	

Tab. 2 Porovnanie rôznych metód detekcie anomálií

V Tabuľke 2 je možné vidieť porovnanie výsledkov otestovaných metód. Pri ich implementácii sme vyskúšali rôzne parametre, aké má daná metóda k dispozícii. Parametre, ktoré sme použili sú zvýraznené tučným písmom. Väčšina metód má podobný výber parametrov. K tým najčastejším, ktoré sme skúmali patrí *kontaminácia* (v Tabuľke 2 označená ako „cont.“), ktorou manuálne nastavujeme pomer anomálií v datasete. Pri všetkých metódach s týmto parametrom sme dospeli k najlepšiemu výsledku nastavením kontaminácie na hodnotu 0.1, čo je 10%. Táto hodnota je aj predvolená pri volaní vybraných metód. Ďalšími skúmanými parametrami je počet susedov na použitie pri metódach založených na blízkosti, *n_neighbors* a *n_estimators*, ktorým sa určuje počet základných odhadov v súbore pri Outlier ensembles metódach. Pri týchto parametroch sme taktiež videli ideálne výsledky s predvolenými hodnotami. V stĺpcoch „TP“ (True Positive) a „FP“ (False Positive) možno vidieť, akú úspešnosť má daná metóda, pričom do tabuľky sme vyberali dva najlepšie výsledky konkrétnej metódy. Hľadaných bolo 15

podozrivých inodov[1], avšak nie každá metóda ich všetky vedela detegovať. Niektoré metódy taktiež detegovali relatívne veľké množstvo falošne pozitívnych inodov. To zdôrazňuje dôležitosť starostlivého výberu vhodnej metódy pre danú úlohu, pretože rôzne metódy môžu mať rôzne silné a slabé stránky. Aby sme vybrali najlepšie metódy pre náš výskum, zvážili sme niekoľko faktorov vrátane doby trvania každej metódy, počet výsledkov, kde metóda nedetegovala žiadne anomálie, ako aj jej schopnosti odhaliť všetkých 15 inodov a minimalizovať falošne pozitívne inody. Analýzou výsledkov sme boli schopní vybrať niekoľko metód, ktoré spĺňali naše kritériá a poskytovali najlepšiu rovnováhu medzi výkonom a efektívnosťou.

Metóda	Čas behu	Parametre	TP	FP
ECOD	5,3 s	**cont.**, n_jobs	15	31
			15	32
COPOD	5,8 s	**cont.**, n_jobs	15	21
			15	30
LOF	4 min 15 s	**n_neighbors**, n_jobs, **metric**, **cont.**, …	15	30
			15	31
CBLOF	3 min 39 s	**n_clusters**, **cont.**, …	13	31
			13	31
SOD	15 min 31 s	**n_neighbors**, ref_set, alpha, **cont.**	14	29
			14	34
ROD	30 min 11 s	**cont.,** parallel_exec	14	23
			14	26
IForest	4 min 42 s	**n_estim.**, **cont.**, bootstrap, …	15	27
			15	30
iNNE	4 min 29 s	**n_estim.**, **cont.**, …	15	26
			15	27
Loda	5 min 30 s	**cont.**, **n_bins**, …	15	16
			15	33
PCA	4,45 s	n_components, **cont.**, copy, whiten, ...	15	28
			15	30

[1] Inode je dátová štruktúra, ktorá obsahuje metadáta pre každý súbor a adresár [37].

OCSVM	1 min 6 s	**kernel**, **gamma**, **nu**, shrinking, …	15 14	30 27
DeepSVDD	50 min 8 s	hidden_neurons, epochs, **cont.**, ...	15 15	25 27

Tab. 3 Porovnanie výsledkov metód

Z hľadiska výsledkov úspešného detegovania všetkých 15 inodov a efektívnej dĺžky trvania metód, sme sa rozhodli pre bližšiu analýzu porovnať metódy ECOD, PCA, IForest a OCSVM.

3.4 Vybrané metódy

Porovnaním výsledkov metód v predošlej kapitole sme vybrali štyri metódy, ktoré si bližšie opíšeme v tejto podkapitole. Tieto metódy boli vybrané na ďalšiu analýzu vzhľadom na ich sľubné výsledky a potenciál pre praktickú aplikáciu v oblasti detekcie anomálií.

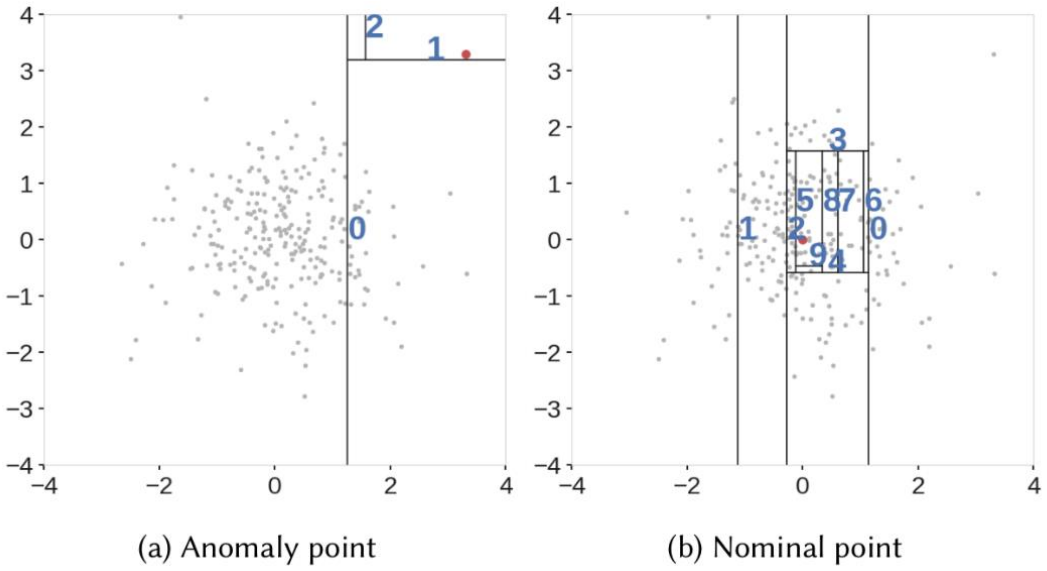
3.4.1 Empirical-Cumulative-distribution-based Outlier Detection (ECOD)

Táto pomerne nová metóda [22] z roku 2022 odhaduje rozdelenie vstupných údajov neparametrickým spôsobom pomocou výpočtu empirických kumulatívnych distribúcií každej dimenzie. Pomocou týchto distribúcií ECOD odhaduje konečné pravdepodobnosti pre každý údajový bod v každej dimenzii. Potom agreguje tieto konečné pravdepodobnosti vo všetkých dimenziách, aby vypočítal celkové anomálne skóre pre každý údajový bod. To umožňuje ECOD odhaliť anomálie v datasete bez spoliehania sa na špecifické predpoklady distribúcie alebo zložité ladenie parametrov [22]. Aj napriek tomu, že metóda je definovaná ako bezparametrická, v knižnici PyOD je implementovaná s parametrami: **contamination** a **n_jobs**.

3.4.2 Isolation Forest (IForest)

IForest [29] funguje tým spôsobom, že náhodne vyberie prvok a rozdelí hodnotu medzi maximálnu a minimálnu hodnotu tohto prvku, čím vytvorí štruktúru podobnú

binárnemu stromu. Tento proces sa rekurzívne opakuje, aby sa vytvorilo niekoľko takýchto stromov. Na detegovanie anomálie IForest zoberie údajový bod a meria, ako dlho trvá izolovanie tohto bodu od zvyšku údajov. Robí sa to spočítaním počtu častí potrebných na oddelenie bodu od zvyšku údajov. Ak je údajový bod anomália, malo by vyžadovať menej rozdelení, aby ho bolo možné izolovať od zvyšku údajov [29].



Obr. 4 Rozdelenie anomálie a bežného údajového bodu [38]

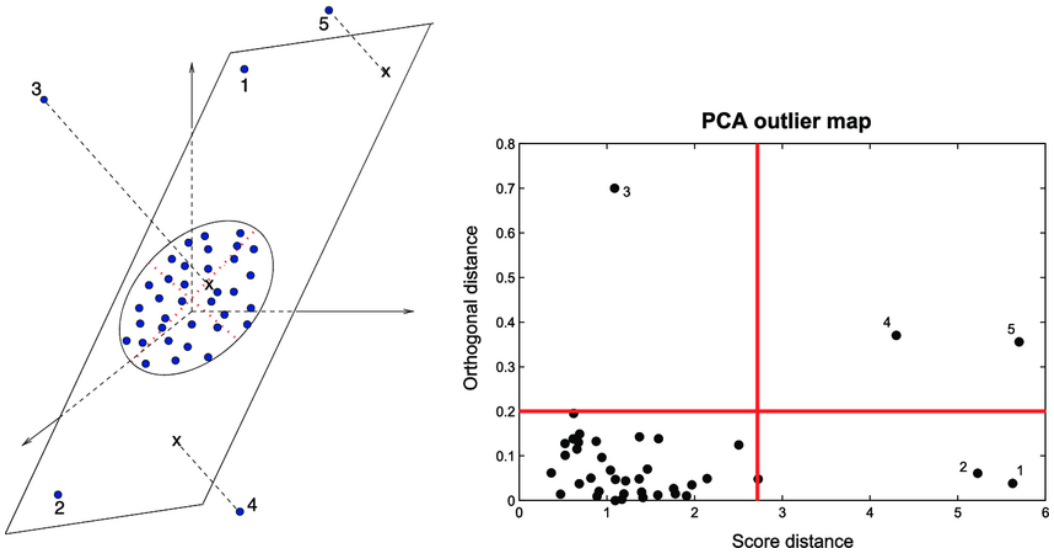
Z Obrázku 4 je možné pozorovať, že bežné dátové body vyžadujú porovnateľne väčší počet rozdelení ako dátový bod anomálie. IForest je implementovaný v knižnici scikit-learn aj PyOD s viacerými funkciami.

Parametre: n_estimators, max_samples, contamination, max_features, bootstrap, n_jobs, behaviour, random_state, verbose

3.4.3 Principal Component Analysis (PCA)

PCA [32] je lineárna redukcia rozmerov pomocou singulárneho rozkladu údajov na ich premietanie do priestoru s nižšou dimenziou. V tomto postupe možno kovariančnú maticu dát rozložiť na ortogonálne vektory, nazývané vlastné vektory, spojené s vlastnými hodnotami. Vlastné vektory s vysokými vlastnými hodnotami zachytávajú väčšinu rozptylov v dátach. Preto nízko rozmerná nadrovina skonštruovaná pomocou vlastných vektorov môže zachytiť väčšinu rozptylov v údajoch. Anomálie sa

však líšia od normálnych údajových bodov, čo je zrejmejšie na nadrovine skonštruovanej vlastnými vektormi s malými vlastnými hodnotami. Preto je možné získať anomálne skóre ako súčet projektovanej vzdialenosti vzorky na všetkých vlastných vektoroch [32]. Metóda PCA je implementovaná v knižnici PyOD.



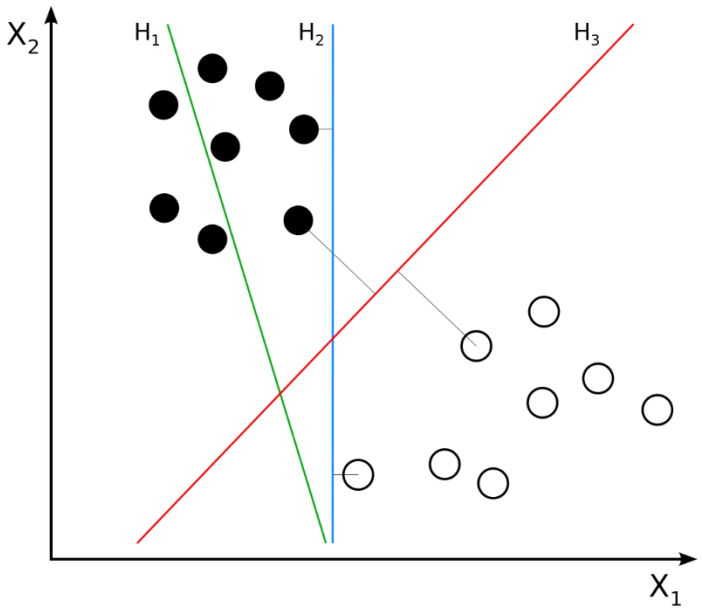
Obr. 5 Detekcia anomálií pomocou PCA [39]

Parametre: n_components, n_selected_components, contamination, copy, whiten, svd_solver, tol, iterated_power, random_state, weighted, standardization

3.4.4 One-Class Support Vector Machine (OCSVM)

One-Class SVM [33] je metóda bez učiteľa, ktorá je variáciou metódy SVM. Tento algoritmus funguje tak, že vytvára hranicu rozhodovania, ktorá zahŕňa väčšinu údajových bodov vo vysoko rozmernom priestore. Táto rozhodovacia hranica je navrhnutá tak, aby maximalizovala odstup medzi hranicou a najbližšími dátovými bodmi, a zároveň minimalizovala počet dátových bodov, ktoré spadajú mimo hranice. OCSVM potom klasifikuje nové údajové body vo vnútri alebo mimo rozhodovacej hranice. Ak je údajový bod klasifikovaný ako mimo hranice, považuje sa za anomáliu. Výkon algoritmu tohto závisí od výberu funkcie kernelu a výberu parametrov modelu. Niektoré bežne používané funkcie kernelu zahŕňajú gaussov (RBF) kernel a sigmoidný kernel. Táto

metóda je podobne ako IForest implementovaná v knižnici scikit-learn aj PyOD s pridanými funkciami.



Obr. 6 Rozdelenie dvoch tried pomocou OCSVM [40]

Na Obrázku 6 môžeme vidieť spôsob, ako sa postupne vytvára najlepšia rozhodovacia hranica. H1 nerozdeľuje triedy, H2 áno, ale len s malým odstupom. H3 ich oddeľuje s maximálnym odstupom.

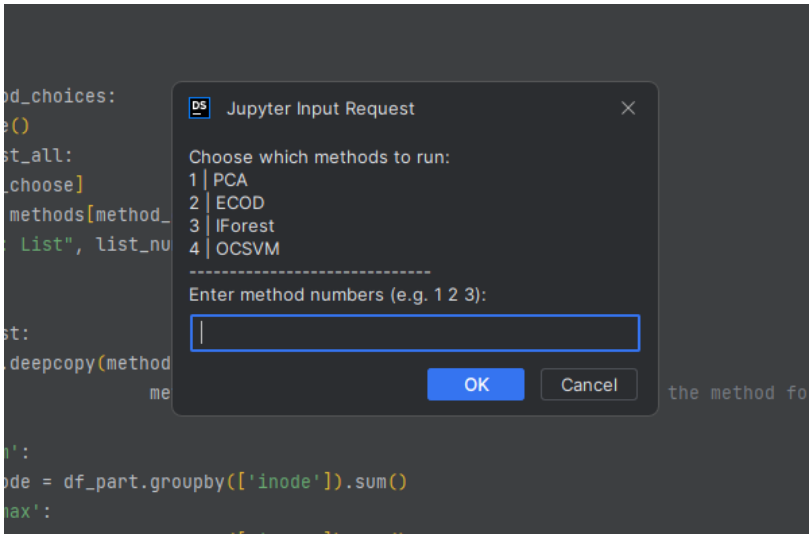
Parametre: kernel, nu, degree, gamma, coef0, tol, shrinking, cache_size, verbose, max_iter, contamination

4 Návrh nástroja a vyhodnotenie

V tejto kapitole si predstavíme nami navrhnutý nastroj, ktorý používateľom umožňuje vybrať si z ponuky metód detekcie anomálií a spustiť ich na datasete. Následne poskytneme výsledky z nášho testovania ECOD, IForest, PCA a OCSVM pomocou nášho nástroja, pričom poukážeme na metódu, ktorú považujeme za najefektívnejšiu. Táto kapitola preskúma výsledky týchto testov a poskytne pohľad na najúčinnejšie metódy zisťovania anomálií, a či bol náš výber podľa zistení v Tabuľke 1 a 2 správny.

4.1 Nástroj

Navrhnutý nástroj je implementovaný v programovacom jazyku Python verzie 3 a využíva viacero knižníc, ako PyOD, scikit-learn, pandas, seaborn, matplotlib a numpy. Užívateľovi sú poskytnuté štyri metódy, ktoré sme bližšie testovali. Avšak užívateľ si môže jednoducho importovať alebo odobrať metódy podľa jeho uváženia. Pri spustení kódu, ktorý je Python skript, bude používateľ vyzvaný na vyber metód, ktoré chce spustiť. Toto menu môžeme vidieť na Obrázku 7, škáluje sa automaticky podľa počtu importovaných metód. Po vykonaní detekcie metód, ktoré boli vybrané, sa spustia ďalšie skripty, ktoré automaticky vygenerujú graf doby trvania jednotlivej metódy v sekundách. Tento graf sa škáluje podľa toho, či niektorá z metód presiahla čas behu cez dve minúty. Vtedy sa graf upraví a ukáže dobu trvania v minútach pre lepšiu čitateľnosť. Následne sa vytvorí celkové porovnanie metód pomocou stĺpcového grafu, heatmapy a boxplotu.

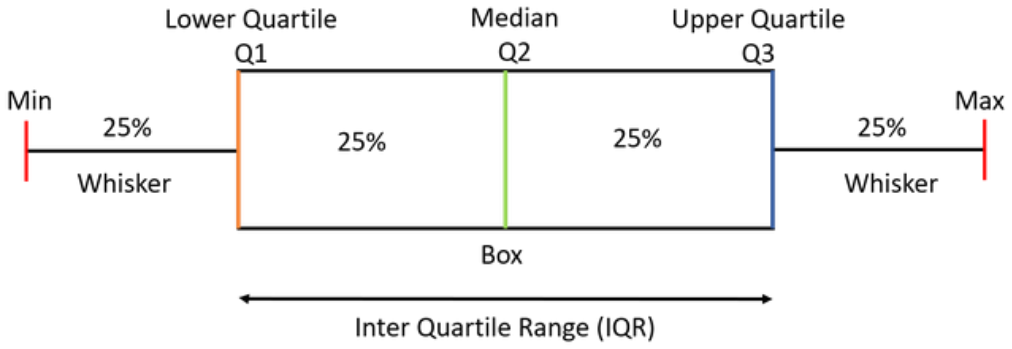


Obr. 7 Ukážka menu

Stĺpcový graf pozostáva z troch stĺpcov pre každú metódu. Výsledky sa v podstate rozdelia na tri hlavné kvartily: Q1, Q2 a Q3. V štatistike je kvartil typ kvantilu a sú to tri body, ktoré rozdeľujú zoradené dáta do štyroch rovnakých skupín, z ktorých každá predstavuje štvrtinu vzorky dát. Prvý kvartil (dolný kvartil) sa rovná 25. percentilu dát (oddelí najmenších 25% dát od najvyšších 75%). Druhý (stredný) kvartil (medián) súboru dát sa rovná 50. percentilu dát (rozdelí zotriedené dáta na polovice). Tretí kvartil, tzv. horný kvartil je rovný 75. percentilu dát (oddelí najmenších 75% dát od najvyšších 25%).

Heatmapa graficky znázorní údaje, kde sú jednotlivé hodnoty obsiahnuté v matici znázornené ako farby. Farby v heatmape označujú veľkosť hodnôt v matici, pričom rôzne farby predstavujú rôzne rozsahy hodnôt.

Boxplot vizuálne zobrazí distribúciu a odchýlenie údajov zobrazením kvartilov údajov a priemerov. Boxploty zobrazujú päťčíselný súhrn výsledkov: minimálne skóre, prvý (dolný) kvartil, medián, tretí (horný) kvartil a maximálne skóre. Toto zobrazenie môžeme vidieť na Obrázku 8.

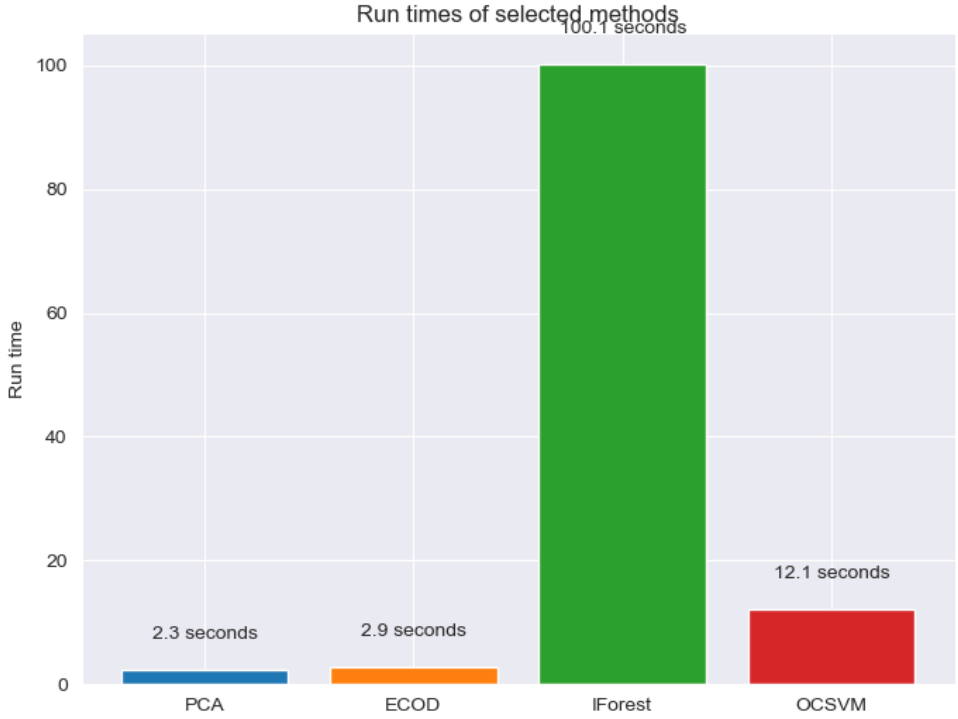


Obr. 8 Schéma boxplotu [41]

4.2 Výsledky

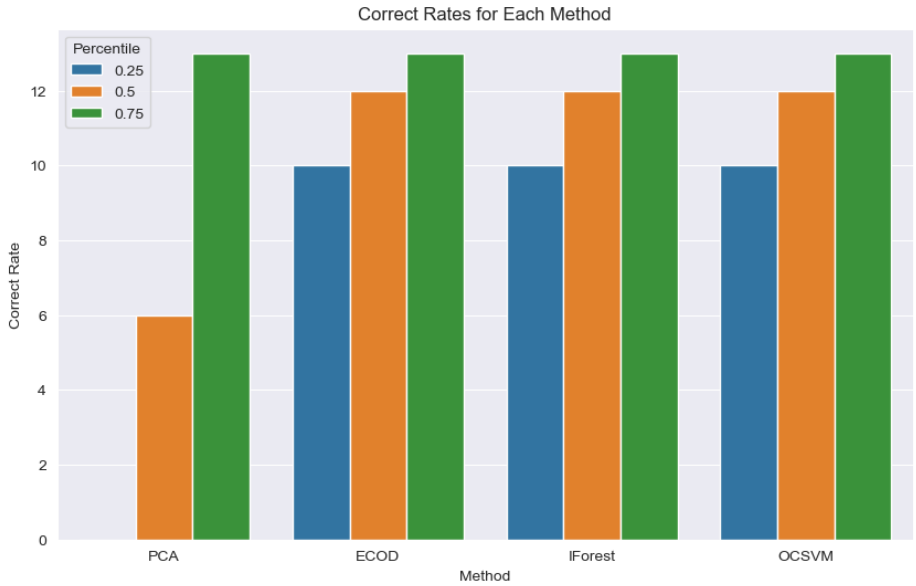
Ako bolo spomenuté v úvode kapitoly, v našom implementovanom nástroji sme uskutočňovali porovnanie metód ECOD, IForest, PCA a OCSVM. Výsledky sa ukladajú do jednoduchého CSV súboru so 7 stĺpcami: „Method“, „Attributes“, „Agg“, „Outl“, „Outl_count“, „Correct“ a „Rate“. S týmto súborom sa ďalej pracuje na vizualizáciou výsledkov.

Parametre metód sme nemenili. Ako sme spomenuli vyššie v kapitole 3.3, predvolené nastavenia nám dávali pri týchto metódach ideálne výsledky a nemuseli sme ich nijako meniť. Vďaka tomuto spôsobu sme dosiahli ešte rýchlejšie časy behu každej metódy. Porovnanie doby trvania môžeme vidieť na Obrázku 9. ECOD a PCA sú jednými z najrýchlejších metód vôbec. OCSVM bol pomerne prekvapujúco taktiež rýchly, čo pravdepodobne zapríčinil predvolený kernel RBF. Naopak IForest trval necelé dve minúty. V porovnaní s dobou trvania manuálnej forenznej analýzy, ktorá v závislosti od druhu a závažnosti incidentu môže trvať dlho, je to však stále veľmi rýchla metóda.

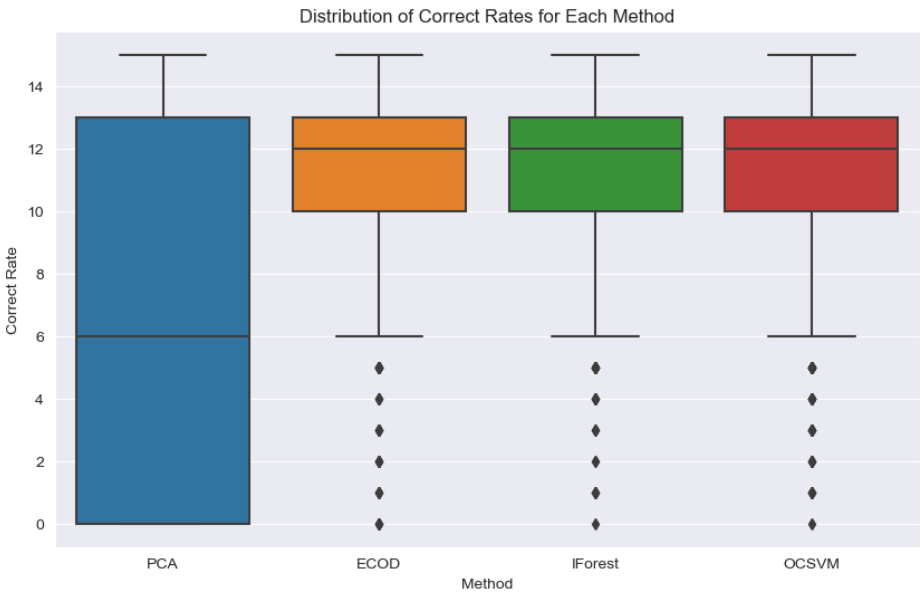


Obr. 9 Doba trvania metód

Pre ďalšiu vizualizáciu sme súbor s našimi výsledkami prefiltrovali na už spomínane kvartily pre lepšie pochopenie výsledkov. Na Obrázku 10 a 11 môžeme vidieť, že všetky metódy okrem PCA nám dávajú na prvý pohlaď rovnaké výsledky. Boxploty nám ukazujú, že všetky metódy vedia detegovať 15 inodov. Medián nájdených inodov metód ECOD, IForest a OCSVM je 12, pričom dolný kvartil je 10. Metóda PCA viditeľne zaostáva pri detekcii anomálií, aj keď vie detegovať všetky inody. Avšak rozpätie jej výsledkov je pomerne veľké a veľakrát tato metóda nedeteguje žiadne anomálie.



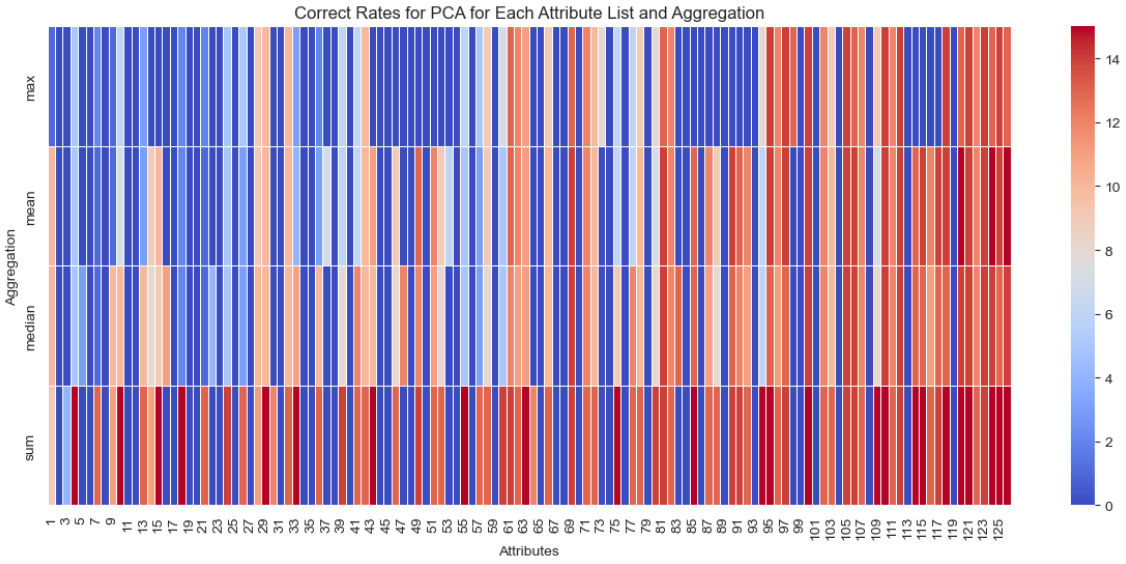
Obr. 10 Graf počtu správne detegovaných anomálií



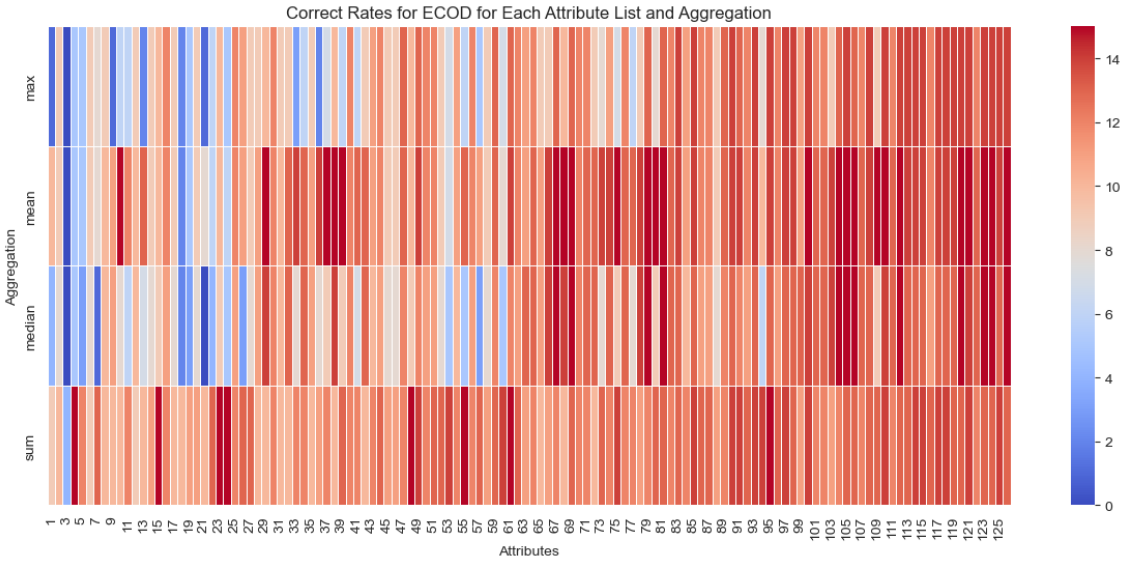
Obr. 11 Boxplot rozpätia správne detegovaných anomálií

Pomocou heatmapy vieme bližšie preskúmať jednotlivé metódy na Obrázkoch 12, 13 a 14. Ako sme si mohli všimnúť, ECOD, IForest a OCSVM nám na prvý pohlaď dávajú rovnaké výsledky zo stĺpcového grafu a boxplotu. Na heatmape však tieto metódy majú rozdielnu distribúciu výsledkov. Každá metóda deteguje anomálie podľa atribútov a použitej agregačnej funkcie rozdielne. Kombinácii použitých atribútov bolo 126, ktoré sme bližšie opísali v kapitole 3.2. Taktiež môžeme bližšie vidieť výsledky metódy PCA na Obrázku 11, ktorá má veľké množstvo nulových inštancii, hlavne pri agregovaní

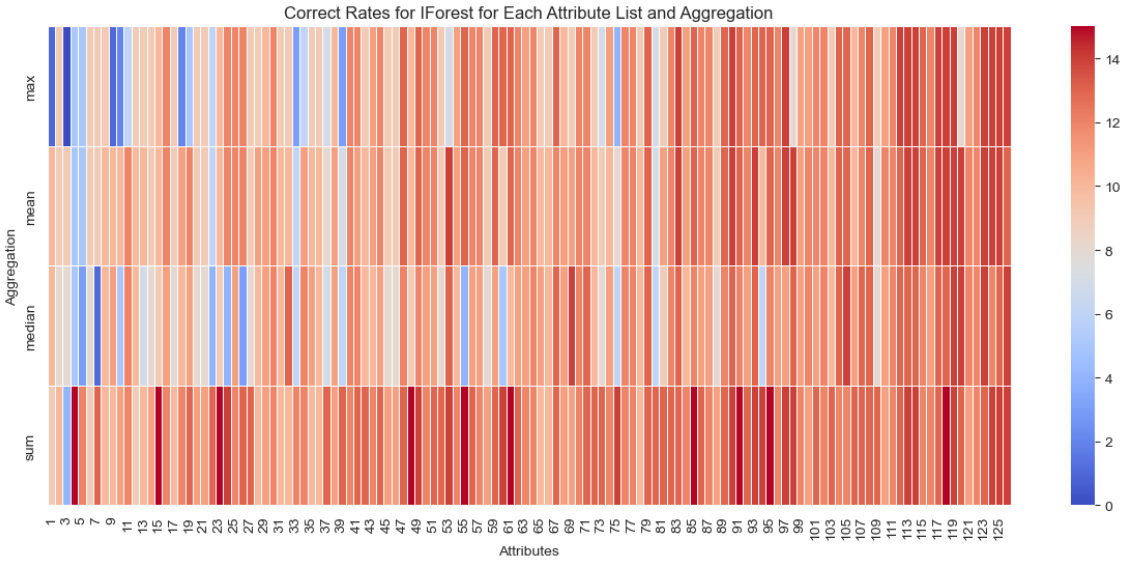
s funkciou *max*. Pri funkcii *sum* deteguje najviac plný počet podozrivých inodov, avšak tieto výsledky sú zmiešané s veľkým počtom nulových výsledkov. Porovnaním heatmapy PCA na Obrázku 11 s heatmapou ECOD na Obrázku 12 vidíme značný rozdiel v detekcii anomálií. ECOD dosahuje najlepšie výsledky pri agregácii *mean* a *sum*, predovšetkým prvá menovaná.



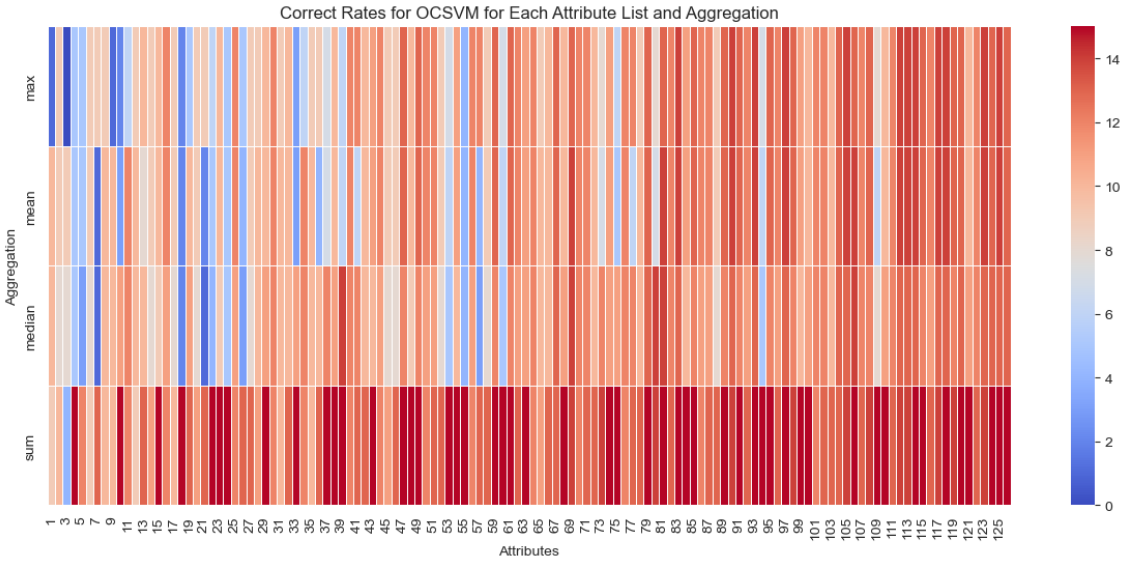
Obr. 12 Heatmapa metódy PCA



Obr. 13 Heatmapa metódy ECOD



Obr. 14 Heatmapa metódy IForest



Obr. 15 Heatmapa metódy OCSVM

IForest a OCSVM majú veľmi podobné heatmapy. Môžeme si všimnúť, že sú svojimi výsledkami bližšie k ECOD metóde. Všetky metódy okrem ECOD majú svoje najlepšie výsledky pri agregačnej funkcii *sum*. OCSVM má pri nej veľmi vysokú detekciu inodov.

Náš nástroj taktiež vypočítava metriky Precision, Recall a F1 skóre. **Precision** meria podiel skutočných pozitív (TP) medzi všetkými predpokladanými pozitívami. Inými slovami, meria, koľko prípadov, ktoré model klasifikoval ako pozitívne, bolo

skutočne pozitívnych. Vyššia presnosť znamená, že model má menej falošných pozitív (FP). Vzorec na výpočet je: $\frac{TP}{TP+FP}$.

Recall na druhej strane meria podiel skutočných pozitív medzi všetkými skutočnými pozitívami. Inými slovami, meria, koľko skutočných pozitívnych prípadov bol model schopný správne identifikovať. Vyššie číslo znamená, že model má menej falošne negatívnych výsledkov (FN). Vzorec na výpočet: $\frac{TP}{TP+FN}$.

F1 skóre je metrika, ktorá spája Precision a Recall do jednej hodnoty, ktorá predstavuje ich harmonický priemer. Je užitočné v situáciách, keď sa porovnávajú modely, ktoré môžu mať rôzne kompromisy medzi Precision a Recall metrikou [42]. Počíta sa takto: $\frac{2 \times Precision \times Recall}{Precision + Recall}$.

Nástroj tieto metriky vypočíta, spriemeruje a na výstupe nám vyrobí tabuľku podobnú Tabuľke 4.

Metóda	Agregácia	Precision priemer	Recall priemer	F1 skóre priemer
PCA	max	0.106145	0.289947	0.152132
	mean	0.170802	0.396296	0.227802
	median	0.190101	0.426455	0.248266
	sum	0.182022	0.552381	0.273472
ECOD	max	0.340927	0.676190	0.428853
	mean	**0.373327**	0.805291	**0.482171**
	median	0.350344	0.702116	0.435387
	sum	0.312525	0.814286	0.445033
IForest	max	0.339989	0.685714	0.429447
	mean	0.348646	0.742857	0.448242
	median	0.325510	0.685714	0.410545
	sum	0.287209	0.822222	0.421321
OCSVM	max	0.338436	0.682540	0.427162
	mean	0.326846	0.684127	0.416920
	median	0.339862	0.690476	0.422614
	sum	0.307985	**0.892063**	0.453429

Tab. 4 Výsledky metrík

Z Tabuľky 4 vidíme, že ECOD má v priemere najmenej falošne pozitívnych detekcií pri použití agregačnej funkcie *mean*. Naopak PCA má celkovo najviac pri agregácii *max*. Metóda OCSVM má v priemere najmenej falošne negatívnych výsledkov pri funkcii *sum*, čo sme si mohli všimnúť pri analýze heatmapy. Avšak celkovo najlepšie F1 skóre v priemere má metóda ECOD pri použití agregačnej funkcie *mean*. Výsledky Isolation Forest sú pomerne presné pri všetkých agregačných funkciách, ale s dlhou dobou trvania oproti ostatným metódam. PCA bola pri našom testovaní tesne najrýchlejšia, avšak výsledkami je najhoršia.

Ako bolo naznačené, pri našej analýze bolo dôležité porovnanie viacerými spôsobmi. Na vyvodenie presnejších a pravdivejších záverov sme museli použiť viacero grafov a štatistických metód. Porovnaním boxplotov a heatmáp sme videli nedostačujúce výsledky PCA. Na prvý pohľad boli najlepšie výsledky OCSVM pri agregovaní *sum*. Ale pri finálnom F1 skóre sa ukázalo, že značne najlepším kompromisom medzi detekciou falošne pozitívnych a falošne negatívnych anomálii je metóda ECOD.

Záver

Cieľom tejto práce bolo analyzovať forenzné artefakty v operačnom systéme a porovnať existujúce prístupy k identifikácii anomálií v rámci forenzného vyšetrovania. Na základe tohto porovnania sme navrhli nástroj pre identifikáciu podozrivých forenzných artefaktov, a pomocou neho sme otestovali najlepšie metódy. Z našej práce je zrejmé, že forenzné artefakty poskytujú dôležité informácie pre forenzných vyšetrovateľov a môžu byť úspešne využité pri identifikácii podozrivých aktivít.

V prvej kapitole sme analyzovali forenzné artefakty v operačnom systéme Windows. Priblížili sme si dôležitosť digitálnej forenznej analýzy, jej proces a techniky, ktoré sa počas tohto procesu môžu použiť. Bližšie sme si opísali, čo je forenzný artefakt a aké artefakty poznáme v súborovom systéme NFTS operačného systému Windows.

V druhej kapitole sme si zadefinovali, čo je to anomália a aké typy anomálií poznáme. Predstavili sme metódy detekcie anomálií. Poznáme niekoľko typov podľa toho, akú techniku používajú na detekciu, ako napríklad štatistické metódy, metódy s použitím lineárneho modelu či metódy založené na blízkosti.

V tretej kapitole bližšie opisujeme dataset, s ktorým sme pracovali. Implementovali sme 12 z nich pomocou knižníc PyOD a scikit-learn. Otestovali sme ich na modelovom prípade ukradnutej sečuánskej omáčky z portálu DFIR Madness. Každá z vybraných metód má na výber niekoľko parametrov, ktoré sa dajú manuálne nastaviť. Kombinácie niekoľkých z nich sme taktiež otestovali, ale pri našom datasete nám stačili predvolené parametre. Vďaka predošlému výskumu sa identifikovalo 15 podozrivých inodov, ktoré sa metódy snažia detegovať. V tejto kapitole sme bližšie opísali metódy, ktoré mali pre našu prácu najväčší potenciál. Boli štyri a to ECOD, IForest, PCA a OCSVM.

V poslednej kapitole je spracovaný tretí cieľ tejto práce. Opísali sme návrh nášho nástroja, ktorý je napísaný v jazyku Python. Používateľ tohto nástroja ma možnosť importovať rôzny počet metód, ktoré chce použiť. Následne si v menu vyberie metódy, ktoré chce aplikovať. Pre porovnanie ma používateľ k dispozícii graf doby trvania každej metódy, grafy porovnania úspešne detegovaných inodov, boxploty a heatmapy metód. Z nich je možné usúdiť, aká metóda je najefektívnejšia pre potreby užívateľa. Okrem grafov sa taktiež automaticky vygeneruje tabuľka s metrikami Precision, Recall a F1

skóre, pri ktorých je možné odpozorovať, aká metóda má menej falošných pozitív, menej falošne negatívnych výsledkov.

V závere práce sme zhodnotili výsledky štyroch techník, ktoré sme si vybrali bližšie analyzovať. Z našich záverov vyplýva, že metóda založená na štatistickom modeli ECOD dosahuje najlepšie výsledky, menovite pri použití agregačnej funkcie *mean*. Tato metóda ma jednoduchú časovú zložitosť, tým pádom je rýchla na našom datasete. IForest a OCSVM mali podobne výsledky, ale OCSVM je o viac ako minútu rýchlejšia. Prekvapením bola metóda PCA, ktorá dosahovala veľmi rýchle časy behu, na prvý pohlaď aj celkom dobre výsledky, ale v závere mala najhoršie skóre oproti všetkým ostatným metódam.

Zoznam použitej literatúry

1. Understanding Digital Forensics: Process, Techniques, and Tools. [online] 20.3.2022 Dostupné z: https://www.bluevoyant.com/knowledge-center/understanding-digital-forensics-process-techniques-and-tools#digital-forensic-tools

2. The Basics of Digital Forensics. [online] 10.4.2022 Dostupné z: https://www.exterro.com/basics-of-digital-forensics

3. What Are the 5 Stages of a Digital Forensics Investigation? [online] 10.4.2022 Dostupné z: https://ermprotect.com/blog/what-are-the-5-stages-of-a-digital-forensics-investigation/

4. What Is Digital Forensics? [online] 20.3.2022 Dostupné z: https://www.simplilearn.com/what-is-digital-forensics-article#steps_of_digital_forensics

5. Link Analysis & Timeline Analysis in Digital Forensics Investigation [online] 20.3.2022 Dostupné z: https://www.mailxaminer.com/blog/link-analysis-timeline-analysis-in-digital-forensic/

6. Windows Forensics Analysis — Windows Artifacts (Part I) [online] 20.3.2022 Dostupné z: https://nasbench.medium.com/windows-forensics-analysis-windows-artifacts-part-i-c7ad81ada16c

7. Investigating ThumbCache. [online] 20.3.2022 Dostupné z: https://forensafe.com/blogs/thumbCache.html

8. Digital Forensics: Artifact Profile – Recycle Bin. [online] 20.3.2022 Dostupné z: https://www.magnetforensics.com/blog/artifact-profile-recycle-bin/

9. Investigating OpenSaveMRU. [online] 20.3.2022 Dostupné z: https://forensafe.com/blogs/opensavemru.html

10. Windows Forensics Analysis — Windows Artifacts (Part II). [online] 20.3.2022 Dostupné z: https://nasbench.medium.com/windows-forensics-analysis-windows-artifacts-part-ii-71b8fa68d8a1

11. Investigating Prefetch. [online] 20.3.2022 Dostupné z: https://forensafe.com/blogs/prefetch.html

12. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection. *ACM Computing Surveys*, *41*(3), 1–58. Dostupné z: https://doi.org/10.1145/1541880.1541882
13. Ahmed, M., Mahmood, N., & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, *60*, 19–31. Dostupné z: https://doi.org/10.1016/j.jnca.2015.11.016
14. Aggarwal, C. C. (2017). Outlier Analysis. *Springer EBooks*. Dostupné z: https://doi.org/10.1007/978-3-319-47578-3
15. Han, S., Hu, X., Huang, H., Jiang, M., & Zhao, Y. (2022). ADBench: Anomaly Detection Benchmark. *Social Science Research Network*. Dostupné z: https://doi.org/10.2139/ssrn.4266498
16. Zhao, Y., Nasrullah, Z., & Li, Z. (2019). PyOD: A Python Toolbox for Scalable Outlier Detection. *Journal of Machine Learning Research*, *20*(96), 1–7. Dostupné z: https://jmlr.org/papers/volume20/19-011/19-011.pdf
17. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, Édouard Duchesnay; (2011). *Scikit-learn: Machine Learning in Python*. Dostupné z: https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html
18. Studiawan, H., & Sohel, F. (2020). *Performance Evaluation of Anomaly Detection in Imbalanced System Log Data*. Dostupné z: https://doi.org/10.1109/worlds450073.2020.9210329
19. Xu, W., Huang, L., Fox, A., Patterson, D. A., & Jordan, M. I. (2010). Detecting Large-Scale System Problems by Mining Console Logs. In *International Conference on Machine Learning* (pp. 37–46). Dostupné z: https://icml.cc/Conferences/2010/papers/902.pdf
20. Li, Z., Zhao, Y., Hu, X., Botta, N., Jansson, P., & Chen, G. (2022). ECOD: Unsupervised Outlier Detection Using Empirical Cumulative Distribution Functions. *IEEE Transactions on Knowledge and Data Engineering*, 1. Dostupné z: https://doi.org/10.1109/tkde.2022.3159580
21. Metrics, Techniques and Tools of Anomaly Detection: A Survey. [online] Dostupné z: https://www.cse.wustl.edu/~jain/cse567-17/ftp/mttad/index.html
22. Li, Z., Zhao, Y., Hu, X., Botta, N., Jansson, P., & Chen, G. (2022b). ECOD: Unsupervised Outlier Detection Using Empirical Cumulative Distribution Functions.

IEEE Transactions on Knowledge and Data Engineering, 1. Dostupné z: https://doi.org/10.1109/tkde.2022.3159580

23. Li, Z., Zhao, Y., Botta, N., Jansson, P., & Hu, X. (2020). COPOD: Copula-Based Outlier Detection. *ArXiv (Cornell University)*. Dostupné z: https://doi.org/10.1109/icdm50108.2020.00135
24. He, Z., Xu, X., & Deng, S. (2003). Discovering cluster-based local outliers. *Pattern Recognition Letters*, *24*(9–10), 1641–1650. Dostupné z: https://doi.org/10.1016/s0167-8655(03)00003-5
25. Kriegel, H., Kröger, P., Schubert, E., & Zimek, A. (2009). Outlier Detection in Axis-Parallel Subspaces of High Dimensional Data. *Springer eBooks*, 831–838. Dostupné z: https://doi.org/10.1007/978-3-642-01307-2_86
26. Breunig, M., Kriegel, H., Ng, R. T., & Sander, J. (2000). LOF. *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data - SIGMOD '00*. Dostupné z: https://doi.org/10.1145/342009.335388
27. Almardeny, Y., Boujnah, N., & Cleary, F. (2020). A Novel Outlier Detection Method for Multivariate Data. *IEEE Transactions on Knowledge and Data Engineering*, *34*(9), 4052–4062. Dostupné z: https://doi.org/10.1109/tkde.2020.3036524
28. Bandaragoda, T., Ting, K. M., Albrecht, D. W., Liu, F., Zhu, Y., & Wells, J. C. K. (2018). Isolation-based anomaly detection using nearest-neighbor ensembles. Computational Intelligence, 34(4), 968–998. Dostupné z: https://doi.org/10.1111/coin.12156
29. Liu, F., Ting, K. M., & Zhou, Z. (2008). Isolation Forest. *International Conference on Data Mining*. Dostupné z: https://doi.org/10.1109/icdm.2008.17
30. Pevný, T. (2016). Loda: Lightweight on-line detector of anomalies. *Machine Learning*, *102*(2), 275–304. Dostupné z: https://doi.org/10.1007/s10994-015-5521-0
31. Chiang, A., David, E., Lee, Y., Leshem, G., & Yeh, Y. (2017b). A study on anomaly detection ensembles. Journal of Applied Logic, 21, 1–13. Dostupné z: https://doi.org/10.1016/j.jal.2016.12.002
32. Shyu, M., Chen, S., Sarinnapakorn, K., & Chang, L. (2003). A Novel Anomaly Detection Scheme Based on Principal Component Classifier. *International Conference on Data Mining*, 172–179. Dostupné z: https://apps.dtic.mil/sti/pdfs/ADA465712.pdf

33. Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the Support of a High-Dimensional Distribution. *Neural Computation*, *13*(7), 1443–1471. Dostupné z: https://doi.org/10.1162/089976601750264965
34. Ruff, L., Vandermeulen, R. A., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., Müller, E., & Kloft, M. (2018). Deep One-Class Classification. *International Conference on Machine Learning*, 4393–4402. Dostupné z: http://proceedings.mlr.press/v80/ruff18a/ruff18a.pdf
35. The Case of the Stolen Szechuan Sauce. [online] 10.4.2022 Dostupné z: https://dfirmadness.com/the-stolen-szechuan-sauce/
36. *Detection of relevant digital evidence in the forensic timelines*. (2022). IEEE Conference Publication | IEEE Xplore. Dostupné z: https://ieeexplore.ieee.org/document/9847438
37. Carrier, B. (2005). *File System Forensic Analysis*. Addison-Wesley Professional.
38. *Extended Isolation Forest*. (2021). IEEE Journals & Magazine | IEEE Xplore. Dostupné z: https://ieeexplore.ieee.org/abstract/document/8888179
39. Rousseeuw, P. J., & Hubert, M. (2018). Anomaly detection by robust statistics. *Wiley Interdisciplinary Reviews-Data Mining and Knowledge Discovery*, *8*(2). Dostupné z: https://doi.org/10.1002/widm.1236
40. Support vector machine. [online] 10.4.2022 Dostupné z: https://en.wikipedia.org/wiki/Support_vector_machine#/media/File:Svm_separating_hyperplanes_(SVG).svg
41. Box Plot. [online] 10.4.2022 Dostupné z: https://www.geeksforgeeks.org/box-plot/
42. The F1 score. [online] 10.4.2022 Dostupné z: https://towardsdatascience.com/the-f1-score-bec2bbc38aa6

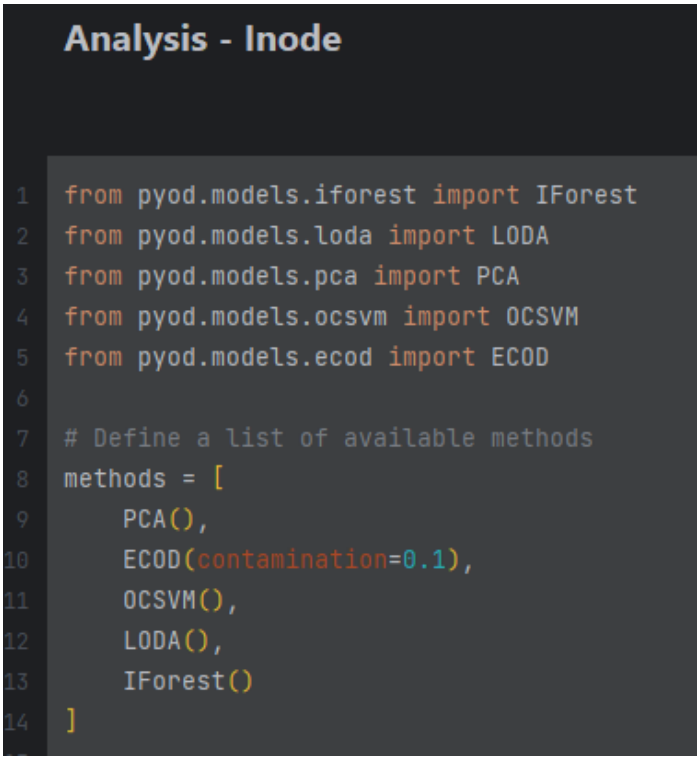
Prílohy

Príloha A: Zdrojové kódy implementácii metód a nástroja na detekciu anomálií

Príloha B: Návod na použitie

Príloha B: Návod na použitie

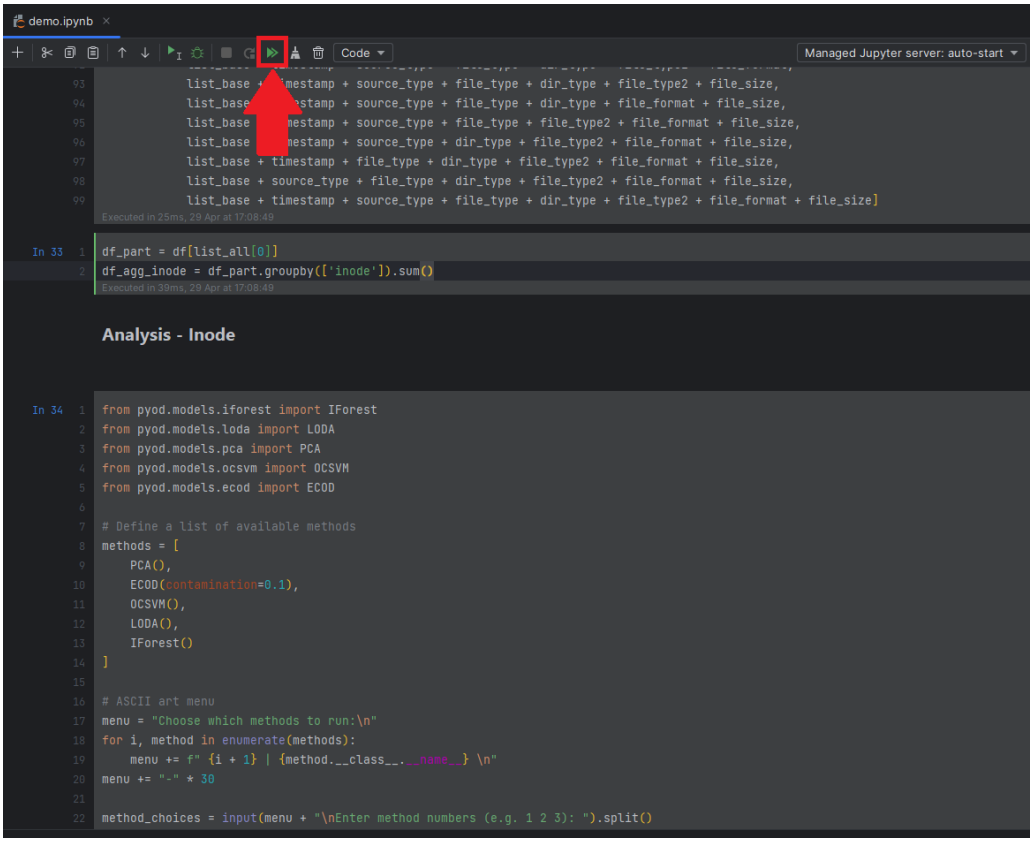
Náš nástroj je implementovaný v jazyku Python ako Jupyter Notebook. Názov súboru je *demo.ipynb*. Pre začiatok práce s nástrojom na detekciu anomálií treba súbor načítať pomocou programu Jupyter alebo Dataspell od spoločnosti JetBrains. Tento návod bol tvorený v Dataspelli, avšak postup je rovnaký ako v programe Jupyter. Prvá časť kódu je predspracovanie dát, ktoré užívateľ nemení. Nástroj nebol otestovaný na inom datasete, ale pri podobnom spracovaní dát má tento nástroj potenciál fungovať korektne. Po prejdení na označenú časť ako „Analysis - Inode“, si užívateľ môže vybrať metódy z ponuky knižníc PyOD a scikit-learn. Základné metódy, ktoré sme skúmali my sú pridané. Iné sa dajú jednoducho pridať, ako je vidno na Obrázku 1.



Obr. 16 Pridanie metód

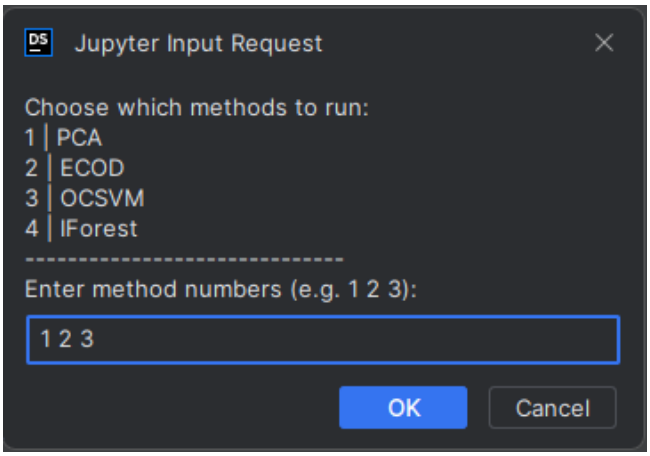
Pre niektoré metódy je možné nastaviť parametre podľa požiadaviek v zátvorkách konkrétnej metódy. Tento prístup je taktiež vidno na Obrázku 1, kde pre metódu ECOD sme nastavili *kontamináciu* na hodnotu 0,1. Avšak, ak si užívateľ nie je istý, ktoré parametre použiť, môže ich ponechať na predvolených hodnotách.

Po nastavení metód a prípadných parametrov, kliknite na tlačidlo "Run All" (Obrázok 2) a počkajte, kým sa zobrazí ponuka s možnosťami ako na Obrázku 3.



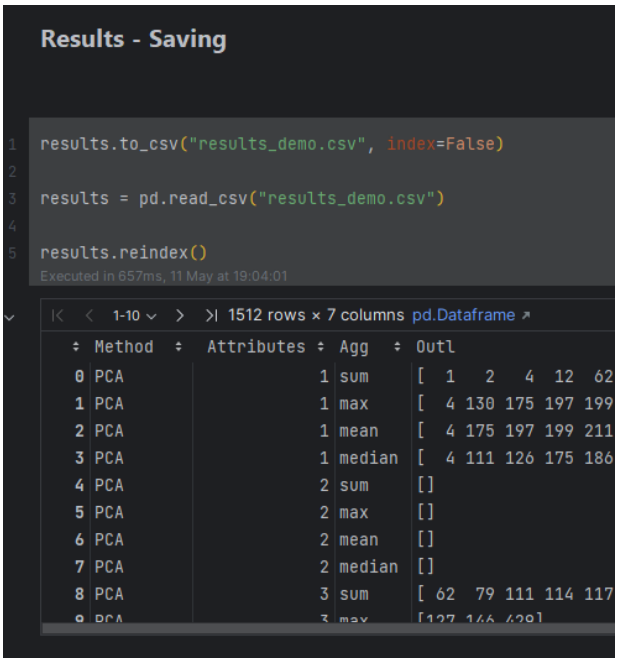
Obr. 17 Spustenie nástroja

Menu je jednoduché a prehľadné. Metódy sú očíslované a užívateľ si vyberá im priradené čísla. Na Obrázku 3 sme si ako príklad vybrali metódy PCA, ECOD a OCSVM. Po stlačení tlačidla „OK“ sa spustí detekcia vybranými metódami.



Obr. 18 Menu výberu

Výsledky sú rozdelené do štyroch častí. Prvá časť „Results – Run Time“ generuje graf s časom behu každej metódy. V časti „Results - Saving“ sa výsledky ukladajú s názvom, aký si užívateľ zadá. Na Obrázku 4 sme si nazvali súbor ako „results_demo.csv“.



Obr. 19 Uloženie výsledkov

Nasledujúca časť „Results - Bar Plots, Boxplots and Heatmaps“ vygeneruje potrebné grafy na detailnejšiu analýzu pomocou vizualizácie výsledkov. Každý graf sa dá uložiť ako PNG súbor. V poslednej časti „Results - Statistical Tests“ sa vygeneruje tabuľka s výsledkami metrík Precision, Recall a F1 skóre.