MISKOLCI EGYETEM

GÉPÉSZMÉRNÖKI ÉS INFORMATIKAI KAR

HATVANY JÓZSEF INFORMATIKAI TUDOMÁNYOK DOKTORI ISKOLA

RENDELLENESSÉG ALAPÚ BEHATOLÁS ÉRZÉKELŐ RENDSZEREK GÉPI TANULÁSI MÓDSZERREL TÖRTÉNŐ TANÍTÁSA

című PhD értekezés

KÉSZÍTETTE:
Göcs László
okleveles informatika szakos tanár

DOKTORI ISKOLA VEZETŐ:
Prof. Dr. Szigeti Jenő
egyetemi tanár

TÉMAVEZETŐ:
Dr. habil. Johanyák Zsolt Csaba
főiskolai tanár

Miskolc, 2023.

Nyilatkozat

Alulírott Göcs László (BVX3AL) kijelentem, és sajátkezű aláírásommal igazolom, hogy ezt a doktori disszertációt magam készítettem, és abban csak a megadott forrásokat használtam fel. Minden olyan részt, amelyet szó szerint, vagy azonos tartalomban, átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Miskolc, 2023. augusztus 31.

Göcs László

Témavezetői ajánlás

Göcs Lászlóval 2011-óta dolgozok együtt a Neumann János Egyetem Informatika Tanszékén, illetve annak jogelődeinél. Kezdetben számítógépes hálózatokhoz kapcsolódó közös tárgyak alapozták meg az együttműködésünket, ami később kiterjedt az informatikai biztonsággal kapcsolatos kutatásra is, így örömmel fogadtam el felkérését, hogy vállaljam el PhD tanulmányainak témavezetését. A sok éves közös munka során megbízható, önálló tudományos kutatásra alkalmas lelkiismeretes kollégaként ismertem meg.

Göcs László sikeresen megvalósította a fokozatszerzés érdekében célként kitűzött tudományos feladatokat. Az általa elért eredmények jól alkalmazhatók a sokdimenziós és nagyszámú rekorddal dolgozó gépi tanulási feladatok során különös tekintettel behatolásérzékelő rendszerek osztályozó moduljának gépi tanulással történő fejlesztésére.

Göcs László kutatómunkája során elért eredményeit magyar és angol nyelvű konferencia és folyóirat cikkekben tette közzé. Összesen 4 angol nyelvű folyóiratcikk (ebből egy Q2 minősítésű 2,8-as impakt faktorral rendelkező folyóiratban jelent meg), 3 magyar nyelvű folyóiratcikk, 3 angol nyelvű konferenciacikk és 2 magyar nyelvű konferenciacikk született PhD tanulmányaihoz kapcsolódóan.

Az értekezés tézisei Göcs László saját kutatási munkájának eredményeit foglalják össze. Az értekezés és a tézisekhez kapcsolódó publikációk alapján messzemenően támogatom és javaslom számára a Ph.D. fokozat odaítélését.

Miskolc, 2023. augusztus 31.

Dr. habil. Johanyák Zsolt Csaba
témavezető

Köszönetnyilvánítás

Ezúton szeretném megköszönni témavezetőmnek, Dr. habil Johanyák Zsolt Csabának a kutatómunkában nyújtott óriási támogatását, szakmai segítségét, különösen a gépi tanulás témakörökben, és hogy a kutatási munkám alatt bármikor fordulhattam hozzá segítségért. Köszönöm a Neumann János Egyetem GAMF Kar vezetésének a lehetőséget, hogy a munkám mellett a doktori tanulmányomat végezhettem, valamint az Informatika Tanszék valamennyi munkatársának, hogy a kutatásom során segítettek szakmai kérdésekben. Továbbá szeretném megköszönni szüleim biztatását, valamint a családom, feleségem és kisfiam türelmét, kitartását és támogatását, ami nélkül e munka nem jöhetett volna létre.

Miskolc, 2023. augusztus 31.

Göcs László

Tartalomjegyzék

Ábrajegyzék

Táblázatok jegyzéke

Rövidítések jegyzéke

Rövidítés	Megnevezés	Rövidítés	Megnevezés
ANOVA	Analysis of variance	**IDS**	Intrusion Detection System
BIDS	Behavior based IDS	**IG**	Information Gain
CART	Classification regression Trees	**IGF**	Information Gain Feature Evaluaton
CFS	Correlation based Feature Selection	**IPS**	Intrusion Prevencion System
ChS-R	Chi-square integrated with RReliefF	**ITLBO**	Improved Teacher Learner Based Optimization
CIDS	Centralized IDS	**KIDS**	Knowledge-based IDS
CIDS	Centralized IDS	**KNN**	k-Nearest Neighbors
DDoS	Distributed Denial of Service	**LR**	Logistic Regression
DFS	Density-Based Feature Selection	**MCDM**	Multi-Criteria Decision-Making
DIDS	Distributed IDS	**NB**	Naive Bayes
DoE	Design of Experiments	**NIDS**	Network based IDS
DoS	Denial of Service	**NN**	Neural Network
DT	Decision Tree	**PART**	Partial Tree
EFS	Ensemble Feature Selection	**PCA**	Principal Component Analysis
EFS-MI	EFS - Mutual-information	**RF**	Random Forest
GR	Gain Ratio	**RT**	Random Tree
HDD	High-Dimensional Datasets	**SA-EFS**	Ensemble Feature Selection based on Sort Aggregation
HDLSS	High-Dimensional, Low Sample Size	**SGD**	Stochastic Gradient Descent
HIDS	Host based IDS	**SVM**	Support Vector Machine
HYDS	Hybrid IDS		

1. Bevezetés

Manapság amikor egy vállalat, egy szervezet az informatikai, hálózati infrastruktúráját tervezi, működteti, a biztonság a legfontosabb szempontok közé tartozik. Napjainkban arra kényszerülünk, hogy ne csak a munkatársakkal, hanem a partnerekkel is számítógépen, illetve számítógépes hálózaton keresztül, minél egyszerűbben, gyorsabban és biztonságosabban tudjunk kommunikálni. Ennek a tendenciának sajnos az a legnagyobb hátránya, hogy a rosszindulatú tevékenységet folytatók előtt is nagyobb lehetőségek nyílnak.

A biztonság több részből tevődik össze: egyrészt védeni kell a hálózati erőforrásokat, hogy azokhoz csak a megfelelő felhasználók és a megfelelő jogosultsággal férjenek hozzá. Másrészt biztosítani kell két fél között a biztonságos kommunikációt [1].

A biztonság meglétét az informatikai rendszerek teljes életciklusaiban (tervezés, fejlesztés, bevezetés, üzemeltetés, megszüntetés) biztosítanunk kell, ahol folyamatosan ellenőrizni kell a fenyegetettség szintjét, a biztonság meglétét, valamint a biztonsági intézkedések végrehajtását és hatékonyságát. Ahhoz, hogy kialakítsunk egy biztonságos rendszert meg kell határozni a biztonsági stratégiánkat, mely az alábbiakból tevődik össze:

- meghatározzuk a védelmi célokat,
- kiválasztjuk és elhatároljuk azokat a területeket, amelyeken a biztonsági rendszereket kialakítani és az intézkedéseket érvényesíteni kell,
- meghatározzuk a biztonsági tervezés módszerét,
- körvonalazzuk a minimális követelményeket,
- megtervezzük és ütemezzük az intézményre vonatkozó biztonsági intézkedéseket, beleértve a katasztrófaelhárítást is,
- meghatározzuk a követhetőség és a menedzselhetőség követelményeit, valamint a felügyelet és az ellenőrzés rendszerét [2].

Fontos, hogy az informatikai **biztonsági stratégia** része az intézmény globális biztonsági stratégiájának, összhangban kell lennie az intézmény működési és informatikai stratégiájával, valamint ki kell szolgálnia az intézmény célkitűzéseit, továbbá biztosítani kell az alapvető funkciókat:

- **Megbízhatóság** (Reliability)
    - Egy információtechnológiai összetevő azon képessége, hogy ellásson egy megkívánt funkciót meghatározott körülmények között, egy meghatározott időtartamra.

- **Karbantarthatóság** (Maintainability)
    - Egy számítógépes komponens vagy szolgáltatás azon képessége, hogy meg lehet tartani egy olyan állapotban, vagy vissza lehet állítani egy olyan állapotba, amelyben képes ellátni a megkívánt funkciót.
- **Szolgáltatási képesség** (Serviceability)
    - Szerződéses kikötés, amely meghatározza az informatikai komponens rendelkezésre-állását az adott összetevőket szolgáltató és karbantartó külső szervezettel való megegyezés szerint.
- **Biztonság** (Security)
    - Lehetővé teszi a számítógépes komponensek vagy informatikai szolgáltatások elérését biztonságos körülmények között.

Az informatikai biztonság tervezéséhez, a stratégia kialakításához szükséges, hogy ismerjük a rendszer különböző területeinek kockázatát. Az informatikai kockázatelemzés nem védelmi intézkedés, elvégzése önmagában nem erősíti a védelmet, de segít, hogy létrejöjjön a biztonságos informatikai rendszer. Kockázatot jelenthet például, ha egy üzemeltetett rendszer nincs megfelelően dokumentálva, implementálva, karbantartva, nincs üzletmenet folytonossági terv, vagy nincs katasztrófa elhárítási terv. A kockázatértékelés alapján kellene eldönteni, hogy egy rendszer elindítható-e vagy sem.

A kockázatelemzés megvalósításának lépései:

1. Információgyűjtés: a vállalat gyenge pontjainak feltárása, és ezekhez védelmi szintek hozzárendelése.
2. Értékelés, kockázatok feltárása és elemzése: a beszerzett információk alapján a hiányosságok, fenyegetettségek meghatározása és a kockázatok értékelése.
3. Kockázatelemzési jelentés: a vállalat vezetősége számára jelentés készül, amelyben a 2. lépésben feltárt hiányosságaikat összefoglalja és javaslatot tesznek a szükséges védelmi intézkedések bevezetésére vagy hiánypótlásra.
4. Kockázatkezelési terv: védelmi intézkedések és a ráfordítandó szükséges erőforrások meghatározása.
5. Rendszeres időközönként felülvizsgálat: a jogszabályban előírtaknak megfelelően legalább két évente felül kell vizsgálni (tanúsított információvédelmi irányítási rendszerrel rendelkezők esetében, évente) a vállalat informatikai kockázatelemzés eredményét. Ez oly módon történik, hogy a már meglévő kockázatelemzést elő kell

venni és meg kell nézni, hogy történt-e változás a korábbi vizsgálat során. Ha eltérés van, dokumentálni kell és a korábban leírtaknak megfelelően a lépéseket újra el kell végezni [3].

Minden jól tervezett és kialakított informatikai rendszerhez tartozik egy **minimális biztonsági követelmény**. Az információvédelem és a megbízható működés mellett nagyon nagy szerepet játszik a rendelkezésre állás fogalma. **Rendelkezésre álláson** azt a valószínűséget értjük, amellyel egy definiált időintervallumon belül az alkalmazás a tervezéskor meghatározott funkcionalitási szintnek megfelelően a felhasználó által használható. Rendelkezésre áll egy alkalmazás vagy erőforrás, mikor a működésének képes eleget tenni, képes feladatokat fogadni, működni. Értékét százalékban adják meg. Szerverek esetén ez az az idő, amikor képesek kiszolgálni a klienseket.

$$Rendelkezésre\ állás\ = \frac{T_{\text{üz}} - \sum_{\text{üz}} T_{ki}}{T_{\text{üz}}} \times 100\%, \qquad (1)$$

ahol $T_{üz}$ az üzemidő periódus, amelyre a rendelkezésre állást értelmezzük és T_{ki} a kiesési idő egy alkalomra.

A kiesési időt befolyásolja

- az újraindítási képesség megvalósítása,
- a hibaáthidalás folyamatának kialakítása,
- a rendszerkonfiguráció hatékony menedzselése.

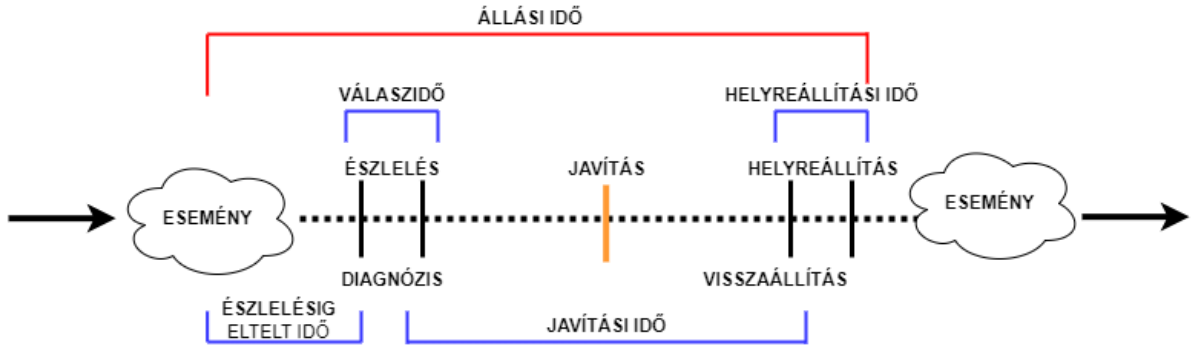
Az informatikai rendszereket és szolgáltatásokat úgy kell tervezni, hogy megbízhatóak, hibatűrők és karbantarthatók legyenek **teljes életciklusuk** során, a tervezéstől a megszüntetésükig. A kézi rendszerekre való visszaállás gyakorlatilag lehetetlen, a felhasználók hatékonysága és eredményessége erősen függ az informatikai szolgáltatások rendelkezésreállásától és megbízhatóságától A szervezeti felhasználók tevékenysége az informatikán alapul, amely nélkül a szervezet működésképtelen [2].

Az informatikai szolgáltatásokat nyújtó rendszerek megbízhatóságát több tényező is befolyásolja. Ezek közé tartozik az informatikai infrastruktúra összetevőinek megbízhatósága és karbantarthatósága, valamint az a környezet, amelyen ezek a rendszerek működnek. Fontos szerepet játszanak ebben a szállítók és külső partnerek, akik felelősek a karbantartásért.

Emellett az informatikai szolgáltató által alkalmazott eljárások és eszközök, valamint az informatikai infrastruktúra konfigurációja is jelentős hatással van a megbízhatóságra.
A rendelkezésreállás javítására két fő lehetőség van:

- csökkenteni kell a hiba fellépésekor megjelenő állásidőt,
- csökkenteni kell az adott időtartamon belüli hibák számát.

Az 1. ábrán látható folyamatábra egy bekövetkezett esemény és a hozzá kapcsolódó helyreállítás közti kapcsolatot mutatja be.



1. ábra Események közti idők meghatározása

Az észlelési időintervallumban a **Behatolás Érzékelő Rendszereknek (IDS – Intrusion Detection Systems)** nevezett automatizált eszközök képesek hatékonyan és leggyorsabban észlelni az informatikai rendszerek elleni támadásokat. Ezek folyamatosan figyelik és értékelik a hálózati csomagok paramétereit. Az IDS olyan szoftveres vagy hardveres megoldás, amely képes észlelni a számítógép vagy akár a hálózat károsítására alkalmas, nem szokványos csomagokat és tevékenységeket. Az IDS-eszköz figyeli a hálózati interfészeken áthaladó forgalmat.
Amint rosszindulatú tevékenységet észlel, riasztási üzenetet küld egy előre konfigurált felügyeleti rendszernek, amely a hálózati eszközök, például biztonsági készülékek vagy forgalomirányítók átkonfigurálásával megakadályozhatja a további támadásokat. Az IDS-t gyakran a megbízható hálózat határán, néha a tűzfalon kívül is telepítik. A behatolásérzékelő rendszerek többféleképpen kategorizálhatók, például az alkalmazott behatolásérzékelési megközelítés (anomália-alapú vagy szignatúra-alapú), a védett rendszer típusa (állomás, hálózat, hibrid), az IDS architektúra (centralizált, elosztott), az elemzéshez használt adatforrás

(hálózati csomagok, rendszerelemzés), a támadás észlelése után nyújtott szolgáltatás szintje (aktív, passzív) és az elemzés időzítése (folyamatos, időintervallum) szerint [4].

Az ebben a tanulmányban közölt eredmények egy olyan vizsgálat során születtek, amely az anomália-alapú behatolásérzékelő rendszerekre (más néven viselkedésalapú IDS-ek - BIDS-ek) összpontosít. Ezek a rendszerek két üzemmódban működnek (tanulás és észlelés). A tanulási üzemmódban a rendszert olyan érzékelőadatokkal táplálják, amelyek tipikus (normál) hálózati és rosszindulatú (támadás) adatokat tartalmaznak. Az osztályozó egységet az adatrekordokhoz tartozó címkék alapján képzik ki és tesztelik.

Érzékelési üzemmódban a teljesen betanított osztályozó modul célja annak meghatározása, hogy az aktuális tevékenység káros-e a rendszer számára vagy sem. Az anomália-alapú megközelítés előnye, hogy gyorsan és dinamikusan képes alkalmazkodni az ismeretlen támadás-típusokhoz. A BIDS-ek az adatfeldolgozás módja alapján három fő kategóriába sorolhatók, nevezetesen statisztikai alapú, tudásalapú és számítási intelligencia alapú [5].

A BIDS-ek legfontosabb összetevője egy osztályozó modul, amely folyamatosan értékeli a hálózati forgalom egyes jellemzőit, és azonosítja a lehetséges fenyegetéseket. Az osztályozó modult általában egy vagy több mintaadatkészlet (normális és rosszindulatú forgalmi adatok) felhasználásával és statisztikai vagy gépi tanulással fejlesztik ki. Az érzékelőktől kapott nyers adatoknak általában több előfeldolgozási lépésen kell átesniük, amíg felhasználhatóvá válnak az osztályozó modul tanításához.

2. Kutatási célok és motiváció

Több mint 10 éve foglalkozom informatikai biztonság szakterülettel. Az egyetemi oktatás mellett vállalatoknak szaktanácsadóként és informatikai igazságügyi szakértőként számtalan esetben találkozom az IT rendszerek sérülékenységeivel és azok problémáival. A jelenleg működő informatikai rendszerek legnagyobb kihívása a biztonság megléte és annak felügyelete. A számítógépes és hálózati biztonsági rendszerek folyamatosan támadásnak vannak kitéve, melyek már szervezett támadások.

Az emberi erővel való monitorozás szinte lehetetlen a mai rendszereknél, ezért fontos szerepet játszik az automatizálása ennek a területnek. A bekövetkezett események, azon belül is a támadások észlelésére a Behatolás Érzékelő Rendszerek (Intrusion Detection System - IDS) vannak rendszeresítve.

Kutatási motivációm az irodalomkutatásban bemutatott módszerekre, eredményekre és technológiákra építve olyan módszer megtalálása, mely az IDS rendszerek tanításában segít. A vizsgálat az egyik leggyakoribb támadási módra, a Brute-Force támadás felismerésére irányul. Az IDS-ek konfigurálására és a legmegfelelőbb algoritmusok meghatározására léteznek tanító adathalmazok, melyek tartalmaznak különféle hálózati kommunikációkat, beleértve a támadási kommunikációkat is.

A célom az, hogy a kiválasztott adathalmaz segítségével olyan osztályozó algoritmust találjak, amely hatékonyan és pontosan tud meghatározni egy esetleges hálózati támadást. Ennek érdekében egyik kitűzött célom az adathalmaz előfeldolgozása, majd a jellemzők fontossági sorrendjének meghatározása különböző módszerek normalizált értékszámainak átlagai alapján, valamint súlyozott átlag módszer alapján végzett rangsorolás segítségével. Ennek köszönhetően több paraméterrel lehet az osztályozó algoritmusokat tanítani és tesztelni, így még pontosabb értékelési szempont alakul ki ahhoz, hogy a legmegfelelőbb algoritmus kerüljön meghatározásra. A kutatásom elméleti eredménye egy olyan módszer kidolgozása, mely segítségével az IDS rendszerek megfelelő paramétereinek és algoritmusainak beállításaival a gyakorlatba való implementálással hatékonyan detektálhatóak a támadások.

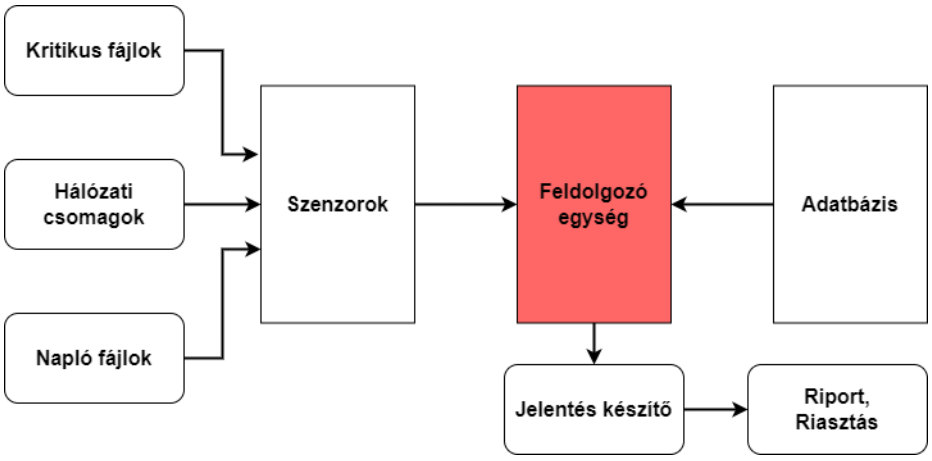
3. Behatolás érzékelő rendszerek (IDS)

A vállalati informatikai rendszerek védelmében fontos szerepet játszanak a behatolás érzékelő rendszerek (Intrusion Detection System – IDS), automatizált védelmet biztosítva a különböző támadások, behatolások ellen [6]. A behatolásérzékelő rendszerek a hálózaton, illetve a számítógépes erőforrásokon olyan speciális események, nyomok után kutatnak, amelyek rosszindulatú tevékenységek, támadások jelei lehetnek. A támadásgyanús helyzeteket felismerve jeleznek vagy beavatkoznak a támadás megakadályozása érdekében [7].

A behatolás érzékelési technikák három fő kategóriába sorolhatók [8]: ezek a szignatúra alapú (tudás alapú), az anomália alapú (viselkedés alapú), valamint a hibrid megoldások [9] családja. Míg az első megközelítés a legegyszerűbb és a leghatékonyabb ismert támadástípusok esetén, addig a második hatékonyan felismeri az új és előre látott sérülékenységek kihasználását. A hibrid rendszerek az előzőekben említett technikákat együttesen alkalmazzák.

Mivel a támadások többsége alkalmazás sérülékenységet céloz meg, és a támadások szintaxisa módosítható a szemantikai háttér megőrzése mellett, ezért a szignatúra alapú rendszerek könnyebben megkerülhetővé váltak, és előtérbe kerültek az anomália alapú megoldások [10].

Az anomália alapú hálózati behatolás érzékelő rendszerek figyelembe vehetik a csomag fejlécet, az adattartalmat vagy ezek kombinációját [11]. Először egy viselkedésmodellt állítanak fel, ami leírja a normális hálózati forgalmat, majd a későbbiekben támadásnak tekintik az összes olyan viselkedést, ami szignifikánsan eltér ettől (pl. [12]). Előnyük, hogy könnyen felismerik az új támadástípusokat. Számos viselkedésmodell felállítási megközelítés létezik, pl. statisztikai, kognitív vagy számítási intelligencia módszereken alapuló [13] [14] [15] [16]. Az IDS-ek elvi működését a 2. ábra szemlélteti.

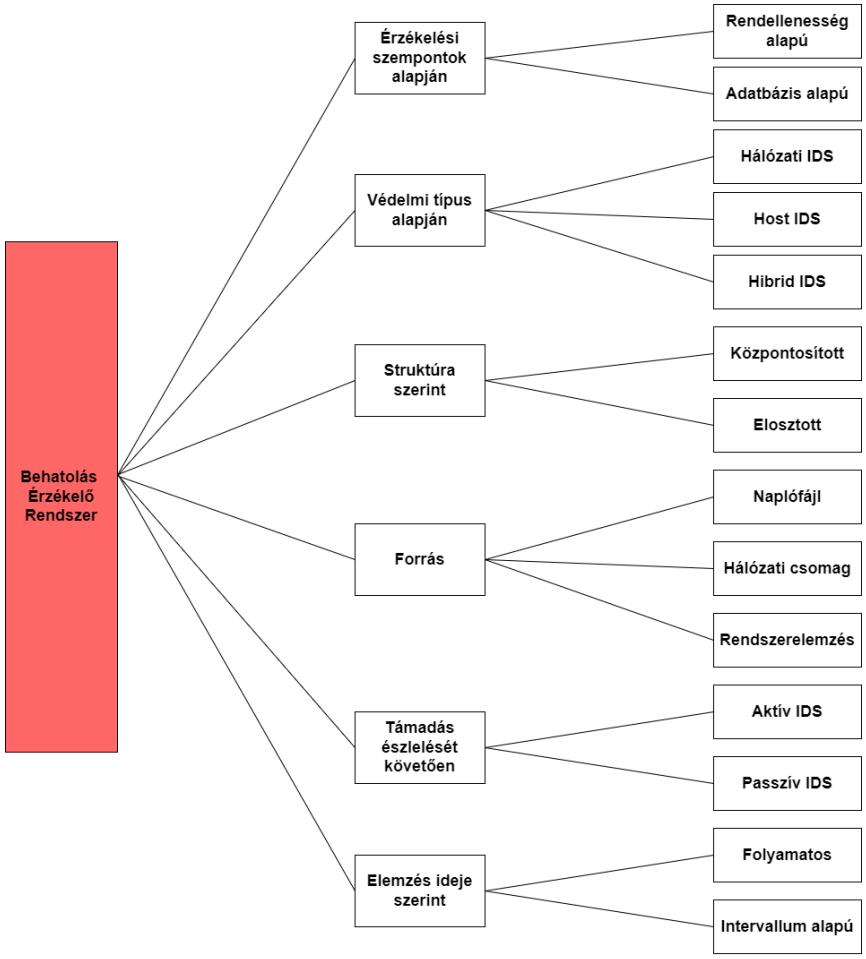


2. ábra Egy IDS rendszer elvi működése

- Szenzorok: figyelemmel kíséri és rögzíti az IDS által feldolgozott tevékenységeket.
- Feldolgozó egység: elemzi a gyűjtött adatokat és összehasonlítja azokat az adatbázisban tárolt ismert rosszindulatú tevékenységi bejegyzésekkel.
- Adatbázis: az ismert és feltételezett káros tevékenységek gyűjteménye.
- Riport generátor: riasztás a rendszergazdák számára és naplózza az IDS tevékenységeket [17].

3.1. A behatolás érzékelő rendszerek csoportosítása

Az IDS rendszereket több szempontból tudjuk csoportosítani, mely lehet az IDS elhelyezkedése, a vizsgálat ideje, vagy épp a kiépítése. A csoportosítás 6 főszempont [18] alapján a 3. ábrán látható.



3. ábra IDS-ek csoportosítása

A következőkben a 3. ábrán bemutatott csoportosítás alapján szeretném részletesen bemutatni az IDS típusok legfontosabb jellemzőit.

3.1.1 Érzékelési szempontok alapján

Az érzékelés szempontjából az alábbi két csoportra oszthatjuk őket:

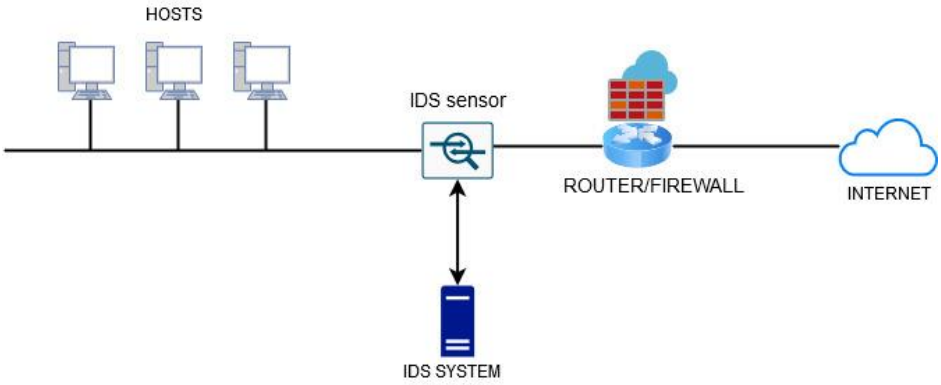
Rendellenesség vagy anomália alapú IDS (Behavior based IDS - BIDS) statisztikai alapon működnek. Megtanulják mind a normál viselkedést, mind a rendszer és a felhasználók viselkedését. A megszerzett ismeretek alapján határozzák meg, hogy az elemzett tevékenység káros-e a rendszerre vagy sem. Gyors és dinamikus alkalmazkodási képesség jellemzi ismeretlen támadástípusok esetén. A BIDS statisztikákat készít a bejelentkezési időről, a fájlok módosításának és mozgatásának gyakoriságáról.

Adatbázis alapú, **másnéven tudás alapú IDS** (Knowledge-based IDS - KIDS) a tárolt minták alapján az IDS eldönti, hogy a megfigyelt tevékenység potenciális támadási kísérletnek minősül-e vagy sem. Jelenleg ez a legszélesebb körben használt IDS-modell. Előnye, hogy a tárolt mintáknak köszönhetően lényegesen kevesebb forgalom kerül feketelistára, mint a BIDS esetében, valamint a riasztási jelzések szabványosítottak és a rendszergazdák számára könnyen értelmezhetőek. A KIDS hátránya, hogy adatbázisa folyamatos frissítést és karbantartást igényel, valamint nem ismeri fel az új támadástípusokat [19].

3.1.2 A védelem típusa alapján

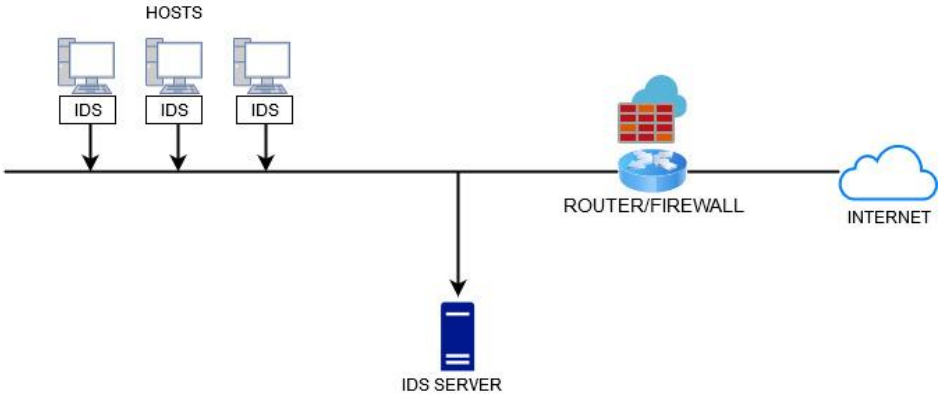
A védelem tárgya alapján háromféle IDS-t különböztethetünk meg:

Hálózati IDS (Network IDS) általában egy hálózati megfigyelő eszközt tartalmaz, ami mögött egy hálózati interfész kártya dolgozik. Ez az IDS típus a hálózat egy szegmensében vagy annak határa mentén helyezkedik el, és vizsgálja a hálózati forgalmat (lásd 4. ábra). Képes egy, vagy akár több rendszert és eszközt is megfigyelni a hálózaton belül, és védeni a hálózatot a támadások ellen. A hálózati IDS-eken belül található egy altípus, a vezeték nélküli IDSek (Wireless IDS - WIDS). A vezeték nélküli hálózat sebezhetőbb a támadásokkal szemben, mint a vezetékes hálózatok, mivel infrastruktúrájuk dinamikus, nagy lefedettséget és korlátlan hozzáférést biztosít [20].



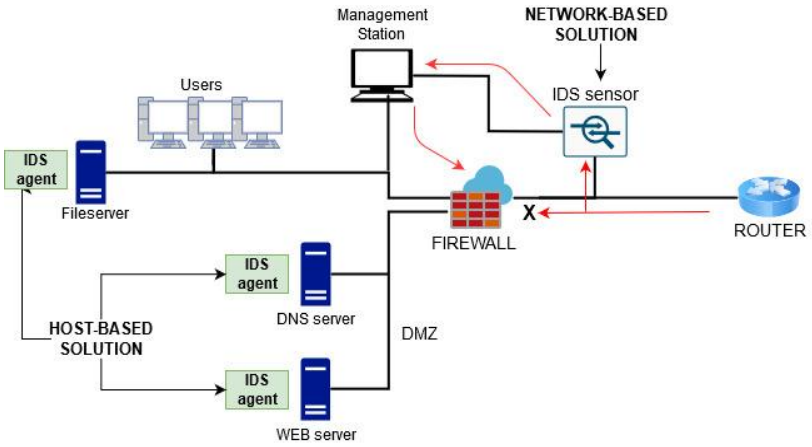
4. ábra Network IDS

Hoszt alapú IDS (Host IDS) egy önálló számítógép megfigyelésére szolgál. Telepíteni és konfigurálni kell az adott gépre. A HIDS-nek szüksége van kisebb, beleépített vizsgáló mechanizmusokra, amelyek az adott rendszer naplófájljaiból szerzi be a szükséges információt a behatolási kísérletek elleni fellépéshez, majd ezt küldheti is akár egy központi IDS rendszerhez (lásd 5. ábra). Képes a rendszert fenyegető hálózati és fizikai támadások jelzésére és kivédésére is egyaránt [21].



5. ábra Host IDS

Hybrid IDS (HYDS). A hostalapú rendszer egyik előnye, hogy csak annak a készüléknek küldött forgalmat kell kezelnie, amelyen fut. A hálózati alapú megoldások néha problémákat okoznak a sok forgalom kezelésével, és nehéz lehet kezelni, amikor a gazdagépen belüli IDS házirendeket kell meghatározni. A hostalapú rendszerek fő hátránya azonban, hogy minél több van, annál nehezebb őket kezelni. Ezért egy jó vállalati megoldásnak általában host és hálózati alapú IDS-megoldások keveréke is van. Az érzékelőket a hálózat kerületén és gerincén, valamint más kulcsfontosságú hozzáférési pontokon használják (lásd 6. ábra). Ez széles spektrumú lefedettséget biztosít az egész hálózat számára. A központi szoftvert ezután telepítik a fő gazdagépekre, hogy extra védelmet biztosítsanak a gazdagépen futó alkalmazások számára [22].



6. ábra Hybrid IDS

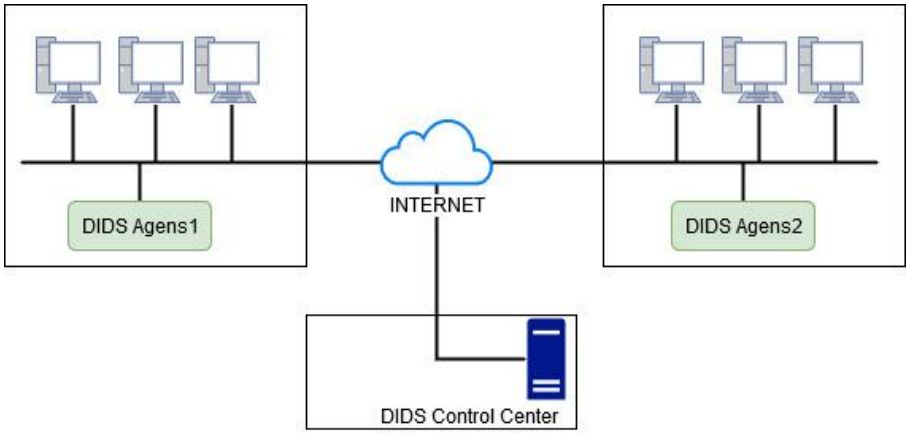
3.1.3. Struktúra szerint

A nagyvállalatok gyakran szembesülnek azzal a problémával, hogy nehéz megfelelő IDS-t létrehozni. A legnagyobb kihívást az jelenti, hogy az egyes észlelőrendszerek földrajzilag nagy távolságban helyezkedhetnek el. Először is el kell dönteni, hogy milyen típusú legyen a kapcsolat közöttük, és milyen hierarchikus struktúrát kell kiépíteni. Ezután meg kell határozni az információ- és parancsáramlást, és a végső kérdés az, hogy az IDS-infrastruktúrát központilag irányítjuk-e, vagy elosztott megközelítést választunk.

Központosított IDS-k (Centralized IDS - CIDS) Az IDS-ek általában a számítógépes hálózat egyes csomópontjaira telepített ügynökalkalmazásokat használnak. Ezeket a csomópontokat monitorcsomópontoknak nevezik. A monitorcsomópont megvizsgálja a hálózati forgalmat. Kétféle üzemmódban, azaz normál és promiszkuózus módban figyelhet. Normál figyelési módban a monitorcsomópont értelmezi és továbbítja az adott belső alhálózatra küldött adatcsomagokat, miután feldolgozta (kiértékelte) azokat. A promiszkuózus figyelési módban a figyelő csomópont minden üzenetet a rendeltetési helytől függetlenül vizsgál meg.

A központosított IDS-ek központi szoftverrel rendelkeznek a hálózat egyik szerverén, amely alkalmazás felelős az elemzésért, észlelésért, osztályozásért és a cselekvésért [23]. A központosított megközelítés előnye, hogy az elosztott rendszerhez képest alacsonyabbak a költségek, emellett a karbantartási és adminisztrációs költségek is alacsonyabbak. Továbbá a teljes hálózati architektúra egyszerűsödik, és így a szervezet biztonsági infrastruktúrájában a sebezhetőségek száma is csökken. Ezen túlmenően a CIDS-ek kezelői képesek a rendszerek felügyeletére és értékelésére a vállalat egész hálózatán.

Elosztott IDS (Distributed IDS- DIDS) számos behatolás jelző rendszert tartalmaz. Hálózatot képeznek és kommunikálnak egymással vagy egy központi szerverrel (lásd 7. ábra). Ennek a megoldásnak számos előnye van a centralizálthoz képest. Először is a támadási adatok nyomon követése, elemzése és feldolgozása könnyebbé válnak. Ezen túlmenően a DIDS képes felismeri a támadási jeleket az egész hálózaton, ahol az egyes szegmensek a vállalaton belül egymástól földrajzilag távol helyezkednek el, vagy akár különböző időzónákban is lehetnek.
A DIDS lehetővé teszi a behatolás korai észlelését, ami a behatolás blokkolását eredményezheti és a teljes hálózatba bejövő forgalom megakadályozását meghatározott IP-címekről, de képesek a vállalaton belüli támadások azonosítására. Továbbá, ha egy fenyegetést azonosítanak egy szegmensben, akkor nem lesz szükség a további elemzésre más szegmensekben [24].



7. ábra Elosztott IDS

3.1.4 Forrás típusa szerint

Az adatok származhatnak különböző forrásokból, például ellenőrzési naplókból, hálózati forgalomból vagy rendszerelemzésből, így adatforrás tekintetében az alábbi kategóriákra lehet bontani az IDS elemzését:

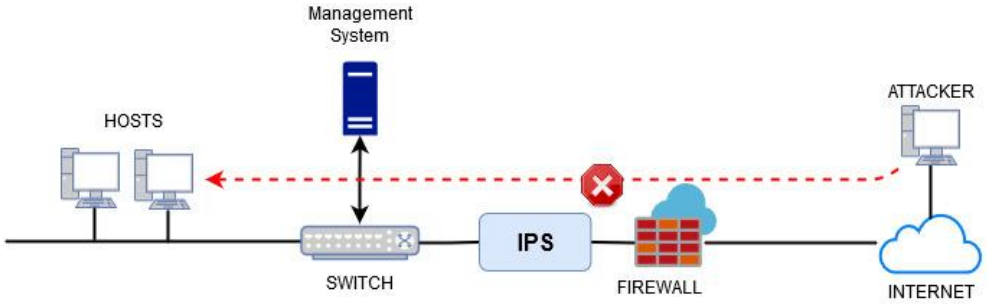
A **naplófájlok** információkat tartalmaznak a rendszerről, a rendszer és az alkalmazási folyamatok tevékenységéről, valamint a felhasználói tevékenységekről. Az IDS-ek az ismert behatolási kísérleteket a felhasználói viselkedés sorozataként modellezik. Ezeket a viselkedéseket az esemény nyomkövetési eseményeként modellezik. Az IDS felelős annak meghatározásáért, hogy az azonosított felhasználói viselkedés hogyan alakul ki a naplófájlokban [25].

Hálózati csomag vizsgálatkor az IDS a csomagokat mélyrehatóan megvizsgálhatja a hálózati forgalomban, és az eredmények alapján engedélyezheti vagy megtilthatja azok haladását, mindezt valós időben. Így minden gyanús esemény vagy csomag azonnal blokkolva van. Ezért a hálózati csomag alapú IDS előnye, hogy a teljes csomag ellenőrzése miatt az új támadásokat is blokkolhatja, amelyek ellen a tűzfal eredeti konfigurációja nem tudna védelmet nyújtani. Így a hálózati csomag alapú IDS képes megállítani azokat a támadásokat, amelyeket a tűzfal nem.

Rendszervizsgálatkor a rendszertámadás a rosszindulatú szoftverek telepítésével befolyásolhatja a rendszer működését, ezért a nem kívánt folyamatok jelenléte a rosszindulatú tevékenységek jele is lehet, és így információt szolgáltathat az IDS-k számára. Ennek a megközelítésnek a határait az adja, hogy a fejlett rosszindulatú programokat úgy fejlesztették ki, hogy a háttérben rejtve működjenek, és ezeket nehéz felismerni.

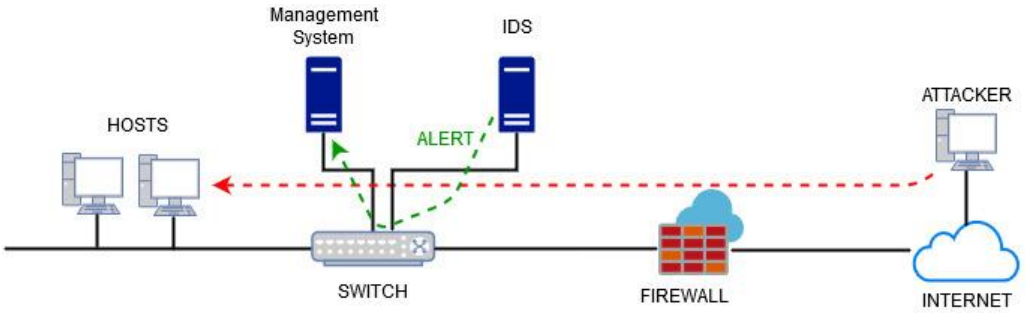
3.1.5. A támadás észlelését követően

Aktív IDS - a behatolás megelőző rendszerként (Intrusion Prevention System – IPS) ismert aktív IDS emberi beavatkozás igénye nélkül, automatikusan blokkolja a gyanúsnak vélt rendszerhozzáférési kísérleteket. Az IPS-t a hálózat határain kell elhelyezni, aminek következtében maga az IPS is érzékennyé válik a támadásokra (lásd 8. ábra). Még az is megtörténhet, hogy saját tevékenységét véli illetéktelen behatolásnak.
Az IPS megfelelő konfiguráció hiányában könnyen tilthatja a rendszer használatára felhatalmazott felhasználókat és alkalmazásokat is. Az IPS típusú megoldás érzékenyebb egy memória túlterhelést irányzó támadásra (Denial of Service – DoS), mint egy passzív IDS. A DoS támadás különböző hálózati címekről indít kérelmeket a rendszer felé egészen addig, amíg a rendszer memória puffere túl nem terhelődik. Az IPS ugyan képes ennek kivédésére, viszont mellékhatásként letilthatja az adott port-ot, vagy akár a teljes hálózati forgalmat is [26].



8. ábra Aktív IDS (IPS)

Passzív IDS - a passzív IDS nem képes automatikus válaszlépésekre, csak a háttérben működve vizsgál (lásd 9. ábra), és támadásgyanús esetben riasztja a rendszergazdát. Előnye, hogy mivel csak passzív megfigyelő a hálózatban, ezért nem válik támadás célpontjává, és az a veszély sem fenyegeti, hogy saját tevékenységét érzékelje támadásként. Hátránya, hogy mire a rendszergazda megkapja az értesítést, elemzi azt, majd döntést hoz a válaszlépésről, addigra nagy valószínűséggel a támadás már lezajlott [27].



9. ábra Passzív IDS

3.1.6. Az elemzés idejét tekintve

Az informatikai rendszerek többsége éjjel-nappal **folyamatosan működik**. Valós idejű védelemre van szükség a maximális rendelkezésre állás biztosítása érdekében. Az ebbe a kategóriába tartozó IDS-k többsége hálózati forgalom elemzésén alapul. Általában a szállítási réteg csomagok fejlécet figyelik, amelyek tartalmazhatnak IP-címeket, TCP/IP flag-eket stb. Egy másik megközelítés elemzi az alkalmazás réteg kommunikációját (FTP, HTTP stb.).
Itt fontos szempont, hogy a csomagok tartalma megfelel-e a protokollnak. Előnyük, hogy folyamatos védelmet és magasabb szintű rendelkezésre állást tudnak biztosítani az IT-rendszerben, mint az **intervallum alapú** rendszerek. Hátrányuk, hogy óriási számítási kapacitást és munkamemóriát igényelhetnek. Az alkalmazott algoritmusok néha nem elég gyorsak és hatékonyak, ami csomagvesztést okozhat.

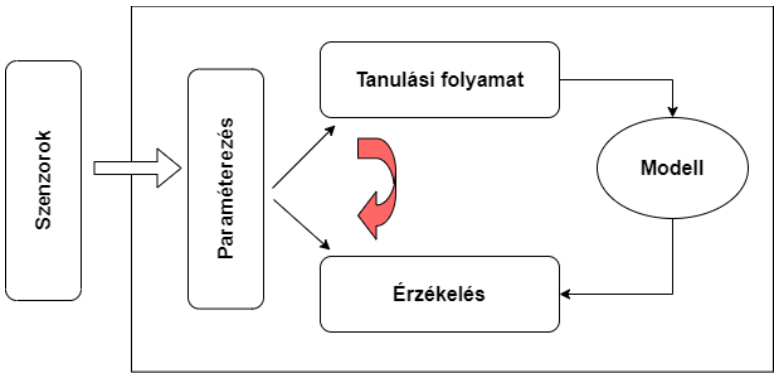
A fejezetben leírt, az IDS rendszerekről összefoglaló leíráshoz tartozó publikációm: [S15]

3.2. Anomália alapú IDS rendszerek

A különböző IDS típusok közül az érzékelési szempontok alapján a rendellenesség alapú IDS-ek azok, melyek működéséhez gépi tanulási módszerek, osztályozó algoritmusok kapcsolódnak, ezért került a kutatásom fókuszába ez a típus.

Bár vannak eltérések a rendellenességfelismerésen alapuló IDS altípusai között, de általában kétféle módban működhetnek (lásd 10. ábra). Tanulási mód esetén az érzékelők által küldött adatokkal tápláljuk, amelyek leírják a tipikus (normál) hálózatot és a felhasználói viselkedést. Ezt a bemenetet a paraméterező modul átalakítja és formalizálja az úgynevezett normál viselkedési profilokba. Ezen profilok alapján a tanulási modul automatikusan, manuálisan vagy kombináltan létrehozza a hálózat normál viselkedési modelljét.

Ezután a BIDS-et átkapcsoljuk észlelési módba, ahol a fő cél, hogy a hálózati biztonság megerősítésére használják. Érzékelési módban a tényleges érzékelő adatokat a paraméterező modul a tényleges profilba konvertálja, és az érzékelő modul összehasonlítja azokat az előzőleg létrehozott modellel. A megszerzett ismeretek alapján az észlelési modul arra törekszik, hogy meghatározza, vajon az elemzett tevékenység káros-e a rendszerre vagy sem [28].



10. ábra BIDS üzemmódjai

A BIDS-ek statisztikákat készítenek a bejelentkezési időről, a felhasználó bejelentkezési idejéről, az általában hozzáférhető fájlokról, valamint a fájlok módosításának és mozgatásának gyakoriságáról stb. Az anomália alapú megközelítés előnye a gyors és dinamikus adaptációs képesség ismeretlen támadástípusokhoz.

A BIDS-ek hátránya, hogy a téves riasztások magas aránya miatt gyakrabban riasztják a rendszergazdát és gyakrabban tesznek ellenintézkedéseket, mint a tudásalapú IDS-ek [29]. Ezenkívül a BIDS kevésbé hatékony olyan rendszerek esetén, amelyek viselkedési mintái nem elég statikusak statisztikák létrehozásához, vagy olyan rendszerek esetén, ahol a felhasználói tevékenység nem monoton. Külön óvintézkedéseket kell tenni a tanulási időszak során, hogy elkerülhető legyen a tényleges behatolás normál viselkedésként történő „megtanulása”.

A BIDS az adatfeldolgozási mód alapján három fő kategóriába sorolható, statisztikai, tudás-alapú és számítási intelligencia alapú.

Statisztikai alapú BIDS-ek

A statisztikai alapú BIDS-ek figyelemmel kísérik a hálózati forgalmat, megvizsgálják a kommunikációs sebességet, a használt protokollokat és az IP-címeket, összpontosítva az engedélyezett értékektől való eltéréseket. Megvizsgálják az aktuális profilt, összehasonlítva a normál profillal. Ha rendellenesség fordul elő a hálózatban, akkor a rendellenességet is kiszámítják. Ha meghaladja az előre meghatározott küszöböt, riasztás aktiválódik. Ez az IDS-típus különféle tevékenységeket követnek és statisztikai módszerekkel jelentéseket készítenek a rosszindulatú tevékenységekről. Hiánya az, hogy a normál forgalom során bekövetkező támadást normál tevékenységnek tekintik [30].

Tudás alapú BIDS-ek

A tudásalapú BIDS-ek egy olyan adatbázissal dolgoznak, amely a korábbi támadásokkal kapcsolatos információkat tartalmazza. E minták alapján dől el, hogy a megfigyelt tevékenység ellenségesnek minősül-e vagy sem. Valójában ez a leggyakrabban használt BIDS-modell. Előnye, hogy lényegesen kevesebb forgalmat sorol tévesen a feketelistára, mint a többi megközelítés. Továbbá szinte minden esetben szabványos riasztásokat használ, amelyek a rendszergazdák számára könnyen érthetőek.

Hátrányának tekinthető azonban, hogy a mintaadatbázis folyamatos frissítésére és karbantartására van szükség. Ezenkívül ez a fajta IDS nem képes felismerni az új támadástípusokat. Ez a hiányosság azt eredményezheti, hogy egy új, korábban ismeretlen támadástípus fehér listára kerülhet, és hozzáférhet a rendszerhez [31].

Számítási intelligencia alapú BIDS-ek

A számítógépes intelligencia (CI) alapú BIDS-ek önállóan vagy emberi segítséggel, mintaadatok alapján felismerhetik és meghatározhatják a szabályokat és szabályszerűségeket. Nemcsak mintákat tanulnak, hanem korábban nem látott adatok esetén is képesek önállóan döntéseket hozni. A CI alapú BIDS-ek statisztikai elemzési technikákat is alkalmaznak teljesítményük javítására. A feldolgozandó hatalmas adatmennyiség miatt azonban általában nagy számítási erőforrásokat igényelnek.

Ebben az alfejezetben található leírásról a publikációm: [S11]

4. Adathalmaz feldolgozás

Az adathalmaz feldolgozása során, különösen nagydimenziós esetekben, kiemelkedően fontos a megfelelő előfeldolgozás és dimenziócsökkentés alkalmazása. A nagydimenziós adatok gyakran tartalmaznak sok felesleges vagy zajos információt, ami negatív hatással lehet az analízis pontosságára és hatékonyságára. Az előfeldolgozási lépések, mint például a hiányzó adatok kezelése, az outlier értékek azonosítása és kezelése, valamint a normalizálás vagy skálázás segítenek tisztább és megbízhatóbb adatok létrehozásában. Emellett a dimenziócsökkentés technikái lehetővé teszik, hogy a nagy mennyiségű változót kevesebb, de lényeges dimenzióban jelenítsük meg az adatokat. Ez javítja az értelmezhetőséget, csökkenti a zajt, és segít az analízis és modellezés hatékonyságának növelésében.

Az IDS rendszerek tanítására szolgáló adathalmazok kiemelkedő fontosságúak a kiberbiztonsági alkalmazások fejlesztésében. Ezek az adathalmazok általában tartalmazzák a hálózati forgalom és rendszeresemények rögzített adatait, beleértve a normális és potenciálisan káros tevékenységeket is. A tanítóadathalmazok lehetővé teszik az IDS rendszerek számára, hogy megtanulják az olyan káros viselkedésmintákat, mint a kibertámadások vagy a hálózati behatolási kísérletek, és képesek legyenek azokat megkülönböztetni a normális forgalomtól. Az adathalmazok sokfélesége és mérete lehetővé teszi a modellek széles körű tesztelését és finomhangolását, hogy a rendszer minél hatékonyabban reagálhasson az új és változó kiberbiztonsági fenyegetésekre.

4.1. IDS-ek tanítására használható adathalmazok

Az első IDS-ek tanítására használt mintaadatkészlet a KDD'99 [32] volt, amely később számos IDS megoldás kifejlesztésének kiindulópontjául szolgált. Ez normál tevékenységeknek és számos támadástípusnak (DOS, guesspassword, buffer overflow, remote FTP, synflood, Nmap, rootkit) megfelelő szimulált forgalomról tartalmaz információkat. Ezt követően jónéhány más adathalmaz is készült, amelyek új támadástípusok mintáit, valamint néhány további jellemzőt tartalmaznak.

Egy jól használható adathalmaznak tartalmaznia kell az alábbiakat [33]:

- Teljes hálózati konfiguráció: egy modellezett hálózat minden olyan eszközt tartalmazzon, melyek valós rendszerek részei is (Switch, Router, Tűzfal),
- Különböző operációs rendszerek legyenek jelen (Windows, Linux, MacOS),
- Tartalmazzon teljes forgalmat (áldozat és támadó hálózatok),

- Címkézett adatkészlet, hogy a tanítás során meg lehessen állapítani a normál és a különböző támadás során fellépő forgalmakat,
- Teljes adatforgalom felvétel történjen (mirror port),
- Az ismert elérhető protokollok szerepeljenek (HTTP, HTTPS, FTP, SSH),
- A lehető legtöbb és legkülönbözőbb támadások szerepeljenek,
- A forgalmak különböző helyekről való rögzítése (Switch, Memory Dump, PC),
- Funkcionalitás (PCAP analyzer, CSV generator),
- Metaadatok (time, attacks, flows and labels).

Az elmúlt 25 évben történt számos olyan adathalmaz kialakítás, melyek az IDS-ek tanításához jól használhatóak. Sok olyan adathalmaz is van, ami nem tartalmaz támadásokat, vagy épp nem címkézett. Összegyűjtésre került néhány olyan adathalmaz, amelyek számos tudományos kutatás alapját képezik, és amelyek kifejezeten támadási adatokat is tartalmaznak, ezek az 1. táblázatban találhatóak. Fontos paramétere az adathalmazoknak, hogy milyen mennyiségű jellemzőt tartalmaznak, melyek a feldolgozást és annak eredményét tudják meghatározni (lásd 2. táblázat).

1. táblázat Tanító adathalmazok az elmúlt 25 évben

Évszám	Adathalmaz	Támadási típusok
2018	CSE-CIC-IDS2018 on AWS Canadian Institute for Cybersecurity [34]	Bruteforce attack, DoS attack, Web attack, Infiltration attack, Botnet attack, DDoS+PortScan
2017	CICIDS2017 Canadian Institute for Cybersecurity [33] [35] [36]	botnet (Ares), cross-site-scripting, DoS (Hulk, GoldenEye, Slowloris, Slowhttptest), DDoS (LOIC), heartbleed, infiltration, SSH brute force, SQL injection
2017	CIDDS Coburg University, Germany [37] [38]	DoS, port scans (ping-scan, SYN-Scan), SSH brute force, port scans (ACK-Scan, FIN-Scan, ping-Scan, UDP-Scan, SYN-Scan)
2016	UGR'16 Universidad de Granada, Spain [39]	botnet (Neris), DoS, port scans, SSH brute force, spam, UDP Scan, SSH Scan

2015	TUIDS Tezpur University, India [40]	botnet (IRC), DDoS (Fraggle flood, Ping flood, RST flood, smurf ICMP flood, SYN flood, UDP flood), port scans (FIN-Scan, NULL-Scan, UDP-Scan, XMAS-Scan), coordinated port scan, SSH brute force
2015	UNSW-NB15 ACCS Cyber Range Lab, Australia [41]	backdoors, DoS, exploits, fuzzers, generic, port scans, reconnaissance, shellcode, spam, worms
2013	ADFA Australian Defence Force Academy University of New South Wales [42] [43] [44]	Password bruteforce, Add new superuser, Java Based Meterpreter, Linux Meterpreter Payload, C100 Webshell
2012	ISCXIDS2012 University of New Brunswick [45]	FTP és SSH password bruteforce, Java based Meterpreter, Linux Meter-preter payload and C100 Webshel attack vectors
2010	HTTPCSIC2010 Spanish Research Nationa [46]	SQL injection, buffer overflow, information gathering, files disclosure, CRLF injection, XSS
2009	Kyoto KyotoUniversity [47]	Trojan.Fakealer, Trojan.Agent, HTML.Phishing.Bank, Trojan.Goldun, Trojan.Goldun
2009	CDX UnitedStatesMilitaryAcademy [48]	Buffer Overflow
2000	DEFCON The ShmooGroup [49] [33]	DEFCON -8: port scanning and buffer overflow attacks, DEFCON-10: port scan and sweeps, bad packets, administrative privilege, FTP by Telnet protocol attacks
1998	KDD'99 University of California, Irvine [32] [50]	DoS (back, land, neptune, pod, smurf, teardrop), Probe (satan, ipsweep, nmap, portsweep) R2L(Guess_psw, ftp_write, imap, phf, multihop, warezmaster, warezclient, spy) U2R (Buffr_owerflow, loadmodule, pearl, rootkit)

2. táblázat Adathalmazok jellemző száma

Adathalmaz	jellemzők száma	Adathalmaz	jellemzők száma
CSE-CIC-IDS2018	80	ISCXS2012	19
CICIDS2017	80	HTTPCSIC2010	18
CIDDS	14	Kyoto	24
UGR'16	34	CDX	5
TUIDS	34	DEFCON	na
UNSW-NB15	49	KDD'99	41
ADFA	42		

4.2. A kutatás során vizsgált adathalmaz

Az ebben a tanulmányban ismertetett kutatás során használt adathalmaz a CSE-CIC-IDS2018 on AWS [34], amelyet a Canadian Institute for Cybersecurity laboratórium hozott létre. Ez az adathalmaz azért lett kiválasztva, mert a kutatásom kezdeti szakaszában ez volt a legfrissebb adathalmaz, a kutatásban szereplő támadásokat tartalmazza, és megfelel a kutatáshoz szükséges összes kritériumnak (pl. teljes forgalom, címkézés stb.). Az adathalmazban található végrehajtott támadások listája és időtartama a 3. táblázatban található.

3. táblázat CSE-CIC-IDS2018 adathalmaz alkotóelemei

Támadás típusa	Támadás eszközei	Időtartam	Támadó	Áldozat
Bruteforce attack	FTP – Patator SSH – Patator	1 nap	Kali linux	Ubuntu 16.4 (Web Server)
DoS attack	Hulk, GoldenEye, Slowloris, Slowhttptest	1 nap	Kali linux	Ubuntu 16.4 (Apache)
DoS attack	Heartleech	1 nap	Kali linux	Ubuntu 12.04 (Open SSL)
Web attack	Damn Vulnerable Web App (DVWA) In-house selenium framework (XSS and Brute-force)	2 nap	Kali linux	Ubuntu 16.4 (Web Server)
Infiltration attack	First level: Dropbox download in a windows machine Second Level: Nmap and portscan	2 nap	Kali linux	Windows Vista and Macintosh
Botnet attack	Ares (developed by Python): remote shell, file upload/download, capturing screenshots and key logging	1 nap	Kali linux	Windows Vista, 7, 8.1, 10 (32-bit) and 10 (64-bit)
DDoS+PortScan	Low Orbit Ion Canon (LOIC) for UDP, TCP, or HTTP requests	2 nap	Kali linux	Windows Vista, 7, 8.1, 10 (32-bit) and 10 (64-bit)

A hálózati forgalom minden egyes rekordja 80 attribútumból épül fel, amelyeket a rögzített forgalomból a CICFlowMeter-V3 [51] segítségével nyertek ki. A CICFlowMeter-V3 kétirányú áramlások generálására használható, ahol az első csomag határozza meg az előre (forrásból a célba) és a hátra (célból a forrásba) irányt, így több mint 80 statisztikai hálózati forgalmi jellemző, mint például az időtartam, a csomagok száma, a bájtok száma, a csomagok hossza stb. külön-külön kiszámítható az előre és a hátra irányban. A további funkciók közé tartozik a jellemzők kiválasztása a meglévő jellemzők listájából, új jellemzők hozzáadása és az áramlási időkorlát időtartamának ellenőrzése. Az alkalmazás kimenete CSV formátumú fájl, amely hat oszlopot tartalmaz minden egyes áramláshoz (FlowID, SourceIP, DestinationIP, SourcePort, DestinationPort és Protocol), hálózati forgalomelemzési jellemzővel.

4. Táblázat CSE-CIC-IDS2018 adathalmazban lévő jellemzők listája

#	Jellemző	#	Jellemző	#	Jellemző
0	*Dst Port*	27	*Bwd IAT Tot*	54	*Pkt Size Avg*
1	*Protocol*	28	*Bwd IAT Mean*	55	*Fwd Seg Size Avg*
2	*Timestamp*	29	*Bwd IAT Std*	56	*Bwd Seg Size Avg*
3	*Flow Duration*	30	*Bwd IAT Max*	57	*Fwd Byts/b Avg*
4	*Tot Fwd Pkts*	31	*Bwd IAT Min*	58	*Fwd Pkts/b Avg*
5	*Tot Bwd Pkts*	32	*Fwd PSH Flags*	59	*Fwd Blk Rate Avg*
6	*TotLen Fwd Pkts*	33	*Bwd PSH Flags*	60	*Bwd Byts/b Avg*
7	*TotLen Bwd Pkts*	34	*Fwd URG Flags*	61	*Bwd Pkts/b Avg*
8	*Fwd Pkt Len Max*	35	*Bwd URG Flags*	62	*Bwd Blk Rate Avg*
9	*Fwd Pkt Len Min*	36	*Fwd Header Len*	63	*Subflow Fwd Pkts*
10	*Fwd Pkt Len Mean*	37	*Bwd Header Len*	64	*Subflow Fwd Byts*
11	*Fwd Pkt Len Std*	38	*Fwd Pkts/s*	65	*Subflow Bwd Pkts*
12	*Bwd Pkt Len Max*	39	*Bwd Pkts/s*	66	*Subflow Bwd Byts*
13	*Bwd Pkt Len Min*	40	*Pkt Len Min*	67	*Init Fwd Win Byts*
14	*Bwd Pkt Len Mean*	41	*Pkt Len Max*	68	*Init Bwd Win Byts*
15	*Bwd Pkt Len Std*	42	*Pkt Len Mean*	69	*Fwd Act Data Pkts*
16	*Flow Byts/s*	43	*Pkt Len Std*	70	*Fwd Seg Size Min*
17	*Flow Pkts/s*	44	*Pkt Len Var*	71	*Active Mean*
18	*Flow IAT Mean*	45	*FIN Flag Cnt*	72	*Active Std*
19	*Flow IAT Std*	46	*SYN Flag Cnt*	73	*Active Max*
20	*Flow IAT Max*	47	*RST Flag Cnt*	74	*Active Min*
21	*Flow IAT Min*	48	*PSH Flag Cnt*	75	*Idle Mean*
22	*Fwd IAT Tot*	49	*ACK Flag Cnt*	76	*Idle Std*
23	*Fwd IAT Mean*	50	*URG Flag Cnt*	77	*Idle Max*
24	*Fwd IAT Std*	51	*CWE Flag Count*	78	*Idle Min*
25	*Fwd IAT Max*	52	*ECE Flag Cnt*	79	*Label*

4.2.1. A kiválasztott támadástípusok

A Brute Force támadás egy olyan kiberbiztonsági technika, amely során a támadó minden lehetséges kombinációt kipróbál a támadási célpont felépítésének vagy biztonsági intézkedéseinek feltörésére. Ez általában akkor alkalmazható, amikor nincs könnyen kihasználható sebezhetőség vagy gyenge pont a rendszerben, de a támadó továbbra is illegális módon szeretne hozzáférni az adott rendszerhez, fiókhoz vagy adathoz. Ezzel a módszerrel a támadó próbálkozik véletlenszerűen a jelszó felülírásával, mivel minden lehetséges kombinációt végig próbál. A Brute Force támadások rendkívül időigényesek lehetnek, különösen akkor, ha a célpont biztonsági intézkedései erős jelszavakat, lekérdezések számának korlátozását vagy más védekezési mechanizmusokat alkalmaznak. Ennek ellenére, amennyiben a támadó rendelkezik elegendő erőforrással és idővel, a Brute Force támadás hatékony lehet.
Az SQL injection (Structured Query Language injection) egy olyan kibertámadás, amely során a támadó manipulálja vagy befolyásolja az adatbázis lekérdezéseket egy alkalmazás vagy weboldal segítségével. Az SQL injection támadások akkor történnek, amikor a webalkalmazás nem megfelelően kezeli a felhasználóktól érkező bemeneti adatokat, és a támadó kihasználja ezt a sebezhetőséget azáltal, hogy rosszindulatú SQL lekérdezéseket injektál a rendszerbe.

4.2.2. A kiválasztott adatok

Az adatkészlet valójában több fájlból áll. A vizsgálatra kiválasztott fájlokat és az általuk lefedett támadástípusokat a 5. táblázat tartalmazza. Az előfeldolgozás ezen fájlok összevonásával kezdődött.

5. táblázat A vizsgálathoz kiválasztott fájlok

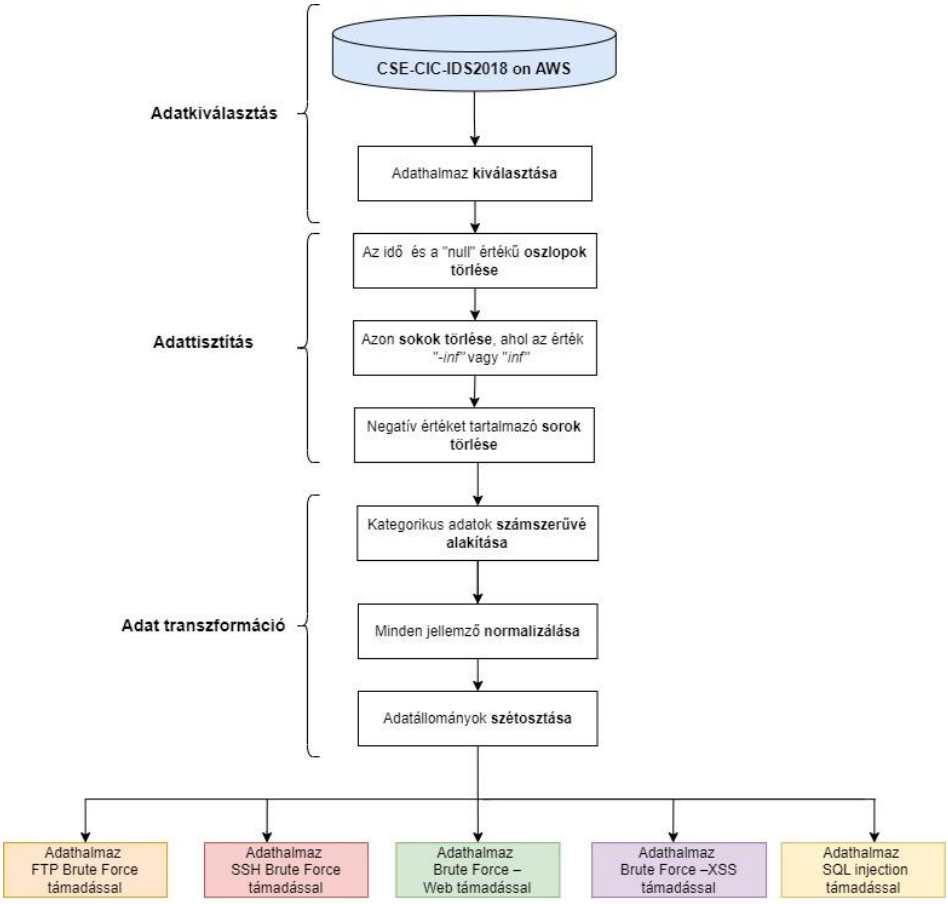
Fájlnév	Támadás típusa
Wednesday-14-02-2018 TrafficForML CICFlowMeter.csv	FTP-BruteForce SSH-BruteForce
Thursday-22-02-2018 TrafficForML CICFlowMeter.csv	Brute Force–Web Brute Force –XSS SQL Injection
Friday-23-02-2018 TrafficForML CICFlowMeter.csv	Brute Force –Web Brute Force –XSS SQL Injection

Az előfeldolgozási munkafolyamatot a következő alfejezetben mutatom be.

4.3. Dimenziócsökkentés

Az adatcsökkentési szakasz a jellemzők kiválasztására és a dimenziócsökkentésre összpontosít, ami számos előnnyel járhat. Az egyik legfontosabb előny az, hogy számos adatbányászati algoritmus jobban működik, ha a dimenziók száma - az adatok attribútumainak (oszlopainak) száma - kisebb. Ez részben azért van így, mert a dimenziócsökkentés kiküszöböli az irreleváns attribútumokat és csökkenti a zajt. Egy másik előnye, hogy érthetőbb modellhez vezethet, mivel kevesebb jellemző lesz benne. Ezenkívül a csökkentett adatmennyiség kevesebb tárhelyet és kevesebb időt igényel a feldolgozásához. A kutatásomban lévő adathalmaz előfeldolgozásának teljes menete a 11. ábrán látható.

Számos olyan szoftvereszköz létezik, amely nagy adathalmazok előfeldolgozására és elemzésére használható (pl. MS Excel, Matlab, SPSS stb). A kutatás első szakaszában megállapítást nyert, hogy a kiválasztott adathalmaz mérete (jellemzők száma és rögzített adatok mennyisége) miatt nehézkes a feldolgozás a hagyományos szoftverekkel, így az adatfeldolgozáshoz a Python programozási nyelvet használtam, melynek moduljai és könyvtárai gyors és hatékony eredményeket képesek produkálni.



11. ábra Adathalmaz előfeldolgozása

4.3.1. Adattisztítás

Ahhoz, hogy megfelelő adathalmazt hozzunk létre a modell képzéséhez és teszteléséhez, előzetesen fel kell dolgozni a nyers adatokat. Az előfeldolgozási munkafolyamat az adattisztítással kezdődik, amely általában magában foglalja az érvénytelen vagy hiányzó adatokat tartalmazó sorok (rekordok) törlését, az azonos értékű oszlopok törlését (pl. olyan oszlopok, ahol minden érték nulla), az osztályozás szempontjából irrelevánsnak ítélt jellemzők (oszlopok) törlését.

A kutatási művelet során nincs szükség az időparaméterre, és azokra az oszlopokra sem, ahol minden érték "nulla", mivel ezek nem befolyásolják a kimenetet, azaz az utolsó oszlop értékét. Ez a lépés 69 megmaradt oszlopot eredményezett, miután az eredeti 80 oszlopból 11-et töröltem. A következő lépés az érvénytelen értékeket tartalmazó sorok törlése volt. Ezért először az "inf" és "-inf" értékű sorok kerültek törlésre, majd az adatállományban negatív értékkel rendelkező sorok. Ezekkel a műveletekkel az adatállomány tisztítása befejeződött.

4.3.2. Adattranszformáció

A kutatás esetében az adattranszformációs szakasz három műveletet tartalmazott, azaz a kategorikus adatok numerikus adatokká történő átalakítását, normalizálást és az adathalmaz felosztását. A számítások elvégzéséhez az adathalmaz minden nem numerikus elemét számokká kellett átalakítani. Ezt minden kategorikus oszlop esetében úgy végeztem el, hogy minden kategóriához egy számot rendeltem. Például az "FTP-BruteForce" karakterlánc minden egyes előfordulását 1 értékkel helyettesítettem.

Az értékek sorrendje önkényesen lett meghatározva, nem lett figyelembe véve semmilyen fogalmi távolságmérő, főként azért, mert később az adathalmazt olyan részhalmazokra osztottam, amelyek csak az egyik támadástípusnak és a normál forgalomnak megfelelő adatokat tartalmaznak.

4.3.3. Normalizálás

Az adatok normalizálása a gépi tanulásban elterjedt gyakorlat, amely a numerikus oszlopok közös skálára való átalakításából áll. A gépi tanulásban egyes jellemzőértékek többszörösen nagyobbak, mint mások. Így a nagyobb értékekkel rendelkező jellemzők dominálhatnak a tanulási folyamatban. Ez azonban nem jelenti azt, hogy ezek a változók fontosabbak a modell

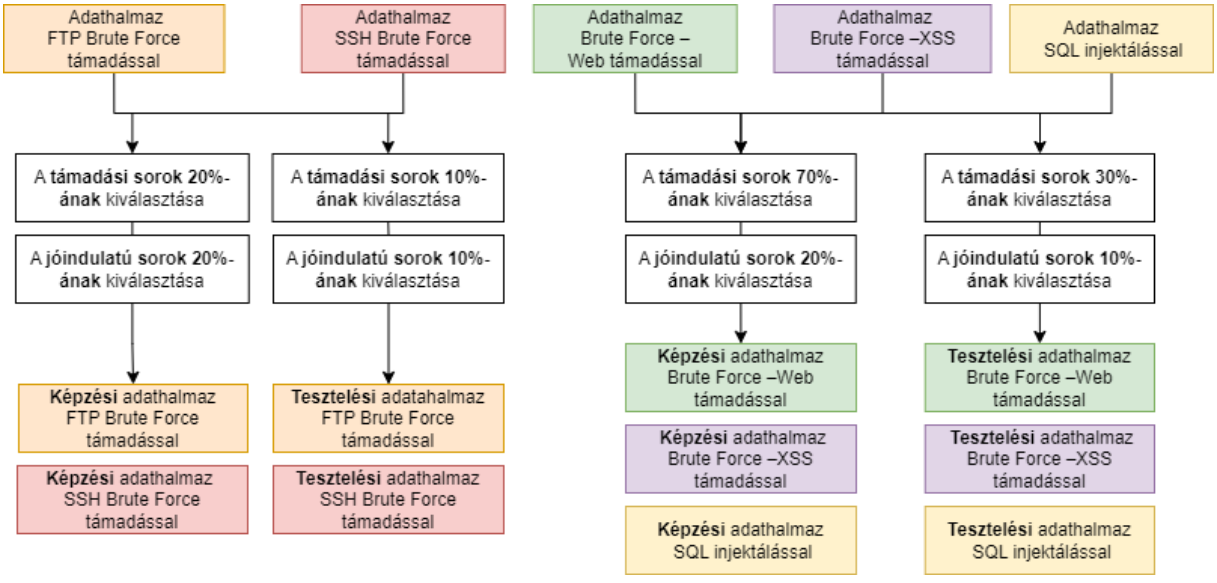
kimenetének előrejelzésében. Az adatok normalizálása a többszörösen skálázott adatokat azonos skálára alakítja át.

A normalizálás után minden változó hasonló hatással van a modellre, ami javítja a tanulási algoritmus stabilitását és teljesítményét. A statisztikában többféle normalizálási technika létezik. A legegyszerűbb és leggyakrabban használt típus a min-max skálázás, amely egy jellemzőt a [0,1] rögzített tartományba skáláz át (lásd 2. képlet) úgy, hogy a jellemző minimális értékét (x_{min}) kivonjuk az aktuális értékből (x), majd az eredményt elosztjuk a tartománnyal.

$$x_{new} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{2}$$

4.3.4. Adatok felosztása

Mind az öt adatkészlet nagyon nagy számú példányt tartalmaz. Ezért a tanító- és tesztminták létrehozásakor az eredeti adatoknak csak egy része lett felhasználva a vizsgálathoz. A tanító- és tesztminták létrehozásának lépéseit a 12. ábra mutatja be.



12. ábra Tanítási és tesztelési halmazok létrehozása

Mind az FTP, mind az SSH Brute Force támadások esetében a gyakorló minták az eredeti példányok 20%-át tartalmazzák. Rétegzett mintavételt alkalmaztam annak biztosítására, hogy minden osztály (támadás és jóindulatú forgalom) képviselve legyen a mintában. Az így kapott

rekordgyűjtemény a támadó sorok 20%-át, a jóindulatú forgalmat leíró sorok 20%-át tartalmazta. A mintavételezés csere nélkül történt.
A tesztmintákat hasonló módon hoztam létre a fennmaradó adatkészletekből kiválasztott rekordok kiválasztásával úgy, hogy az így kapott adatpontok gyűjteménye az eredeti adatpontok 10%-át képviselje.
A Brute Force Web, Brute Force XSS és SQL Injection támadástípusok esetében a kiválasztási folyamat némileg eltérő volt, mivel a rosszindulatú forgalmat leíró rekordok száma nagyon alacsony volt. Ezért minden esetben a támadássorok gyűjteményét két részre osztottam, azaz 70%-ot használtam a képzéshez, a maradék 30%-ot pedig tesztelési célokra. Ezután a gyakorló mintákat a jóindulatú forgalomhoz tartozó adatpontok 20%-ának hozzáadásával hoztam létre. Végül a tesztmintákat úgy állítottam össze, hogy a jóindulatú forgalmi rekordok 10%-át hozzáadtam a tesztelésre kijelölt támadási sorokhoz.

4.4. Gyakorlati megvalósítás

Az adathalmaz feldolgozásra majd az azt követő jellemzőkiválasztási módszerekre a Python programozási nyelv adott hatékony megoldást. A Python egyszerű és olvasható szintaxissal rendelkezik, amely lehetővé teszi a könnyebb kezelhetőséget. Sokoldalú és széles körben használt nyelv, rengeteg külső könyvtára és modulja van, amelyek lehetővé teszik a fejlesztők számára, hogy számos előre elkészített funkciót és eszközt használjanak. Az interpretált jellege azt jelenti, hogy a Python forráskódot nem fordítják gépi kóddá, hanem byte-kód interpreter futtatja azt soronként.
Ez lehetővé teszi, hogy a kód könnyen hordozható legyen különböző operációs rendszereken, anélkül, hogy szükség lenne különálló fordításra. A kutatási munka során az adathalmaz feldolgozására használt Python könyvtár a „*Pandas*”. A Pandas egy hatékony és rugalmas adatkezelési eszköz, amely lehetővé teszi adatok táblázatos formában való kezelését és manipulálását. A „*DataFrame*” lehetővé teszi az adatok könnyű és hatékony feldolgozását [52] [53].
A Pandas DataFrame olyan kétdimenziós táblázatot reprezentál, amelynek oszlopai lehetnek különböző típusú adatok, például szöveg, szám vagy dátum. A DataFrame rendelkezik indexsorokkal és oszlopnevekkel, amelyek segítségével könnyen hivatkozhatunk a táblázat elemeire.

4.5. Eredmények

Az adathalmaz előfeldolgozása során jelentős dimenziócsökkentést értem el, hiszen a 3 kiindulási adathalmaz oszlopszáma 80 volt, és ez redukálódott 69-re. Továbbá az adattranszformáció segítségével támadástípus alapján már könnyedén szét lehetett osztani az adatállományokat különböző fájlokba (lásd 6. táblázat), így a további vizsgálatokat már támadási típusonként folytattam.

6. táblázat Az előfeldolgozás során létrejött adatkészletek

Fájlnév	Sorok száma	Oszlopok száma
dataset-ftp.csv	857,162	69
dataset-ssh.csv	851,397	69
dataset-web.csv	2,085,515	69
dataset-xss.csv	2,085,134	69
dataset-sql.csv	2,085,134	69

Az adatok felosztását követően a gyakorló- és tesztminták létrehozásakor az eredeti adatoknak csak egy része lett felhasználva a vizsgálathoz. Mindkét esetben az így kapott rekordszámok a 7. és a 8. táblázatban találhatóak.

7. táblázat Tanítóhalmazok a dimenziócsökkentés után

Fájlnév	Sorok száma	Oszlopok száma
dataset-ftp-tr.csv	171,433	69
dataset-ssh-tr.csv	170,280	69
dataset-web-tr.csv	417,592	69
dataset-xss-tr.csv	417,211	69
dataset-sql-tr.csv	417,068	69

8. táblázat Teszthalmazok a dimenziócsökkentés után

Fájlnév	Sorok száma	Oszlopok száma
dataset-ftp-ts.csv	85,716	69
dataset-ssh-ts.csv	85,140	69
dataset-web-ts.csv	209,101	69
dataset-xss-ts.csv	208,720	69
dataset-sql-ts.csv	208,577	69

Az adathalmaz előfeldolgozását követően a 9. táblázatban láthatóak a jellemzők sorszáma és a nevük.

9. táblázat Az adathalmaz jellemzőinek listája az előfeldolgozást követően

#	Jellemző	#	Jellemző	#	Jellemző
0	*Dst Port*	23	*Fwd IAT Std*	46	*URG Flag Cnt*
1	*Protocol*	24	*Fwd IAT Max*	47	*ECE Flag Cnt*
2	*Flow Duration*	25	*Fwd IAT Min*	48	*Down/Up Ratio*
3	*Tot Fwd Pkts*	26	*Bwd IAT Tot*	49	*Pkt Size Avg*
4	*Tot Bwd Pkts*	27	*Bwd IAT Mean*	50	*Fwd Seg Size Avg*
5	*TotLen Fwd Pkts*	28	*Bwd IAT Std*	51	*Bwd Seg Size Avg*
6	*TotLen Bwd Pkts*	29	*Bwd IAT Max*	52	*Subflow Fwd Pkts*
7	*Fwd Pkt Len Max*	30	*Bwd IAT Min*	53	*Subflow Fwd Byts*
8	*Fwd Pkt Len Min*	31	*Fwd PSH Flags*	54	*Subflow Bwd Pkts*
9	*Fwd Pkt Len Mean*	32	*Fwd Header Len*	55	*Subflow Bwd Byts*
10	*Fwd Pkt Len Std*	33	*Bwd Header Len*	56	*Init Fwd Win Byts*
11	*Bwd Pkt Len Max*	34	*Fwd Pkts/s*	57	*Init Bwd Win Byts*
12	*Bwd Pkt Len Min*	35	*Bwd Pkts/s*	58	*Fwd Act Data Pkts*
13	*Bwd Pkt Len Mean*	36	*Pkt Len Min*	59	*Fwd Seg Size Min*
14	*Bwd Pkt Len Std*	37	*Pkt Len Max*	60	*Active Mean*
15	*Flow Byts/s*	38	*Pkt Len Mean*	61	*Active Std*
16	*Flow Pkts/s*	39	*Pkt Len Std*	62	*Active Max*
17	*Flow IAT Mean*	40	*Pkt Len Var*	63	*Active Min*
18	*Flow IAT Std*	41	*FIN Flag Cnt*	64	*Idle Mean*
19	*Flow IAT Max*	42	*SYN Flag Cnt*	65	*Idle Std*
20	*Flow IAT Min*	43	*RST Flag Cnt*	66	*Idle Max*
21	*Fwd IAT Tot*	44	*PSH Flag Cnt*	67	*Idle Min*
22	*Fwd IAT Mean*	45	*ACK Flag Cnt*	68	*Label*

A táblázatban szereplő számokkal történik meg a jellemzőkre való hivatkozás a későbbi fejezetekben szereplő eredménytáblázatokban.

5. Jellemzőkiválasztási módszerek és osztályozási algoritmusok irodalom feldolgozása

Az IDS osztályozó modulok mintaadatokon alapuló fejlesztési lehetőségeit az elmúlt évtizedben intenzíven vizsgálták. Ebben a fejezetben a korábbi kutatások eredményei találhatóak, hogy mely jellemzőkiválasztási módszerek és milyen osztályozó algoritmusok használatával történt meg az IDS-ekhez létrehozott adathalmazok feldolgozása.

Kurniabudi és tsa. [54] a CICIDS-2017 adathalmaz jellemzőinek rangsorolására és klaszterezésére az információnyereség (IG) módszerét alkalmazta, majd a jellemzők kiválasztására a Random Forest (RF), Bayes Net (BN), Random Tree (RT), Naive Bayes (NB) és J48 osztályozó algoritmusokat alkalmazta, amelyek jó osztályozási eredményeket hoztak. Rahman és kollégái [55] az AWID-adatkészlet elemzését az SVM (Support Vector Machine) és a C4.5 mint jellemzőválasztási módszerek alkalmazásával végezte, mesterséges neurális hálózatokon (ANN) alapuló osztályozással, 99,95%-os pontosságot elérve.
Javadpour és tsa. [56] Pearson lineáris korrelációt és IG-t használtak a KDD99 adathalmaz jellemzőinek kiválasztásához, valamint a CART, ANN, Decision Tree és Random Forest (RF) algoritmusokat az osztályozáshoz. A legjobb eredményeket (99,98%-os pontosság) a neurális hálózatos módszerrel érték el.
Taher és tsa. munkájukban az NSL-KDD-adatkészlethez korrelációs és Chi-négyzet alapú technikákat használtak jellemzőválasztási módszerként, majd az ANN és SVM osztályozó algoritmusok 94,02%-os felismerési arányt értek el [57].
Kocher és tsa. az UNSW-NB15 adathalmazon a Chi-négyzet megközelítést használta a dimenzionalitás csökkentésére, majd k-Nearest Neighbors (KNN), Stochastic Gradient Descent (SGD), Random Forest, Logistic Regression (LR) és Naive Bayes (NB) algoritmusokat használt az osztályozáshoz, mely 99,64%-os osztályozási pontosságot eredményezett. [58].
Alkasassbeh [59] BayesNet, MLP és SVM gépi tanulási módszereket, valamint IG, ReliefF és Genetic Search módszereket használt a jellemzők kiválasztásához. A legjobb pontosságot (99,9%) a BayesNet és a GS segítségével érték el.
Thaseen és tsa. a Chi-square megközelítést használta az NSL KDD-adatkészlet jellemzőinek kiválasztására, és az osztályozást SVM osztályozóval végezte. A javasolt modell jó felismerést eredményezett sok hamis riasztás nélkül [60].

Awotunde és tsa. összehasonlította az NSL-KDD és az UNSW-NB15 adatkészleteket hibrid szabályalapú jellemzőválasztás és a DFFNN mély tanulási algoritmus segítségével, a felismerési arány 99,0% volt az NSL-KDD és 98,9% az UNSW-NB15 esetében. [61].

Sasan és tsa. a J48 és a CART módszereket alkalmazta az NSL-KDD adathalmazra 29 jellemzőt használva, és 88,23%-os pontosságot ért el. [62]. A cikk azonban nem írja le, hogyan választották ki a 29 jellemzőt.

Biswas bemutatta a jellemzőválasztási módszerek (CFS, IGR, PCA) és osztályozó algoritmusok (NB, SVM, DT, NN, k-NN) összehasonlítását az NSL-KDD adathalmazon, amely azt mutatja, hogy a k-NN osztályozó jobban teljesít a többinél, és a jellemzőválasztási módszerek közül az IGR jellemzőválasztási módszer a legjobb [63].

Shaukat és tsa. a CICIDS-2017 adatkészletet vizsgálta CFS és Naive Bayes feature selection módszerekkel, MLP és IBK algoritmusokkal, ami azt mutatta, hogy az IBK pontosabb, mint az MLP [64].

Malhotra és tsa. több különböző osztályozót használt az NSL-KDD adathalmaz elemzéséhez, amelyek közül a Random Forest, a Bagging, a PART és a J48 volt a legjobb négy a modellépítési idő szempontjából [65].

Krishnaveni és tsa. IG, Chi-square, Gain Ratio, Symmetric Uncertainty és Relief módszereket alkalmaztak a Real-Time Honeypot, NSL-KDD és Kyoto adathalmazok jellemzőinek kiválasztására, valamint SVM, Naive Bayes, Logistic Regression és Decision Tree osztályozó algoritmusokat használtak, és az egyváltozós ensemble-szűrős jellemzőkiválasztási módszert (UEFFS) javasolnak, amivel jobb előrejelzési pontosságot, észlelési arányt és téves riasztási arányt lehet elérni [66].

Kumar és munkatársai az NSL-KDD adathalmaz esetében a CFS, IGF és GR módszereket használták a jellemzők kiválasztására, és Naive Bayes, J48 és RepTree algoritmusokat alkalmaztak az osztályozáshoz. A GR és a Ranker által azonosított jellemző részhalmaz javította a javasolt Naive Bayes osztályozást [67].

Pattawaro és Polprasert az NSL-KDD adathalmazra egy attribútumarányon (AR) alapuló jellemzőkiválasztási módszert használtak k-Means klaszterezéssel és XGBoost osztályozással kombinálva. A javasolt modell 84,41%-os pontosságot ért el [68].

T. Ahmed és tsa. k-Nearest Neighbor (k-NN), SVM és Naive Bayes osztályozókkal dolgozott a KDD Cup99, Kyoto 2006 és UNSW-NB15 adathalmazokon, ahol a legjobb teljesítményt az SVM érte el 99,929%-os pontossággal és 0%-os hamis pozitív rátával [69].

M. Manonmani és S. Balakrishnan kutatásukban a sűrűségalapú jellemző kiválasztási (DFS) módszert használták szűrési megközelítésként az adathalmaz jellemzőinek rangsorolására. A DFS eredményeit ezután egy burkolásalapú optimalizálási technikának, az ITLBO (Improved Teacher Learner Based Optimization) algoritmusnak adták át, hogy megtalálják a legfontosabb jellemzőket tartalmazó optimális jellemzőkészletet a nagy pontosságú előrejelzéshez. Az EFS módszer eredményeit SVM, Gradient Boosting és CNN osztályozó algoritmusok segítségével értékelték. Az SVM 93%-os, a Gradient Boosting 97%-os, a CNN pedig 97,75%-os osztályozási pontosságot ért el a származtatott optimális jellemzőkészlet esetében. A javasolt munka az SVM és a CNN osztályozási algoritmusok segítségével kiválasztott 8 jellemző esetében 62,5%-os, a Gradient Boosting osztályozási algoritmus segítségével kiválasztott 9 jellemző esetében pedig 66,6%-os jellemzőcsökkentést ért el [70].

A. Hashemi, M. B. Dowlatshahi és H. Nezamabadi a közös jellemző választást többkritériumos döntéshozatali folyamatként (MCDM) modellezte 10 valós adatkészletre, változó számú jellemzővel. A VIKOR-módszert használták a jellemzők rangsorolásához több jellemző kiválasztási módszer értékelése alapján, mint különböző döntési kritériumok. A javasolt módszer először egy döntési mátrixot kap az egyes jellemzők különböző rangsorolási kritériumok szerinti rangsorolásával. Ezután minden egyes jellemzőhöz egy pontszámot rendelnek. Végül a kimenet a jellemzők rangsorvektora, amelyből a felhasználó kiválaszthatja a kívánt számú jellemzőt. A megközelítésük eredményei bizonyítják, hogy a pontosság, az F-pontszám és az algoritmus futási ideje tekintetében felülmúlja más hasonló módszereket. A megközelítésük gyorsan és hatékonyan teljesít [71].

N. Hoque és tsa. bemutat egy Ensemble Feature Selection using Mutual Information (EFS-MI) nevű módszert, amely a különböző feature selection módszerek - többek között az InfoGain, GainRatio, ReliefF, Chi-square és SymmetricUncertainty - eredményeinek részhalmazait kombinálja, hogy a jellemzők optimális részhalmazát kapja [72].

A.S. Sumant és D. Patil kutatásukban nagydimenziós adathalmazokat (HDD) dolgoztak fel többlépcsős módszerekkel, konkrétan a Chi-négyzet integrált RReliefF (ChS-R) és a szimmetrikus bizonytalanság integrált RReliefF segítségével. Az eredményeket ezután RF, KNN és SVM osztályozókkal validálták. A javasolt ChS-R rendszer 13,28%-os pontosság javulást ért el, míg az SU-R 9,47%-os pontosság javulást ért el [73].

Chih-Fong Tsaiand és Ya-Ting Sung kutatásukban több jellemző kiválasztási módszert írnak le, beleértve a PCA-t, a GA-t és a C4.5 döntési fát, kifejezetten a nagy dimenziós, alacsony mintaméretű (HDLSS) adatokra. Az eredményekhez kilenc párhuzamos és kilenc soros kombinatorikus megközelítést is megvizsgáltak, beleértve az uniót, a metszést. Tesztelési

eredményeik azt mutatják, hogy a soralapú együttes jellemző kiválasztási megközelítés különösen alkalmas a nagyon nagy dimenziójú adathalmazok feldolgozására [74].
Tanulmányukban J. Wang és társai az UCI gépi tanulási adathalmazt használták fel, hogy javaslatot tegyenek az SA-EFS-re, amely a rendezési aggregáción alapul. Az alkalmazott jellemző kiválasztási módszerek között szerepelt a Chi-négyzet, a maximális információs együttható és az XGBoost. A módszer teljesítményét KNN, Random Forest és XGBoost osztályozókkal értékelték, aminek eredménye a rendezési integráción alapuló funkcióválasztási módszer mindegyik (sonar, hcc-survival, musk) adatkészleten a legjobb eredményeket éri el 0,1-es küszöbértékkel, és az AUC 0,873, 0,840, illetve 0,859, ami magas szintű [75].

Számos tudományos publikáciban az IDS rendszerek vizsgálatához sokféle adathalmazzal dolgoznak, melyek feldolgozásához különféle jellemzőkiválasztási módszereket alkalmaznak. Az irodalomfeldolgozást követően azt a 6 jellemezőkiválasztási módszert választottam a kutatásomhoz, amelyeket együttes módszerrel nem vizsgáltak. A gépi tanulás alapú osztályozó algoritmusok összehasonlítási munkáira több kutatómunkában vizsgált osztályozók kombinációjából meghatározott 6 algoritmust használtam.

6. Jellemzők kiválasztása

A jellemzők kiválasztása a legrelevánsabb attribútumok megtalálására összpontosít, amelyek segítségével hatékony osztályozás vagy előrejelzés végezhető [76] [77] [78].

Hozzájárul a probléma dimenzionalitásának csökkentéséhez és így az erőforrásigény (tárolás, számítás) csökkenéséhez, valamint javíthatja a gépi tanuló algoritmusok teljesítményét [79], azaz gyorsabb képzés, csökkentett túlillesztés, és esetenként jobb előrejelző képesség érhető el. Bár úgy tűnhet, hogy ez a megközelítés információveszteséggel jár, ez nem így van, ha redundáns vagy irreleváns információ van jelen. A redundáns jellemzők az egy vagy több más jellemzőben található információk nagy részének vagy egészének másolatai, vagy más jellemzők kombinációjaként nyerhetők.

Az irreleváns attribútumok szinte semmilyen olyan információt nem tartalmaznak, amely az elvégzendő adatbányászati feladat szempontjából hasznos lenne. A redundáns és irreleváns jellemzők csökkenthetik az osztályozás pontosságát és a felfedezett klaszterek minőségét. Míg néhány irreleváns és redundáns attribútum a józan ész vagy a szakmai tudás alapján azonnal eltávolítható, az attribútumok legjobb részhalmazának kiválasztása gyakran szisztematikus megközelítést igényel.

A jellemzők kiválasztásának ideális megközelítése a jellemzők összes lehetséges részhalmazának kipróbálása a használt adatbányászati algoritmus bemeneteként, majd a legjobb eredményt adó részhalmaz kiválasztása. Ez a technika azonban rendkívül sok időt és számítási teljesítményt igényelne. Ezért számos más, elsősorban statisztikai feltételezéseken alapuló módszert fejlesztettek ki erre a célra. A jellemző kiválasztásának három alapvető megközelítése van.

- Wrapper-módszerek: a jellemző választó algoritmusok a tanulási módszert alprogramként használja, amelynek számítási terhe a tanulási algoritmus meghívása a jellemzők minden egyes részhalmazának kiértékelésére. Megkeresi a legjobb jellemzőkészletet egy adott típusú gépi tanulási algoritmushoz.
- Embedded methods: a gépi tanulási algoritmus dönti el, hogy milyen jellemzőket használjon, és melyeket hagyja figyelmen kívül.
- Szűrő módszerek: a jellemzők kiválasztása az adatbányászati algoritmus futtatása előtt történik, az adatbányászati feladattól független módszerrel.

6.1. Többtényezős kiválasztás

A többtényezős kiválasztás (EFS - Ensemble Feature Selection) olyan technika, amely több jellemző választó algoritmus erősségeit használja ki, hogy javítsa a jelentős jellemzők azonosítását egy adathalmazban. Az együttes jellemző választás előnyei közé tartozik a fokozott osztályozási pontosság, a csökkent túlillesztés és a kiválasztott jellemzők nagyobb stabilitása. Ez a megközelítés különösen előnyös lehet a gépi tanulás által vezérelt alkalmazásokban, például a behatolás érzékelő rendszerekben, ahol a jellemzők sokfélesége hatással lehet a modell pontosságára és tanítási időtartamára.
A különböző jellemző választó algoritmusok előnyeinek egyesítésével az együttes jellemzőválasztás megkönnyítheti az adott feladat szempontjából legfontosabb jellemzők azonosítását, ami hatékonyabb és eredményesebb adatelemzést eredményez. Az EFS használatának azonban vannak hátrányai is. Az összes modell futtatása jelentős számítási erőforrásokat igényel, és a modell pontossága és a számítási idő közötti megfelelő egyensúly megtalálása kihívást jelenthet. Összességében az EFS hatékony és népszerű technika az adatok kiválasztására, amely javíthatja a modell pontosságát és csökkentheti a redundanciát [80].
A következő 6 alfejezetben a kutatási munka során használt jellemzőkiválasztási módszerek ismertetése található.

6.2. Információnyereség

Az Információnyereség (Information Gain - IG) két változó közötti kölcsönös függőség mérőszáma. Annak mérésére szolgál, hogy egy véletlen változóból mennyi hasznos információ nyerhető ki egy másik változó által. Más szóval az információnyereség a függőség szimmetrikus mérőszáma. Megmutatja egy jellemzővektor egy adott attribútumának fontosságát. Az attribútum által az osztályozási feladathoz nyújtott információ mennyiségét számszerűsíti [81]. Értéke az entrópián alapul. Egy „Y” osztályjellemző entrópiája [82]:

$$H(Y) = -\sum p(y)\, log_2(\, p(y)), \tag{3}$$

ahol *p(y)* az *Y* véletlen változó marginális valószínűségi sűrűségfüggvénye. Tegyük fel, hogy a tanító adathalmaz *Y* megfigyelt értékeit tartalmazza, amelyek egy másik *X* jellemző értékei alapján oszthatók fel. Ha az *Y* entrópiája csökken az *X* alapján történő felosztással, az arra utal,

hogy *X* és *Y* között korreláció van. Más szóval, az *Y* entrópiája *X* figyelembevétele után a következőképpen fejezhető ki:

$$H(Y \mid X) = \sum p(x) \sum p(y \mid x) \log_2 (p(y \mid x)), \tag{4}$$

ahol *p(y|x)* az *y* feltételes valószínűsége *x* esetén. Az entrópia, mint az adathalmaz tisztátalanságának mérőszáma, segítségével az *X* jellemző által az *Y* célváltozóról nyújtott további információt az *Y* entrópiájának csökkenésével tudjuk számszerűsíteni. Ezt a mértéket információnyereségnek nevezzük, és az *X* és *Y* közötti függőség mértékét jelzi:

$$IG(X,Y) = H(Y) - H(Y \mid X) = H(X) - H(X \mid Y). \tag{5}$$

Az információnyereség (IG) mérőszám szimmetrikus, és egy adathalmaz összes jellemzőjének rangsorolására használható. A jellemzők egy részhalmazának rangsorolás alapján történő kiválasztásához azonban küszöbértéket kell beállítani. Az IG mérték egyik korlátja, hogy hajlamos a nagyobb értékszámú jellemzőket előnyben részesíteni, még akkor is, ha azok nem nyújtanak több hasznos információt [83].

6.3. Nyereségarány

Az információnyereség módszer előnyben részesíti a nagyszámú értékkel rendelkező attribútumok kiválasztását, ami a jellemzőválasztási módszer, a Gain Ratio (GR) kifejlesztéséhez vezetett, amely az információnyereség módosítását jelenti, és célja a torzítás csökkentése. Az eredetileg döntési fákhoz kifejlesztett GR az attribútum kiválasztásakor figyelembe veszi az ágak számát és méretét. Az információnyereség által adott értékelést javítja azáltal, hogy figyelembe veszi a jellemzőben lévő elágazások számát, azaz azt, hogy azok mennyire egyenletesen oszlanak el [84]. A GR az egyes jellemzők relevanciáját tükrözi, minél nagyobb az értéke, annál nagyobb a jellemző befolyása. A nyereségarányt a következő képlettel számoljuk ki:

$$GR(X,Y) = \frac{IG(X,Y)}{SplitInfo}, \tag{6}$$

$$SplitInfo = -\sum p(i) \cdot \log_2 (p(i)), \tag{7}$$

ahol a *p(i)* azon példányok arányát jelöli, amelyek a jellemző egy adott felosztásába esnek [85].

6.4. Relief

A Relief-módszer minden egyes jellemzőhöz kiszámít egy súlyértéket (W_j a j jellemző súlya), amely a jellemző minőségének vagy relevanciájának becslésére használható [86]. A súlyvektort nulla értékkel inicializáljuk, és iteratív megközelítéssel frissítjük. Az algoritmus véletlenszerű m elemű mintát vesz az adathalmazból. A minta minden egyes példánya (R_i) esetében megkeresi a legközelebbi, azonos osztályba tartozó példányt (H_i) és a legközelebbi, másik osztályba tartozó példányt (M_i). Ezután az egyes jellemzősúlyok értékét a képlet segítségével frissíti:

$$W_j = W_j - \frac{D(R_{ij}, H_{ij})}{m} + \frac{D(R_{ij}, M_{ij})}{m}, \qquad (8)$$

ahol R_{ij}, H_{ij}, M_{ij} az i-edik példány j-edik jellemzője. A D függvényt a következőképpen definiáljuk:

$$D(x, y) = \begin{cases} 0, & ha\ x = y, \\ 1, & egyébként. \end{cases} \qquad (9)$$

A Relief hiányossága, hogy nem azonosítja a redundáns jellemzőket, és csak bináris osztályozási problémák esetén használható.

6.5. Szimmetrikus bizonytalanság

A jellemzők rangsorát a jellemzők kiválasztása tekintetében a szimmetrikus bizonytalanság (Symmetric Uncertainty - SU) is meghatározhatja. Ezt úgy számítjuk ki, hogy az információnyereség értékének kétszeresét normalizáljuk a két változó entrópiáinak összegére [87]. A magas SU-érték egy attribútum nagy fontosságát jelzi.

$$SU(X, Y) = \frac{2 \cdot IG(X, Y)}{H(X) + H(Y)}, \qquad (10)$$

ahol $H(X)$ és $H(Y)$ az X és Y változók entrópiaértékei, míg $IG(X,Y)$ az X és Y változókhoz kapcsolódó információnyereség [88].

6.6. Khí-négyzet próba

A Khi-négyzet próba a jellemzők kiválasztására egy olyan statisztikai technika, amelyet egy adott adathalmazban a célváltozó szempontjából legrelevánsabb jellemzők azonosítására használnak. Úgy működik, hogy összehasonlítja egy jellemző értékeinek megfigyelt eloszlását

a jellemző és a célváltozó közötti függetlenség feltételezése szerinti várható eloszlással, és kiválasztja azokat a jellemzőket, amelyeknél a legnagyobb a különbség a megfigyelt és a várható eloszlás között. Számítása az alábbi képleten alapul [89]:

$$\chi^2 = \sum_{i=1}^{n_I} \sum_{j=1}^{n_c} \frac{\left[A_{ij} - \frac{R_i B_j}{N} \right]^2}{\frac{R_i B_j}{N}}, \tag{11}$$

ahol n_I az intervallumok száma, n_c az osztályok száma, N az összes példányszám, A_{ij} az i intervallumban és a j osztályban lévő példányok száma, R_i az i intervallumban lévő példányok számát, B_j pedig a j osztályban lévő példányok számát jelöli. A Khi-négyzet-teszt alapú kiértékelés diszkrét változókra lett kidolgozva. Ezért folytonos jellemzők esetén az alkalmazása előtt diszkretizálást kell végezni.

6.7. Varianciaanalízis

A varianciaanalízis (Analysis of Variance - ANOVA) egy olyan statisztikai elemzési technika, amelyet több csoport átlagának összehasonlítására használnak annak megállapítására, hogy van-e szignifikáns különbség közöttük. A Khi-négyzet megközelítéshez hasonlóan a használata előtt diszkretizálásra van szükség [90]. Az ANOVA kulcsgondolata az adatok teljes varianciájának összehasonlítása a csoportokon belüli és a csoportok közötti eltérésekkel.

A csoporton belüli négyzetek összege (SSW) a csoportokon belüli variációt méri. Ez a következőképpen van meghatározva:

$$SSW = \sum_{i=1}^{k} [(n_i - 1) \cdot SS(i)], \tag{12}$$

ahol n_i az i csoportba tartozó példányok száma, és $SS(i)$ az i csoport varianciája. A csoportok közötti négyzetek összege (SSB) a csoportok átlagai közötti eltérést méri. Ez a következőképpen van meghatározva:

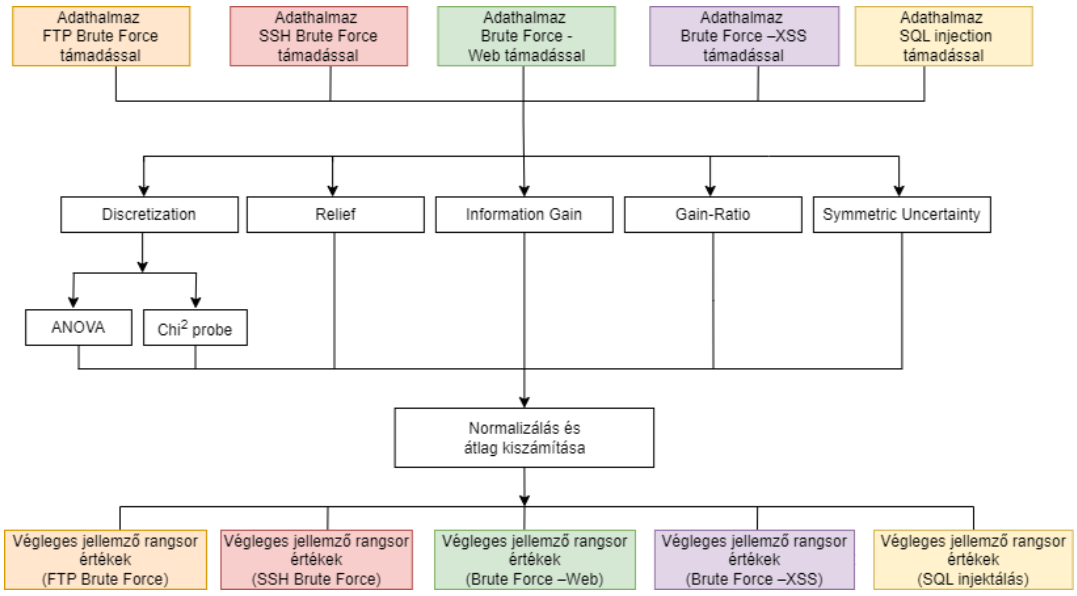
$$SSB = K \cdot \sum (x - \bar{\bar{x}})^2, \tag{13}$$

ahol K a csoportok száma, x_i az i csoport átlaga, és x_i az összes példány átlaga. A teljes négyzetek összege (SST) a következő:

$$SST = SSW + SSB. \tag{14}$$

6.8. Többtényezős kiválasztás számtani középpel

Az előző 6 alfejezetben bemutatott és a vizsgálat során használt összes jellemzőkiválasztási módszer a szűrő módszerek csoportjába tartozik. Előnyük, hogy a három csoport közül (wrapper, embeded, szűrő) az időbonyolultságuk a legalacsonyabb, és általában alkalmazásuk után a gépi tanuló algoritmus kevésbé hajlamos a túlillesztésre.



13. ábra Jellemzőkiválasztási módszerek

A hat jellemzőválasztási módszert mind az öt adatkészletre alkalmaztam 30 egyetemi laboratóriumi számítógépen, valamint az ELKH felhőszolgáltatások [91] segítségével. A jellemzőkiválasztási munkafolyamatot a 13. ábra mutatja be.

Bár több feladatot párhuzamosan végeztem, a teljes folyamat több mint két hónapot vett igénybe.

Minden egyes adatkészlet és minden egyes módszer esetében normalizáltam a jellemzőkiválasztási folyamat végén kapott jellemzőpontszám-értékeket. Ezután a végső jellemzőpontszámot minden egyes adatkészlet esetében külön-külön a normalizált pontszámok átlagaként számoltam ki. A részletes eredmények a Melléklet A.1. - A.5. táblázatában találhatók.

6.9. Küszöbértékek meghatározása

Az előző fejezetben leírt jellemzőkiválasztási módszerek segítségével az adathalmazokban szereplő minden egyes jellemzőhöz egy jellemzőpontszám-értéket számoltam az előzőekben ismertetett átlagolás segítségével. Ezt követően az értékekhez rangsorolási küszöbértéket határoztam meg 0,05-től kezdve, növelve 0,05 lépéssel 0,55-ig. Minden egyes küszöbértékhez azokat a jellemzőket választottam ki, amelyek pontszáma magasabb az adott küszöbértéknél, így csökkentett számúkülönböző jellemzőcsoportokat határoztam meg. A küszöbértékekhez társult jellemző darabszámok adathalmazonként a 10. táblázatban láthatók.

10. táblázat Jellemzőszámok csökkentésének eredményei a rangsorolási küszöbértékekkel

Küszöbérték	FTP	SSH	WEB	XSS	SQL
0,05	56	59	65	65	66
0,10	43	53	60	57	64
0,15	32	48	60	57	60
0,20	23	29	58	51	57
0,25	21	22	56	46	48
0,30	13	17	50	36	37
0,35	8	7	44	31	31
0,40	3	2	34	27	26
0,45	2	2	23	10	12
0,50	2	1	9	6	4
0,55	2	1	1	1	2

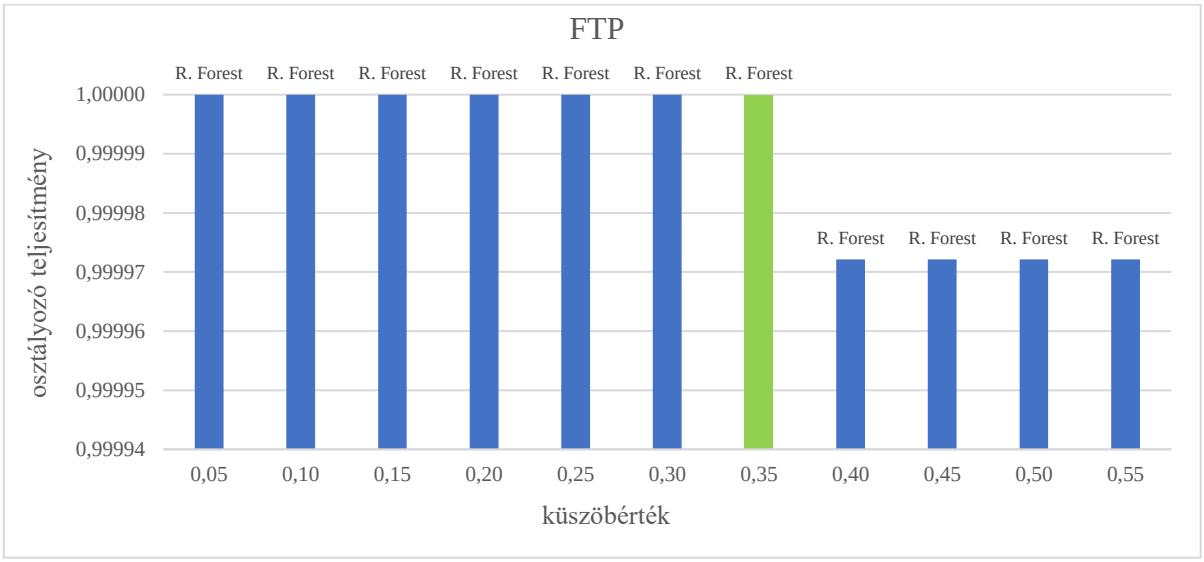
A küszöbértékekhez rendelt jellemzők sorszámainak teljes listája adathalmazonként a Melléklet A.6. táblázatában látható.

6.10. Eredmények értékelése

A küszöbértékekhez meghatározott jellemzők segítségével gépi tanulás alapú osztályozó algoritmusok vizsgálatát végeztem el annak érdekében, hogy alacsony jellemezőszám mellett elfogadható vagy jó osztályozási eredményt érjek el. Minden adathalmaz esetében 5 osztályozó algoritmust vizsgáltam különböző osztályozási teljesítménymérőkkel az adott küszöbértékek alkalmazása esetén kiválaszott jellemzőkkel. Minden osztályozónál a tanító és teszthalmazok vizsgálatával létrejött Accuracy, Precision, Recall, F1 teljesítményértékek (0-1 közé eső szám, ahol az 1 a legjobb teljesítményt mutatja) számtani átlagát figyelembe véve kiválasztottam a legnagyobb értéket. Így meghatároztam, minden adathalmaz esetén azt a küszöbértéket, ahol a legkisebb jellemzőszámmal jó osztályozási eredményt érek el.

A 14.-18. ábrán bemutatott oszlopdiagramokban minden támadás típus esetén külön megmutattam az egyes küszöbértékek mellett elért legjobb átlagos osztályozási teljesítményt. Minden oszlop felett megjelenik azon osztályozó neve, amivel az adott küszöbértéknél a legjobb eredményt sikerült elérni. Az oszlopdiagramokon a függőleges tengely az átlagos osztályozási teljesítmény értéket, a vízszintes tengely a küszöbértékeket ábrázolja. A zölddel jelölt oszlop az az érték, ahol a legkisebb jellemzőszámmal a legjobb osztályozási eredményt értem el, így ott lett a kiválasztott küszöbérték.

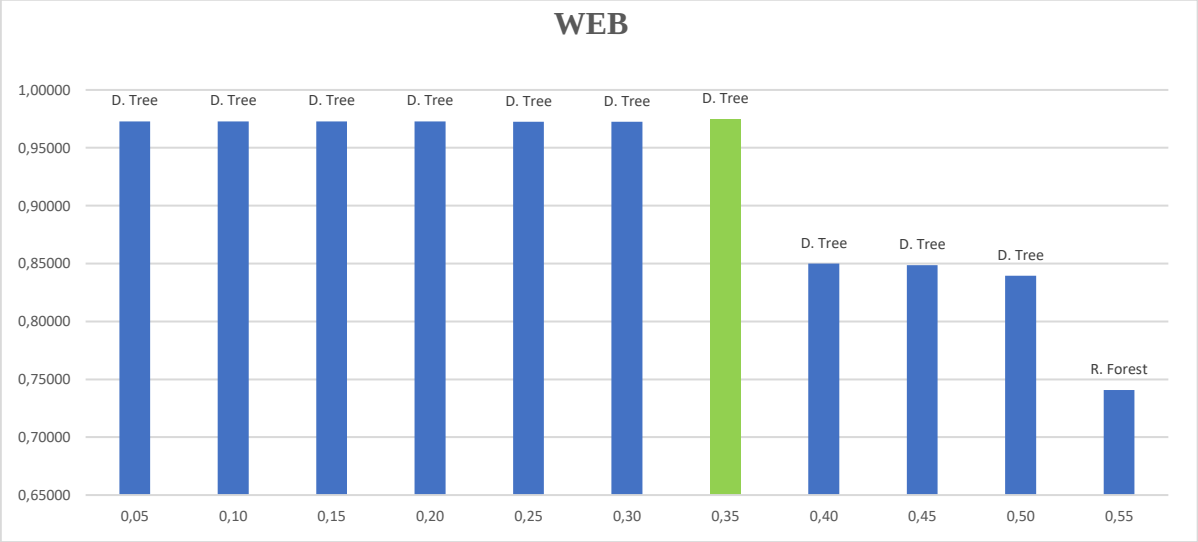
Az FTP és az SSH halmaz esetében nagyon minimális teljesítményeltérés látható, de még ebben az esetben is jól elkülöníthető a meghatározott 0,35 küszöbérték.



14. ábra Küszöbérték meghatározása az FTP adathalmaznál

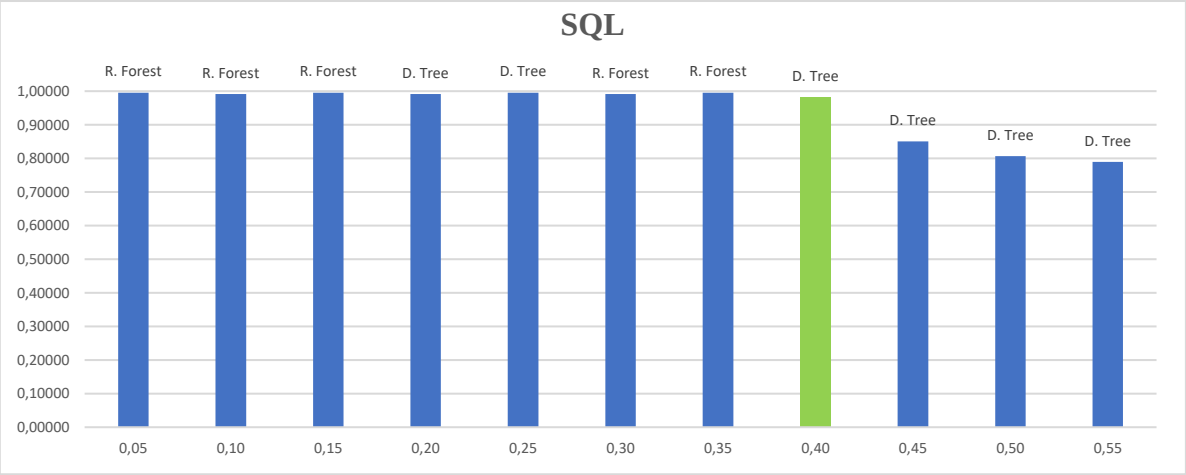
15. ábra Küszöbérték meghatározása az SSH adathalmaznál

A WEB adathalmaz esetében az FTP és SSH adathalmazokhoz hasonlóan 0,35 lett az a küszöbérték szám, ahol a legkisebb jellemzőszámmal jó eredmény érhető el és jelentősen eltér a rosszabb eredménytől.

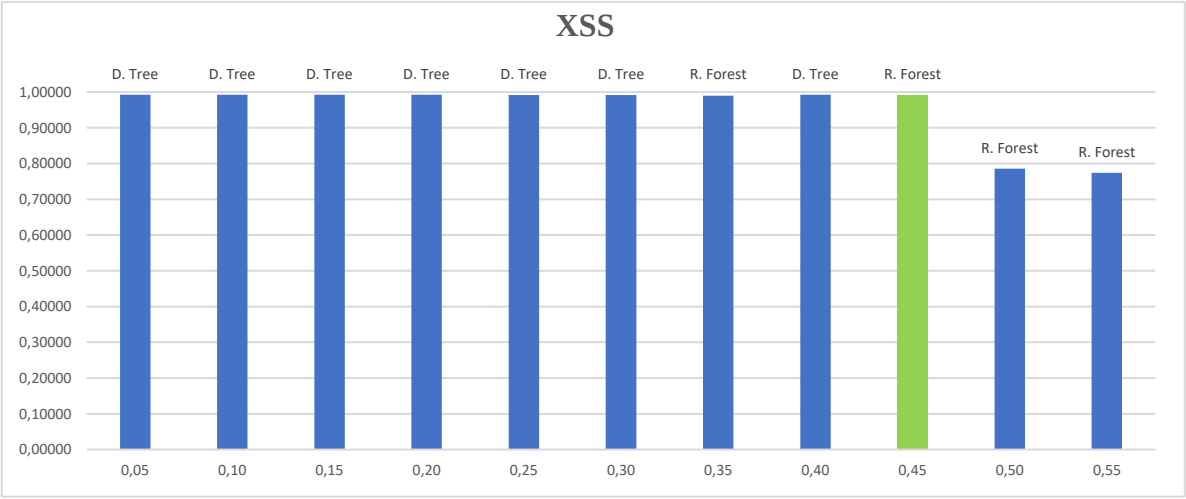


16. ábra Küszöbérték meghatározása a WEB adathalmaznál

Az SQL és XSS adathalmazhoz meghatározott küszöbérték 0,40 és 0,45. Itt a legkisebb jellemzővel a még legjobb osztályozási eredmény is jelentősebb különbséget mutat a többi halmazhoz képest az osztályozási teljesítmény érték.



17. ábra Küszöbérték meghatározása az SQL adathalmaznál



18. ábra Küszöbérték meghatározása az XSS adathalmaznál

Az eredeti adatokban található 69 jellemzőből a meghatározott küszöbértékeknek megfelelően mindegyik támadástípusra meghatározásra került a csökkentett jellemzők darabszáma, melynek eredménye a 11. táblázatban látható.

11. Táblázat Jellemzőszámok csökkentésének eredményei a rangsorolési küszöbértékekkel

Adathalmaz	Küszöbérték	Jellemzők darabszáma
FTP	0,35	8
SSH	0,35	7
WEB	0,35	44
SQL	0,40	26
XSS	0,45	10

1. TÉZIS

Egy IDS rendszer tanítására használt adatbázis segítségével olyan módszert dolgoztam ki, ami alkalmas a beazonosításhoz szükséges jellemzők fontossági sorrendjének meghatározására az Információnyereség, Nyereségarány, Szimmetrikus bizonytalanság, Relief, Khi-négyzet próba és a Varianciaanalízis módszerek normalizált értékszámainak átlagai alapján végzett rangsorolás segítségével.

A tézisemhez tartozó publikációm a következő: [S3]

7. Gépi tanulás alapú osztályozási algoritmusok

Az osztályozási módszereket arra használják, hogy megjósolják egy objektumpéldány osztályát egy jellemzővektor alapján. A gépi tanuláson alapuló osztályozási algoritmusok olyan modelleket építenek fel, amelyek képesek címkézett adathalmazokból tanulni, és ezeket felhasználni az új, nem látott adatpontok osztályának előrejelzésére. Ebben a vizsgálatban öt különböző osztályozó algoritmust használtunk, amelyek négy fő osztályozási csoportot képviselnek.

Ezek a csoportok a lineáris modellek, a valószínűségi modellek, a fa alapú modellek és a kernel alapú modellek. A lineáris modelleket a Logisztikus regressziós módszer képviseli, amely a bináris kimenetel valószínűségét egy szigmoid függvény segítségével modellezi. A valószínűségi modelleket a Naive Bayes-modell képviseli, amely feltételezi, hogy a jellemzők függetlenek az osztályt tekintve, és Bayes tételét használja az egyes osztályok utólagos valószínűségeinek kiszámításához.

A fa alapú modelleket két módszer képviseli: a döntési fa módszer, egy nem parametrikus modell, amely rekurzívan felosztja a jellemzőteret egy fa struktúrára, és a Random Forest módszer, egy együttes modell, amely több döntési fát használ, és a teljesítmény javítása érdekében aggregálja a jóslatokat. A mag alapú modelleket a Support Vector Machine (SVM) módszer képviseli, amely a bemeneti adatokat egy nagydimenziós jellemzőtérbe képezi le, és olyan hipersíkot talál, amely maximálisan elválasztja az osztályokat.

Gyakran használt értékelési metrikák az osztályozó algoritmusok teljesítményének mérésére:

- Pontosság (Accuracy): Az Accuracy a helyesen osztályozott minták aránya az összes mintához képest. Matematikailag az Accuracy a helyesen osztályozott minták számát osztja az összes minta számával.
- Pontossági arány (Precision): A Precision azt méri, hogy az osztályozó mennyire pontosan találta meg a pozitív eredményeket a helyesen osztályozott pozitív és hamis pozitív eredmények arányával. Tehát a Precision a helyesen pozitívnak osztályozott minták aránya az összes pozitívnak osztályozott mintához képest.
- Feldfedés aránya (Recall vagy Sensitivity): azt méri, hogy az osztályozó mennyire hatékonyan azonosította a pozitív eredményeket a helyesen osztályozott pozitív és hamis negatív eredmények arányával. Tehát a Recall azon minták aránya, amelyeket helyesen pozitívnak osztályozott az összes valós pozitív mintához képest.

- F1 Score: Az F1 Score egy olyan összetett metrika, amely figyelembe veszi a Precision (Pontosság) és Recall (Azonosítás) értékeket. Az F1 Score a Precision és Recall harmonikus közepét jelenti, és segít egyensúlyt teremteni a pontosság és azonosítás között. Ez különösen hasznos, ha az osztályok eltérő méretűek vagy a hamis pozitív és hamis negatív eredmények nagy hatással vannak a problémára.

Fontos megjegyezni, hogy ezek a metrikák attól függően, hogy milyen problémával és osztályozási feladattal állunk szemben, változhat a fontosságuk és alkalmazásuk. Például egy olyan feladatban, ahol az egyik osztály ritka vagy a hamis pozitív eredmények különösen károsak lehetnek, a Recall (Azonosítás) lehet a legfontosabb metrika, míg más esetekben az Accuracy (Pontosság) is releváns lehet.

A következő alfejezetek a fent említett osztályozási algoritmusok leírását tartalmazzák.

7.1. Logisztikus regresszió

A Logisztikus regresszió (Logistic Regression - LR) egy lineáris osztályozási technika, amelyet annak meghatározására használnak, hogy egy példány milyen valószínűséggel tartozik egy adott osztályba, például egy támadáshoz. Az LR a lineáris módszerek családjába tartozik, és a diszkriminancia-elemzés alternatívája. Alkalmazásának előfeltételei kevésbé szigorúak, mint a diszkriminancia-analízisé [92]. A Logisztikus regresszió lényege, hogy minden egyes megfigyelési példányra kiszámítjuk a $X=[x_1, x_2...,x_n]$ jellemzőértékek lineáris kombinációját (lásd 15. egyenlet) az osztályozó tréningje során meghatározott $A=[a_0,a_1,...,a_n]$ együtthatóvektor segítségével:

$$Z(x) = a_0 + a_1 \cdot x_1 + ... + a_n \cdot x_n . \tag{15}$$

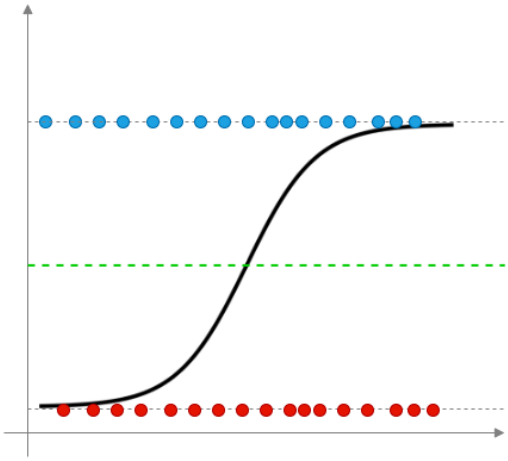
A támadási osztályba (1. osztály) való tartozás valószínűségének meghatározásához a Logisztikus regresszió egy szigmoid függvényt alkalmaz (lásd 19. ábra) *Z*-re, amely *Z*-t a [0, 1] intervallumra képezi le

$$h(z) = \frac{1}{1+e^{-z}} . \tag{16}$$

Végül az osztályozó úgy dönt a megfigyelés végső osztályáról, hogy a kapott *h* valószínűségi értéket összehasonlítja egy h_{tr} küszöbértékkel, ahogyan az az egyenletben látható.

$$C(h) = \begin{cases} 1 & ha \quad h > h_{tr} \\ 0 & egyébként \end{cases}. \tag{17}$$

A küszöbértéket (h_{tr}) az osztályozó tanítási szakaszában határozzuk meg.



19. ábra Logisztikus regresszió

7.2. Naive Bayes

A Naive Bayes osztályozási algoritmus egy valószínűségi módszer, amely feltételezi a jellemzők közötti függetlenséget. Kiszámítja az egyes osztályok valószínűségét a jellemzők halmaza alapján (lásd 20. ábra), majd a legnagyobb valószínűségű osztályt választja ki az új példány megjósolt osztályaként. Egy $X=[x_1, x_2,\ldots, x_n]$ megfigyelés osztályát a következő képlet segítségével jósolja meg

$$C(x) = \arg\max_{c_j \in C} P(c_j) \prod_{i=1}^{n} P(x_i \mid c_j), \tag{18}$$

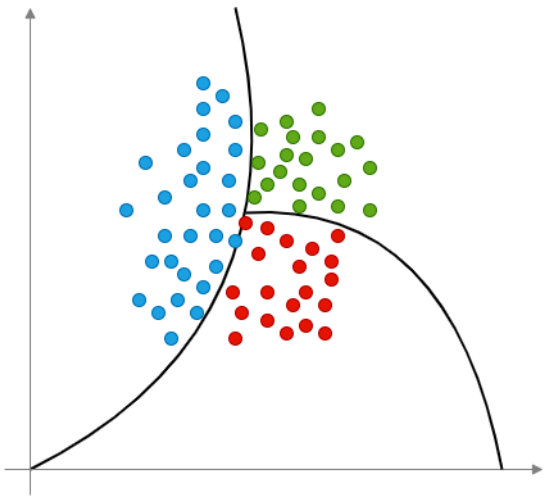
ahol c_j a j-edik osztály, x_i az x_i-edik jellemző értéke, *n* a jellemzők száma, $P(c_j)$ a *j* osztály előzetes valószínűsége, és $P(x_i \mid c_j)$ az x_i értékének feltételes valószínűsége a c_j osztály esetén. A $P(c_j)$ előzetes valószínűséget a *j* osztály relatív gyakoriságával becsüljük a tanító mintában.

Kategorikus jellemzők esetén a $P(x_i \mid c_j)$ becslését az x_i érték relatív gyakoriságával végezzük a c_j osztályba tartozó tanítási mintaelemek között. Folyamatos jellemzők esetén a $P(x_i \mid c_j)$

becslését az x_i -re számított valószínűségi sűrűségfüggvény értéke adja, figyelembe véve a c_j osztályba tartozó tanítási mintaelemeket

$$P(x_i \mid c_j) = \frac{1}{\sigma_i \sqrt{2\pi}} e^{-\frac{(x_i - \mu_i)^2}{2\sigma^2}}, \tag{19}$$

ahol μ_i az átlagérték és σ_i az i-edik jellemző szórása a figyelembe vett tanító mintaelemek között.

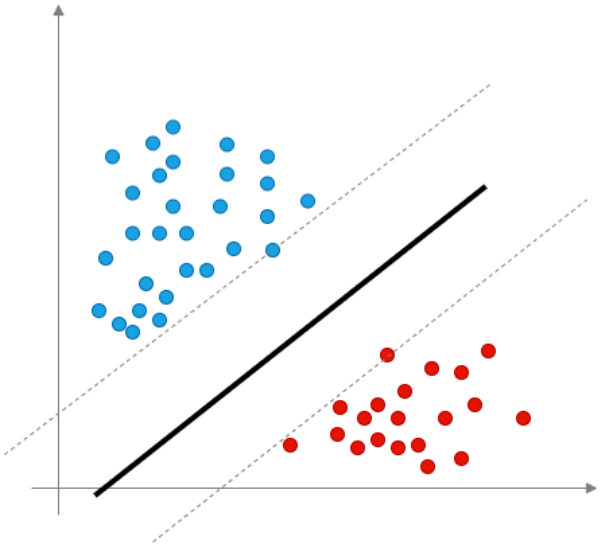


20. ábra Naive Bayes

7.3. Tartóvektor-gép

A Tartóvektor-gépek (Support Vector Machine – SVM) [93] bináris osztályozási problémák esetén egy többdimenziós hipersíkot hoz létre (lásd 21. ábra), amely elválasztja a két osztályt. A többosztályos problémák több bináris osztályozási problémára redukálódnak. Ha nem lehet egyszerű lineáris szétválasztást végezni, akkor az adatokat olyan ún. kernel függvények segítségével transzformálja, amelyek magasabb dimenzióban számítják ki a hipersíkot. A hipersík nemlinearitása a regularizáció és a gamma paraméterek segítségével is hangolható.

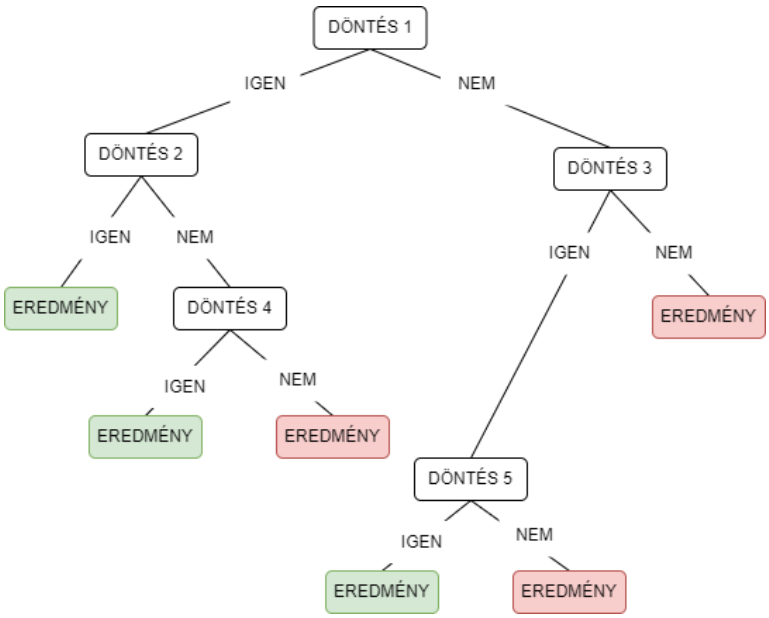
A regularizációs paraméter értéke azt írja le, hogy mennyire akarjuk elkerülni a téves besorolást a tanító példányok esetében. Egy magas érték egy összetettebb hipersíkot eredményezhet, amelyhez kevés tévesen osztályozott adatpont tartozik, ha egyáltalán van ilyen. Magas gammaértékek esetén csak a hipersíkhoz közeli tanító példányokat vesszük figyelembe a hipersík meghatározása során.



21. ábra Tartóvektor-gép

7.4. Döntési Fa

A döntési fák (Decision Tree) könnyen értelmezhető és vizualizálható eszközt kínálnak az osztályozáshoz. A döntést a tanító minta jellemzőértékeiből levezetett szabályok alapján hozzák meg. A fa minden egyes levele egy osztálycímkét jelöl. Minden egyes csomópontban csak egy jellemzőt vesznek figyelembe, és nincs olyan gyökér-levél útvonal, amely kétszer ugyanazt a jellemzőt tartalmazza. Az osztályozási fa (lásd 22. ábra) az osztályozás minőségére vonatkozó megbízhatósági mérőszámot is megadhat. A fa a gyakorló mintából rekurzív módon épül fel [94]. Ez egy iteratív folyamat, amelynek során az adatokat partíciókra osztjuk, majd minden egyes ágon tovább partícionáljuk.



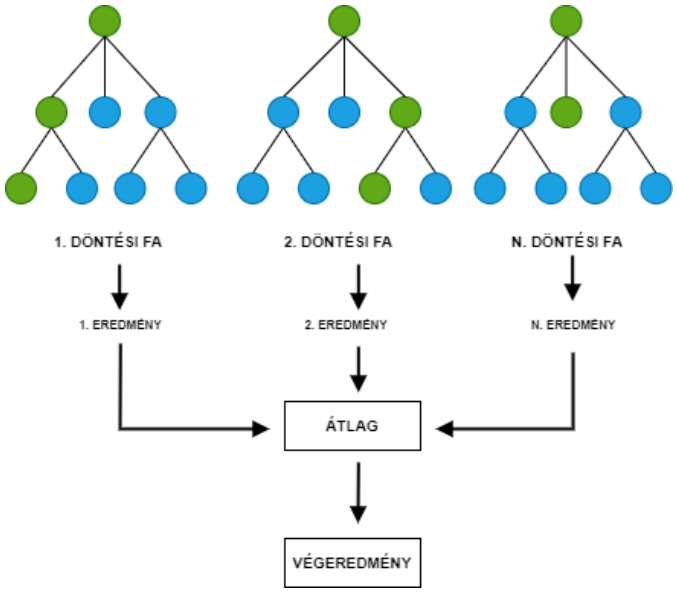
22. ábra Döntési fa

A különböző csomópontokban használt jellemzők kiválasztása olyan statisztikai módszerek segítségével történik, mint az információnyereség vagy a Gini-index [95]. Ha már minden jellemzőt felhasználtak, és a fennmaradó minta egynél több osztályba tartozó példányokat tartalmaz, akkor egy levél jön létre, és annak osztályáról többségi szavazással döntenek.

7.5. Véletlen erdő

A Véletlen erdő (Random Forest - RF) módszert [96] a döntési fák egyik hiányosságának kiküszöbölésére fejlesztették ki (lásd 23. ábra), azaz a mintaadatok túlillesztésére való hajlamot. Az RF ezt a problémát a bootstrapping nevű statisztikai technika alkalmazásával enyhíti, amely több modellt hoz létre, és ezek eredményeit kombinálja a végső döntés meghozatalához.

Az RF fő gondolata az, hogy több osztályozó előrejelzéseinek összesítésével minimalizálható az egyedi hibák hatása. A Bootstrapping során a tanító adathalmazból több kisebb mintát véletlenszerűen, helyettesítéssel húznak ki. Minden egyes mintát egy külön osztályozó képzésére használunk. Így egy új megfigyelés osztályozásakor a végső osztályjóslás az egyes modellek által adott eredmények összesítésével történik. Ez általában többségi szavazással történik.

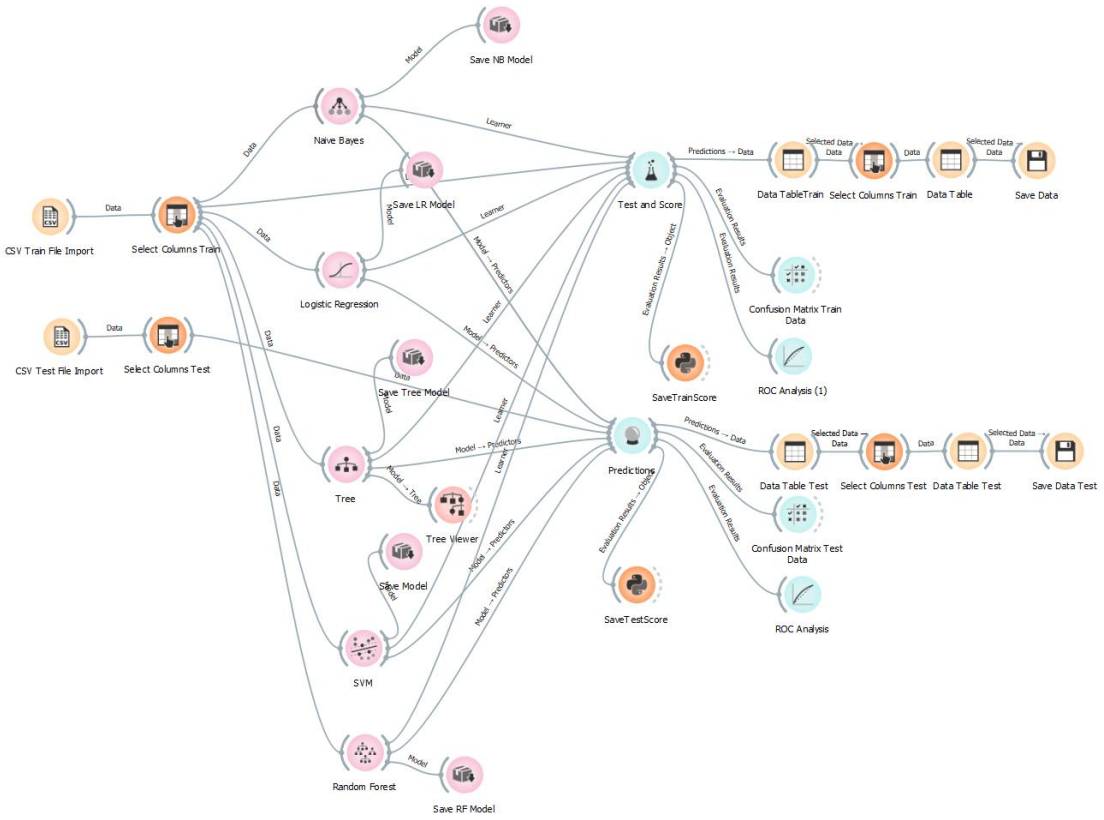


23. ábra Véletlen erdő

7.6. Gyakorlati megvalósítás

Az öt osztályozó képzése és tesztelése az Orange 3.34 program segítségével történt, amely egy nyílt forráskódú adatvizualizációs, gépi tanulási és adatbányászati eszközkészlet. Vizuális programozási front-endet kínál interaktív adatvizualizációhoz és feltáró, gyors kvalitatív adatelemzéshez. Komponensei az úgynevezett widgetek, és az egyszerű adatvizualizációtól, a részhalmazok kiválasztásától és az előfeldolgozástól a tanulási algoritmusok empirikus értékeléséig és a prediktív modellezésig terjednek.

A vizuális programozás egy olyan felületen keresztül valósul meg, amelyen a munkafolyamatok előre definiált vagy a felhasználó által tervezett widgetek összekapcsolásával jönnek létre, míg a haladó felhasználók az Orange-ot Python-könyvtárként használhatják az adatmanipulációhoz és a widgetek módosításához. Az Orange a tudományos számításokhoz használt Python nyílt forráskódú könyvtárakat használja, mint például a numpy, scipy és scikit-learn, míg grafikus felhasználói felülete a platformokon átívelő Qt keretrendszerben működik. A vizsgálat során használt osztályozó tanító és tesztelési munkafolyamatot a 24. ábra szemlélteti. Minden egyes támadástípusra és minden egyes releváns jellemzőgyűjteményre külön-külön végeztem el a folyamatot. Például az FTP Brute Force támadás és a 0,40-es rangsorolási küszöbérték esetében három jellemzőnek (44, 56 és 59) kellett jelentős szerepet játszania. Így összesen 21 munkafolyamat végrehajtására volt szükség, és 105 osztályozó lett képezve.



24. ábra Osztályozó algoritmusok működése az Orange programban

A bináris osztályozás célja, hogy a tanító adathalmaz alapján tanuljon egy osztályozó algoritmust, amely képes új, ismeretlen példákra is osztályozni. A tanító adathalmazban minden példa rendelkezik egy címkével (osztályozási címkével), ami meghatározza a helyes osztályt. Az osztályozó algoritmusok teljesítményének értékelésére a 12. táblázatban szereplő előrejelzési szempontok láthatóak.

12. táblázat osztályozók értékelési szempontjai

Címke tulajdonság értéke	Besorolási érték	Előrejelzés
0	0	TN
0	1	FP
1	0	FN
1	1	TP

A lehetséges előrejelzések:

1. Valódi pozitív (True Positive - **TP**): Ez akkor fordul elő, ha olyan adatpéldányt (támadást) jelöl, amelyre az osztályozó pozitív eredményt adott vissza, és az előrejelzés helyes (valódi támadás).
2. Hamis pozitív (False Positive - **FP**): Ezt akkor tapasztaljuk, amikor az osztályozó helytelenül azonosítja vagy a tesztelő algoritmus helytelenül diagnosztizálja a negatív osztályba tartozó példákat. Tehát az eredmény pozitív, de a valódi osztály negatív (támadásként lett azonosítva, pedig nem volt az).
3. Igaz negatív (True Negative - **TN**): Ez akkor következik be, amikor az osztályozó helyesen azonosítja vagy a tesztelő algoritmus helyesen diagnosztizálja a negatív osztályba tartozó példákat. Tehát az eredmény negatív, és a valódi osztály is negatív (nem volt támadás, és az osztályozó sem tekintette annak).
4. Hamis negatív (False Negative - **FN**): Ez akkor fordul elő, amikor az osztályozó helytelenül azonosítja vagy a tesztelő algoritmus helytelenül diagnosztizálja a pozitív osztályba tartozó példákat. Tehát az eredmény negatív, de a valódi osztály pozitív (támadás történt, de az osztályozó nem tekintette annak).

Az összes osztályozót a gyakorló és tesztminták alapján értékeltem négy, azaz az osztályozási pontosság (Accuracy), a pontosság (Precision), a fedés (Recall) és az F-mérték (F1) mérőszámmal, melyek az alábbi képlettekkel számíthatóak ki:

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP}, \tag{20}$$

$$\mathrm{Pr}ecision = \frac{TP}{TP + FP}, \tag{21}$$

$$\mathrm{Re}call = \frac{TP}{TP + FN}, \tag{22}$$

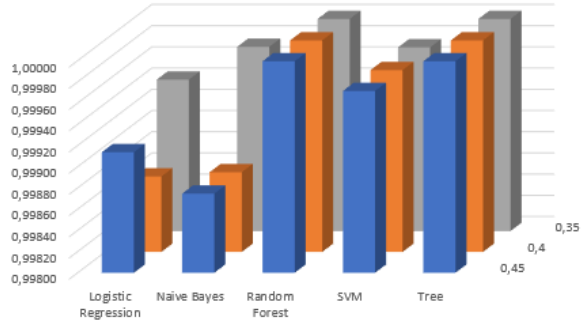
$$F1 = \frac{2(\mathrm{Pr}ecision \cdot \mathrm{Re}call)}{\mathrm{Pr}ecision + \mathrm{Re}call}. \tag{23}$$

A 6.10-es fejezetben leírt módszerrel, mely alapján az osztályozók teljesítményei alapján mindegyik adathalmazra meghatároztam egy olyan küszöbértéket, ahol a legkisebb jellemzőszámokkal a legjobb osztályozási teljesítményt értem el. Mindegyik adathalmazt figyelembe véve a legkisebb küszöbérték a 0,35. Ez alapján az öt adathalmazra megállapított jellemzőcsoportokkal vizsgáltam a 0,35-0,55 közötti küszöbértékeknél az osztályozók teljesítményét. A számítási eredmények megtalálhatóak a Melléklet A.7. – A.11. táblázataiban.

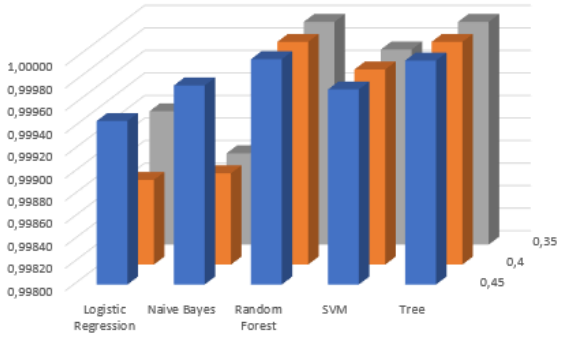
7.7. Eredmények

Az osztályozók teljesítményeinek mérésére és összehasonlítására az osztályozási pontosság (Accuracy) mérőszámokat használtam. Minden adathalmaznál a küszöbértékekhez kapcsolódó jellemzőkkel lett az osztályozók vizsgálva és a különbségeket oszlopdiagramon ábrázoltam, ahol a függőleges tengely az osztályozási pontosság mértékét (0-1 közzé eső szám, ahol az 1 érték a legjobb) ábrázolja, és minden küszöbértékhez külön sorban jelennek meg.

Az FTP-támadások esetében az osztályozók mindegyike jól teljesített, magas pontossági értékekkel és a lehető legmagasabb felidézési arányokkal rendelkezett szinte valamennyi jellemző alcsoport-osztályozó típuspár esetében. A Logisztikus regresszió és a Naive Bayes osztályozók tekintetében a kiválasztott küszöbérték növelésével a pontosság a tanító adathalmaznál csökkent (lásd 25. ábra). Az osztályozók teszt adathalmazzal történő kiértékelésekor azonban a Naive Bayes és a Logisztikus regresszió osztályozók esetében a küszöbérték növelésével a pontossági teljesítmény javult (lásd 26. ábra).

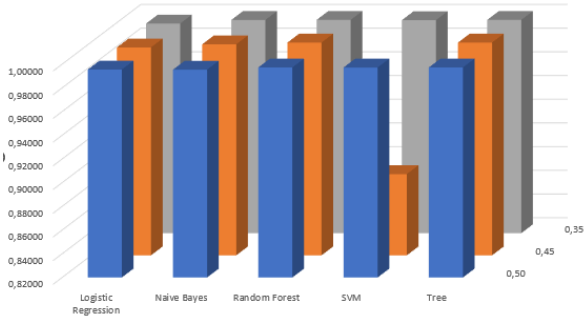


25. ábra Pontossági értékek FTP-támadás esetén a tanító adathalmazra

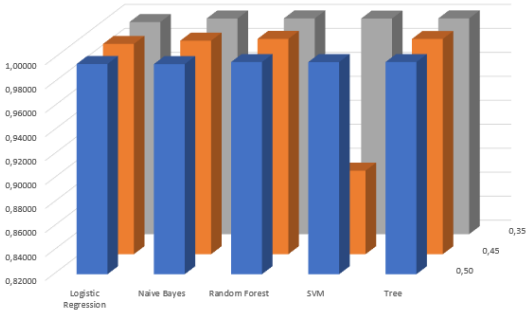


26. ábra Pontossági értékek FTP-támadás esetén a teszt adathalmazra

Az osztályozók az SSH-támadásokkal szemben is erős teljesítményt mutattak, az összes jellemzőcsoport esetében magas pontosság figyelhető meg. A kiválasztott jellemzők számának csökkentése a tanító adathalmazzal a pontosság javulásához vezetett, kivéve az SVM-alapú osztályozó esetében (lásd 27. ábra). Az osztályozók teszt adathalmazzal szembeni kiértékelésekor nagyon hasonló viselkedés mutatkozott (lásd 28. ábra).

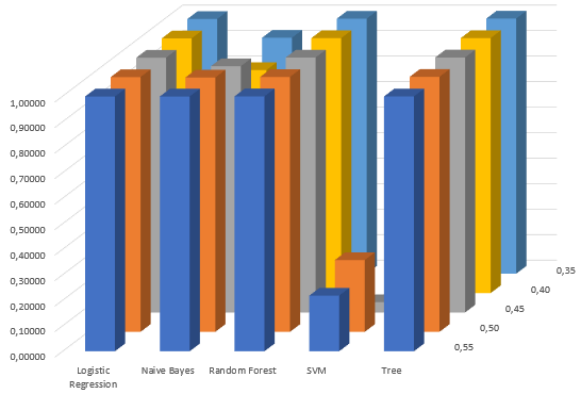


27. ábra Pontossági értékek SSH-támadás esetén a tanító adathalmazra

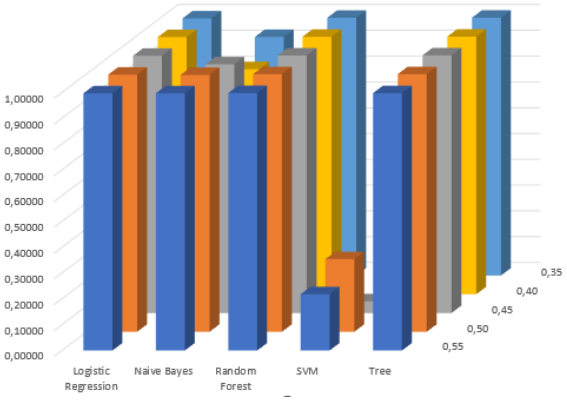


28. ábra Pontossági értékek SSH-támadás esetén a teszt adathalmazra

A webes támadások esetében az SVM osztályozó a többihez képest alacsony pontossági arányt nyújtott mind a tanító (lásd 29. ábra), mind a teszt (lásd 30. ábra) adathalmazok esetében. A Logisztikus regresszió, a Véletlen erdő és a Döntési fa alapú osztályozók azonban nagyon magas pontossággal képesek voltak sikeresen megjósolni a forgalom jellegét. Bár a Naive Bayes modell a 44 (a küszöbérték 0,35) és 34 (a küszöbérték 0,40) kiválasztott jellemző esetében csökkenő teljesítményt mutatott, eredményei nem maradtak el túlságosan.

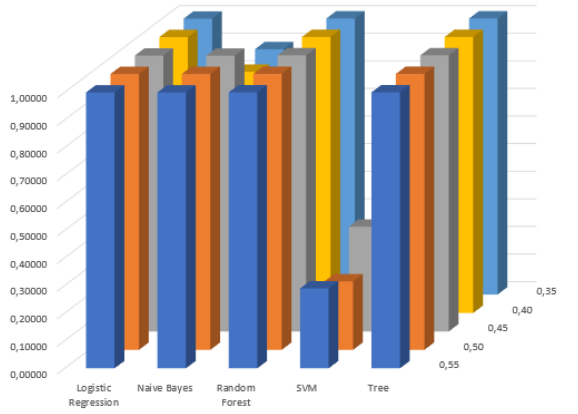


29. ábra Pontossági értékek WEB-támadás esetén a tanító adathalmazra

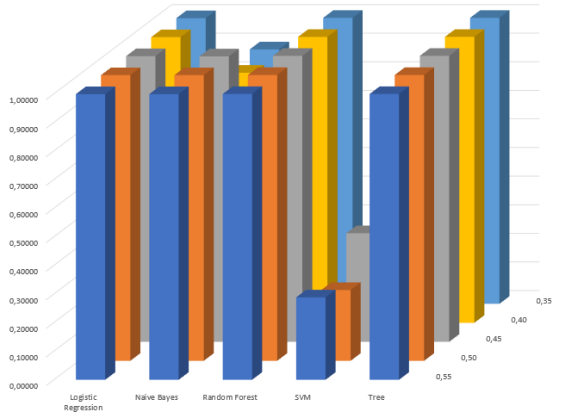


30. ábra Pontossági értékek WEB-támadás esetén a teszt adathalmazra

Az XSS-támadásokra tesztelt osztályozók közül az SVM-osztályozó a többihez képest alacsony pontossági arányt mutatott mind a tanító (lásd 31. ábra), mind a teszt adatkészlet (lásd 32. ábra) esetében. A Logisztikus regresszió, a Véletlen erdő és a Döntési fa alapú osztályozók kivételesen jól teljesítettek, nagyon magas pontossági aránnyal. Bár a Naive Bayes modell 0,40 és 0,35 küszöbértéknél csökkenő teljesítményt mutatott, eredményei még mindig versenyképesek voltak.

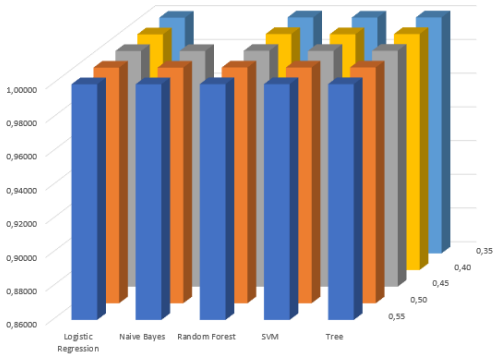


31. ábra Pontossági értékek XSS-támadás esetén a tanító adathalmazra

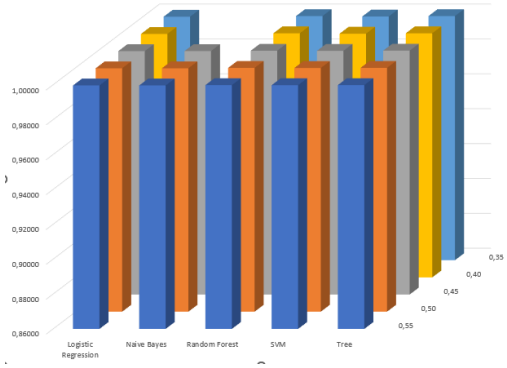


32. ábra Pontossági értékek XSS-támadás esetén a teszt adathalmazra

Az SQL Injection támadások esetében az öt osztályozó modell közül négy magas pontossági arányt mutatott mind a tanító (lásd 33. ábra.), mind a teszt adathalmaz (lásd 34. ábra) esetében. A Naive Bayes modell volt az egyetlen kivétel, amely 26 és 31 kiválasztott jellemző használata esetén kissé csökkenő teljesítményt mutatott, de még mindig 0,9 feletti pontossági értékeket ért el.



33. ábra Pontossági értékek SQL-támadás esetén a tanító adathalmazra



34. ábra Pontossági értékek SQL-támadás esetén a teszt adathalmazra

Ahhoz, hogy megtaláljam a legjobban teljesítő osztályozót a legkevesebb jellemzőszámok mellett, minden osztályozónál a tanító es teszt halmazok vizsgálatával létrejött teljesítmény értékek (Accuracy, Precision, Recall, F1) számtani átlaguk közül a legjobb értéket vettem. Így meghatároztam, minden adathalmaz esetén azt a küszöbértéket, ahol a legkisebb jellemzőszámmal jó osztályozási eredményt érek el. Minden egyes támadástípus esetében külön listát határoztam meg (lásd 13. táblázat) a küszöbértékekhez tartozó releváns jellemzőkkel. Minden egyes jellemzőt a sorszámával ábrázoltam. A táblázat minden sora azokat a jellemzőket tartalmazza, amelyek pontszáma nagyobb vagy egyenlő volt második cellában megadott küszöbértéknél. Ezáltal a jellemzők fontossági sorrendjével a támadások beazonosítását lehet hatékonyabbá tenni.

13. Táblázat A legkevesebb jellemzőszámokkal elérhető legjobb osztályozók támadástípusonként

Adathalmaz	Küszöbérték	A legjobb osztályozó	Jellemzők darabszáma	Jelemzők sorszáma
FTP	0,35	Véletlen erdő	8	*02,17,19,35,00,44,56,59*
SSH	0,35	Véletlen erdő	7	*00,02,17,19,57,56,59*
WEB	0,35	Döntési Fa	44	*16,20,10,49,66,67,35,38,56,64,34,27,07,09,11, 14,15,50,25,60,62,02,17,19,37,63,06,33,55,18, 58,04, 05,53,54,03,21,22,23,24,52,32,65,57*
SQL	0,40	Véletlen erdő	26	*05,26,53,56,25,02,17,19,35,16,18,27,28,34, 06,23,30,55,29,21,22,24,57,37,11,14*
XSS	0,45	Döntési Fa	10	*37,56,33,32,03,11,52,04,54,58*

2. TÉZIS

Átlagolással kapott jellemző-értékszámokhoz kapcsolódóan meghatároztam azokat a küszöbértékeket, amelyek segítségével beazonosítható a jellemzők azon minimális darabszámú csoportja, amely mellett jó teljesítményű osztályozók taníthatók be.

A tézisemhez tartozó publikációm a következő: [S3]

8. Súlyozott átlaggal végzett többtényezős módszer

Az első tézisemben sikeresen csökkentettem a probléma dimenzionalitását bizonyos jellemzők kizárásával. Ezek a jellemzők vagy egyértékű oszlopokkal rendelkeztek, vagy irreleváns információt tartalmaztak. Így 69-re szűkítettem a figyelembe veendő és tovább vizsgálandó jellemzők számát. Ezt követően az egyes jellemzők értékelésére különböző jellemzőkiválasztási módszereket alkalmaztam (lásd 5. fejezet). Az így kapott pontszámokat később normalizáltam és a számtani átlag segítségével összesítettem, így kaptam meg az egy értékű értékelést.

E jellemzőpontszámok és rangsorolási küszöbértékek alapján kiválasztottam a jellemzők részhalmazait, és különböző osztályozási módszerekkel teszteltem őket. Végül az egyes osztályozók teljesítményét a pontosság, a precizitás, a visszahívás és az F1 mérőszámok segítségével értékeltem mind a tanító, mind a tesztadathalmazokon. Azok az esetek, ahol a betanított osztályozók gyenge teljesítményt mutattak, arra ösztönöztek, hogy tovább vizsgáljam a súlyozott átlagolású megközelítést.

A súlyozott együttes rangsorolás a minták értékelésének széles körben használt megközelítése, amely lehetővé teszi az egyes komponensek differenciált értékelését azok jelentősége, fontossága, erőssége vagy bármely más, súlyként említett kritérium alapján. Több jellemző rangsorolási módszer hozzájárulásának figyelembevételével a jellemzők pontszámainak súlyozott átlagát az 24. egyenlet segítségével számoltam ki. Ez az egyenlet egy átfogó értékelési pontszámot ad, amely az együttes értékelését tükrözi.

$$R_{WA} = \frac{R_{IG} \cdot w_{IG} + R_{GR} \cdot w_{GR} + R_{SU} \cdot w_{SU} + R_{\chi^2} \cdot w_{\chi^2} + R_{\mathrm{Re}} \cdot w_{\mathrm{Re}} + R_{AN} \cdot w_{AN}}{w_{IG} + w_{GR} + w_{SU} + w_{\chi^2} + w_{\mathrm{Re}} + w_{AN}}, \qquad (24)$$

ahol az R_{WA} a jellemzőnek az együttes módszerrel számított pontszáma, R_{IG}, R_{GR}, R_{SU}, $R_{\chi 2}$, R_{Re}, R_{AN} az együttesbe bevont egyedi jellemző rangsorolási módszerek által kapott normalizált jellemző pontszámok, míg a w_{IG}, w_{GR}, w_{SU}, $w_{\chi 2}$, w_{Re}, w_{AN} az ezekhez a módszerekhez tartozó súlyokat jelentik.

Több jellemző rangsorolási módszer beépítésével és az egyes módszerek megfelelő súlyozásával az együttes megközelítés hatékonyan kihasználja az egyes technikák erősségeit, miközben enyhíti azok gyengeségeit.

8.1. Súlyoptimalizálás Taguchi DoE megközelítésével

A különböző pontszámok aggregálására a legegyszerűbb módszer az egyes jellemzők pontszámainak számtani átlaga, ahol minden egyes súly azonos. A különböző súlyok alkalmazása azonban néha olyan jellemzőpontszámokat eredményezhet, amelyek nagyobb mértékben járulnak hozzá egy javított jellemzőalcsoport kiválasztásához. Egy ilyen részhalmaz jobb osztályozási eredmények elérését teszi lehetővé. A súlyok optimális kombinációjának meghatározása kihívást jelentő feladat, mivel a pontszámok kiszámításából eredő különböző jellemzőgyűjtemények kiértékeléséhez jelentős időre van szükség. Ezért szükségessé válik a súlyok optimalizálása minimális számú próbálkozással.

Ez a felismerés vezetett a Taguchi-módszer néven ismert kísérlettervezési (DoE) technika felhasználásához. Ezt a Genichi Taguchi által az 1950-es években kifejlesztett megközelítést eredetileg a gyártóiparban alkalmazott minőségirányítás és tervezés [97] célozta. A Taguchi-módszer a különböző gyártási paraméterek termékminőségre gyakorolt hatásának azonosítására és optimalizálására törekedett. A gyártás során az optimális paraméterbeállítások azonosításával a Taguchi-módszer csökkenti az eltérésekkel szembeni érzékenységet és javítja a termék általános minőségét.

Az optimális paraméterbeállítás meghatározásához a Taguchi-módszer a "paramétertervezés" fogalmát alkalmazza. Ennek során a folyamatváltozókat előre meghatározott értéktartományokhoz rendelik, teszteket végeznek, és optimalizálják azokat. A kutatás során hat független változót kell kipróbálni, mindegyiket két szinten. Ezért a $L_8 2^7$ ortogonális tervezési tervet alkalmaztam. Minden egyes faktor esetében két szintet használtam, amelyeket 1 és 2 kóddal jelölem, ez látható a 14. táblázatban.

14. táblázat $L_8 2^7$ ortogonális táblázat

	WIG	WGR	WSU	WKhi	WRe	WAN
1	1	1	1	1	1	1
2	1	1	1	2	2	2
3	1	2	2	1	1	2
4	1	2	2	2	2	1
5	2	1	2	1	2	1
6	2	1	2	2	1	2
7	2	2	1	1	2	2
8	2	2	1	2	1	1

8.2. Megvalósítás

A súlykeresési tér jobb, minimális kísérletekkel történő feltárásának megkönnyítése érdekében a súlyváltozók (a DoE-ban faktoroknak nevezett) két szintjéhez a kiválasztott DoE-designban szereplő súlyváltozókhoz 0,0233 és 0,2336 súlyértéket rendeltem. E választás hátterében az állt, hogy egymástól jelentősen távol eső értékeket használtam.

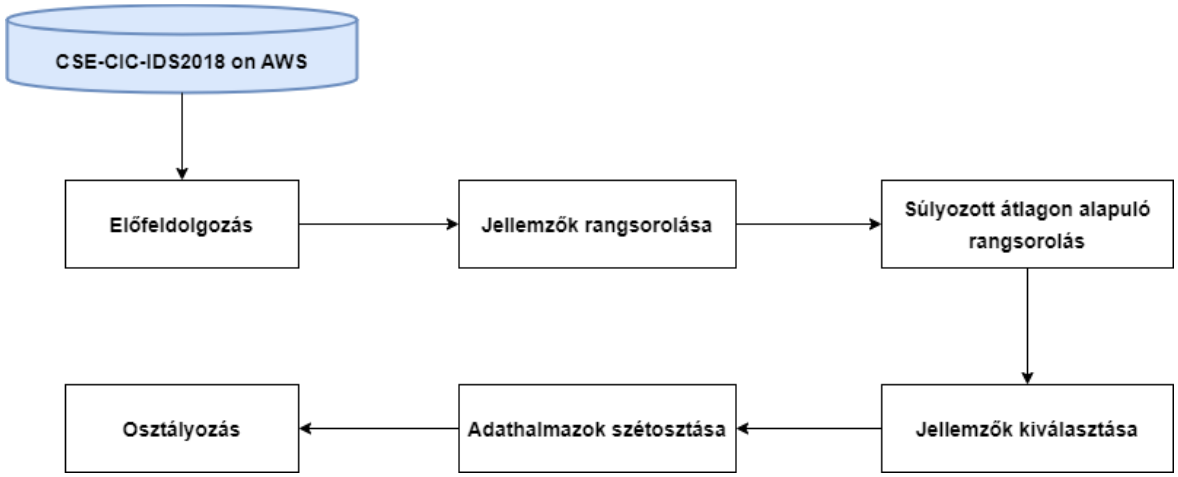
15. táblázat Meghatározott súlyértékek

	WIG	WGR	WSU	WKhi	WRe	WAN
1	0,023256	0,023256	0,023256	0,023256	0,023256	0,023256
2	0,023256	0,023256	0,023256	0,232558	0,232558	0,232558
3	0,023256	0,232558	0,232558	0,023256	0,023256	0,232558
4	0,023256	0,232558	0,232558	0,232558	0,232558	0,023256
5	0,232558	0,023256	0,232558	0,023256	0,232558	0,023256
6	0,232558	0,023256	0,232558	0,232558	0,023256	0,232558
7	0,232558	0,232558	0,023256	0,023256	0,232558	0,232558
8	0,232558	0,232558	0,023256	0,232558	0,023256	0,023256

A kísérletekhez szükséges jelentős idő miatt a kimerítő keresés elvégzése nem volt kivitelezhető. Minden egyes jellemzőhöz nyolc súlycsoportot határoztam meg a 15. táblázat alapján. A súlyok alkalmazása után kapott pontszámok a Mellékletben lévő A.12-16 táblázatban találhatók.

Elsősorban azokra az esetekre irányítottam a figyelmet, ahol a korábbi, számtani átlagot használó vizsgálat nem hozott kielégítő eredményt. Itt két célom volt:

1. vagy kevesebb jellemzőt tartalmazó jellemzőkészletek azonosítása az eredeti osztályozási teljesítmény megtartása mellett, vagy
2. olyan jellemzőkészletek keresése, amelyek osztályozási pontosság (Accuracy), a pontosság (Precision), a fedés (Recall) és az F-mérték (F1) mint teljesítménymérők segítségével javíthatják az osztályozási teljesítményt. A folyamat lépéseit a 30. ábra vázolja.



35. ábra Súlyozott átlag módszer folyamata

A hipotézis az volt, hogy egy súlyozási mechanizmus alkalmazása javíthatja a korábbi megközelítést, amely több egyedi jellemzőpontozási-módszer felhasználását és a normalizált pontszámok számtani átlagának kiszámítását foglalta magában.

8.3. Eredmények értékelése

Az előzőekben bemutatott súlyozott átlag módszerrel az FTP-adatkészlet esetében a figyelembe vett jellemzők számát 8-ról 5-re lehetett csökkenteni, miközben mindhárom osztályozási módszer kiváló teljesítményt nyújtott (lásd 16. táblázat). Hasonlóképpen, az SSH-adatkészlet esetében is hasonló mintázatot eredményezett. Ebben az esetben a jellemzők számát 7-ről 6-ra lehetett csökkenteni, miközben ugyanolyan vagy potenciálisan jobb teljesítményt lehetett elérni (lásd 17. táblázat).

16. táblázat Eredmények az FTP adathalmazra súlyozott átlaggal

Adathalmaz	Átlag típus	Jellemzők száma	Osztályozó	Accuracy	Precision	Recall	F1
Tanító	normál	8	Decision Tree	*1.00000*	*1.00000*	*1.00000*	*1.00000*
	súlyozott	5	Decision Tree	*0.99999*	*0.99997*	*1.00000*	*0.99999*
	normál	8	Random Forest	*1.00000*	*1.00000*	*1.00000*	*1.00000*
	súlyozott	5	Random Forest	*1.00000*	*1.00000*	*1.00000*	*1.00000*
	normál	8	SVM	*0.99973*	*0.99881*	*1.00000*	*0.99941*
	súlyozott	5	SVM	*0.99990*	*0.99956*	*1.00000*	*0.99978*
Tesztelő	normál	8	Decision Tree	*0.99999*	*0.99995*	*1.00000*	*0.99997*
	súlyozott	5	Decision Tree	*0.99997*	*0.99995*	*0.99990*	*0.99992*
	normál	8	Random Forest	*1.00000*	*1.00000*	*1.00000*	*1.00000*
	súlyozott	5	Random Forest	*1.00000*	*1.00000*	*1.00000*	*1.00000*
	normál	8	SVM	*0.99973*	*0.99881*	*1.00000*	*0.99941*
	súlyozott	5	SVM	*0.99988*	*0.99948*	*1.00000*	*0.99974*

17. táblázat Eredmények az SSH adathalmazra súlyozott átlaggal

Adathalmaz	Átlag típus	Jellemzők száma	Osztályozó	Accuracy	Precision	Recall	F1
Tanító	normál	7	Decision Tree	*0.99999*	*0.99997*	*1.00000*	*0.99999*
	súlyozott	6	Decision Tree	*0.99999*	*0.99997*	*1.00000*	*0.99999*
	normál	7	Random Forest	*0.99999*	*0.99997*	*1.00000*	*0.99999*
	súlyozott	6	Random Forest	*1.00000*	*1.00000*	*1.00000*	*1.00000*
	normál	7	SVM	*0.99979*	*0.99928*	*0.99979*	*0.99953*
	súlyozott	6	SVM	*0.99993*	*0.99989*	*0.99979*	*0.99984*
Tesztelő	normál	7	Decision Tree	*1.00000*	*1.00000*	*1.00000*	*1.00000*
	súlyozott	6	Decision Tree	*0.99996*	*0.99984*	*1.00000*	*0.99992*
	normál	7	Random Forest	*0.99999*	*0.99995*	*1.00000*	*0.99997*
	súlyozott	6	Random Forest	*0.99996*	*0.99984*	*1.00000*	*0.99992*
	normál	7	SVM	*0.99985*	*0.99947*	*0.99984*	*0.99965*
	súlyozott	6	SVM	*0.99996*	*1.00000*	*0.99984*	*0.99992*

A Web-adatkészlet esetében a szükséges jellemzők száma 44-ről 13-ra csökkenthető (lásd 18. táblázat). Az SQL-adatkészlet esetében a súlyozott átlaggal kiválasztott jellemzőkkel az SVM osztályozás javult és a szükséges jellemzők száma 26-ról 7-re csökkent (lásd 20. táblázat). Hasonlóképpen az XSS-adatkészlet esetében a szükséges jellemzők száma 10-ről 2-re csökkenthető (lásd 19. táblázat). Továbbá, míg az egyszerű átlag alapú megoldás az SQL és a Web adathalmazok esetében az SVM osztályozóval gyenge eredményeket adott mind a tanító-, mind a tesztadathalmazok esetében, az új megközelítés jelentős javulást eredményezett a teljesítménymutatókban.

18. táblázat Eredmények az WEB adathalmazra súlyozott átlaggal

Adathalmaz	Átlag típus	Jellemzők száma	Osztályozó	Accuracy	Precision	Recall	F1
Tanító	normál	44	Decision Tree	*0.99994*	*0.98997*	*0.96890*	*0.97932*
	súlyozott	13	Decision Tree	*0.99978*	*0.97967*	*0.86743*	*0.92014*
	normál	44	Random Forest	*0.99963*	*0.99142*	*0.75614*	*0.85794*
	súlyozott	13	Random Forest	*0.99963*	*1.00000*	*0.74468*	*0.85366*
	normál	44	SVM	*0.32725*	*0.00077*	*0.35516*	*0.00154*
	súlyozott	13	SVM	*0.99886*	*0.99886*	*0.99886*	*0.99849*
Tesztelő	normál	44	Decision Tree	*0.99972*	*0.93819*	*0.96890*	*0.95330*
	súlyozott	13	Decision Tree	*0.99948*	*0.94982*	*0.86743*	*0.90676*
	normál	44	Random Forest	*0.99928*	*0.99784*	*0.75614*	*0.86034*
	súlyozott	13	Random Forest	*0.99925*	*1.00000*	*0.74468*	*0.85366*
	normál	44	SVM	*0.32654*	*0.00154*	*0.35516*	*0.00307*
	súlyozott	13	SVM	*0.99771*	*0.99772*	*0.99771*	*0.99698*

19. táblázat Eredmények az XSS adathalmazra súlyozott átlaggal

Adathalmaz	Átlag típus	Jellemzők száma	Osztályozó	Accuracy	Precision	Recall	F1
Tanító	normál	10	Decision Tree	*0.99998*	*1.00000*	*0.96957*	*0.98455*
	súlyozott	2	Decision Tree	*0.99994*	*0.93966*	*0.94783*	*0.94372*
	normál	10	Random Forest	*0.99999*	*1.00000*	*0.97391*	*0.98678*
	súlyozott	2	Random Forest	*0.99995*	*0.95217*	*0.95217*	*0.95217*
	normál	10	SVM	*0.37911*	*0.00046*	*0.51304*	*0.00091*
	súlyozott	2	SVM	*0.99945*	*0.99890*	*0.99945*	*0.99917*
Tesztelő	normál	10	Decision Tree	*0.99996*	*0.99554*	*0.96957*	*0.98238*
	súlyozott	2	Decision Tree	*0.99992*	*0.98198*	*0.94783*	*0.96460*
	normál	10	Random Forest	*0.99997*	*0.99556*	*0.97391*	*0.98462*
	súlyozott	2	Random Forest	*0.99993*	*0.98206*	*0.95217*	*0.96689*
	normál	10	SVM	*0.37972*	*0.00091*	*0.51304*	*0.00182*
	súlyozott	2	SVM	*0.99890*	*0.99780*	*0.99890*	*0.99835*

20. táblázat Eredmények az SQL adathalmazra súlyozott átlaggal

Adathalmaz	Átlag típus	Jellemzők száma	Osztályozó	Accuracy	Precision	Recall	F1
Tanító	normál	26	Decision Tree	*0.99999*	*1.00000*	*0.95402*	*0.97647*
	súlyozott	7	Decision Tree	*0.99999*	*1.00000*	*0.95402*	*0.97647*
	normál	26	Random Forest	*0.99998*	*1.00000*	*0.91954*	*0.95808*
	súlyozott	7	Random Forest	*0.99999*	*1.00000*	*0.96552*	*0.98246*
	normál	26	SVM	*0.99987*	*1.00000*	*0.37931*	*0.55000*
	súlyozott	7	SVM	*0.99988*	*0.99988*	*0.99988*	*0.99986*
Tesztelő	normál	26	Decision Tree	*0.99998*	*1.00000*	*0.95402*	*0.97647*
	súlyozott	7	Decision Tree	*0.99999*	*0.98824*	*0.96552*	*0.97674*
	normál	26	Random Forest	*0.99997*	*1.00000*	*0.91954*	*0.95808*
	súlyozott	7	Random Forest	*1.00000*	*1.00000*	*0.97701*	*0.98837*
	normál	26	SVM	*0.99974*	*1.00000*	*0.37931*	*0.55000*
	súlyozott	7	SVM	*0.99977*	*0.99977*	*0.99977*	*0.99972*

Minden egyes adathalmazhoz a meghatározott csökkenett számú jellemzők a 21. táblázatban láthatóak.

21. táblázat Súlyozott átlaggal kapott jellemzők csoportja

Adathalmaz	Súlyozott átlaggal kapott jellemzők csoportja
FTP	*19, 02, 17, 56, 59*
SSH	*33, 32, 00, 56, 57, 59*
WEB	*32, 56, 07, 50, 09, 11, 65, 14, 37, 53, 05, 58, 57*
XSS	*57, 56*
SQL	*56, 43, 47, 57, 37, 11, 14*

A súlyozott módszerrel a meghatározott jellemzőcsoportokban szereplő jellemzők tényleges elnevezése és funkciói a 22. táblázatban láthatóak. A meghatározott jellemzők segítségével már tényleges adatok, információk konfigurálhatóak egy IDS rendszer szenzorai számára.

22. táblázat A meghatározott releváns jellemzők tulajdonságai

Sorszám	Megnevezés	A jellemzők működésbeni jelentése
00	*Dst Port*	A célállomás portja, ahová az adatcsomagokat küldik.
02	*Flow Duration*	Az adatfolyam időtartama az első és az utolsó adatcsomag között.
05	*TotLen Fwd Pkts*	Az összes előre irányuló (forrás felé) adatcsomag mérete összesen.
07	*Fwd Pkt Len Max*	Az előre irányuló adatcsomagok közül a leghosszabb csomag mérete.

09	*Fwd Pkt Len Mean*	Az előre irányuló adatcsomagok átlagos mérete.
11	*Bwd Pkt Len Max*	A visszafelé irányuló adatcsomagok közül a leghosszabb csomag mérete.
14	*Bwd Pkt Len Std*	A visszafelé irányuló adatcsomagok méretének szórása.
17	*Flow IAT Mean*	Az adatfolyam közötti időközök átlagos hossza.
19	*Flow IAT Max*	Az adatfolyam közötti időközök maximális hossza.
32	*Fwd Header Len*	Az előre irányuló adatcsomagok fejlécének mérete.
33	*Bwd Header Len*	A visszafelé irányuló adatcsomagok fejlécének mérete.
37	*Pkt Len Max*	A legnagyobb adatcsomag mérete az összes csomag közül.
43	*RST Flag Cnt*	A RST zászlóval ellátott adatcsomagok száma.
47	*ECE Flag Cnt*	Az ECE zászlóval ellátott adatcsomagok száma.
50	*Fwd Seg Size Avg*	Az előre irányuló adatcsomagok átlagos szegmensmérete.
53	*Subflow Fwd Byts*	Az előre irányuló adatok összmérete az alközlési áramlatokban.
56	*Init Fwd Win Byts*	Az előre irányuló kezdeti ablakméret a TCP kapcsolatban.
57	*Init Bwd Win Byts*	A visszafelé irányuló kezdeti ablakméret a TCP kapcsolatban.
58	*Fwd Act Data Pkts*	Az előre irányuló effektív adatcsomagok száma.
59	*Fwd Seg Size Min*	Az előre irányuló adatcsomagok minimális szegmensmérete.
65	*Idle Std*	Az időtartamok közötti inaktivitás időtartamok szórása az adatfolyamban.

3. TÉZIS

Megmutattam, hogy a különböző módszerekkel előállított jellemzőértékszámok súlyozásával képzett általános mérőszám alkalmazásával ugyanolyan vagy jobb osztályozási eredmények érhetőek el csökkentett számú figyelembe vett jellemző mellett.

A tézisemhez tartozó publikációm a következő: [S4]

9. Osztályozás Catboost algoritmus segítségével

A korábban bemutatott és használt Naive Bayes, Decision Tree, Random Forest, Logistic Regression és SVM olyan osztályozó és modellező módszerek, amelyek hosszú ideje részei a gépi tanulás és adatbányászat eszköztárának. Kutatásom folytatásaként meg kívántam vizsgálni, hogy az utóbbi időben sok területen sikerrel alkalmazott Gradient Boost megközelítés egyik megvalósítása, a CatBoost algoritmus képes-e a hagyományos osztályozókkal közel azonos vagy azoknál jobb osztályozási eredményeket nyújtani.

9.1. A Catboost algoritmus

A CatBoost egy nyílt forrású gépi tanulási algoritmus, amelyet kifejezetten kategória típusú változók kezelésére terveztek. A kategóriai változók olyan jellemzők, amelyek nem numerikus értékeket vesznek fel, hanem diszkrét címkéket vagy kategóriákat jelölnek. Az algoritmus egyedi tulajdonsága, hogy hatékonyan kezeli a kategóriai változókat anélkül, hogy előzetes feldolgozási vagy átalakítási lépéseket igényelne. Ezáltal időt takarít meg a fejlesztőknek és a adattudósoknak, akiknek kevesebb előkészítő munkát kell végezniük az adatokon.

A CatBoost algoritmus alapjául a döntési fákat használja, amelyeket a gradiens függvény módszerrel (gradient boosting) kombinál. A gradiens függvény módszer egy olyan gépi tanulási technika, amely a modell iteratív fejlesztésével próbálja minimalizálni a hibát. A CatBoost különösen hatékony ebben a folyamatban, mivel adaptív mesterséges intelligenciát használ a modell paramétereinek automatikus beállításához. Az algoritmusnak számos előnye van. Az egyik ilyen előnye a magas pontosság, amelyet a modell képes elérni.

A CatBoost különleges kezelést biztosít a kategóriai változóknak, ezáltal növelve a modellek pontosságát és teljesítményét. Emellett ellenálló az adatok zajával szemben, és kevésbé érzékeny a túltanulásra, ami gyakran problémát okoz más algoritmusoknál. A CatBoost kiválóan alkalmazható különböző gépi tanulási feladatokban, mint például osztályozás és regresszió. Továbbá támogatja a többosztályos osztályozást, és lehetővé teszi a jellemzők fontosságának értékelését.

Az algoritmus gyakran használják ipari projektekben, és nagyon jó eredményeket ér el a versenyképes gépi tanulási feladatokban. A CatBoost széles körben támogatott a gépi tanulás közösség által, és számos dokumentáció és példa áll rendelkezésre, hogy segítse az új felhasználókat a használatában [98].

A CatBoost algoritmus működése:

- Adatok előkészítése: Az adatokat megfelelő formára kell hozni a további feldolgozáshoz. A CatBoost képes kezelni a kategorikus változókat és automatikusan feldolgozza őket, így nem szükséges külön kódolni vagy előfeldolgozni őket.
- Döntési fák építése: A CatBoost döntési fákat használ az adatok osztályozásához. Az algoritmus különböző technikákat alkalmaz a döntési fák építése során, például a szimmetrikus fa módszert vagy a helyettesítő változókat. Ez a rugalmas faépítési eljárás segít abban, hogy a CatBoost jobban kezelje a kategorikus változókat és kivédekezze a túltanulást.
- Gradient boosting: A CatBoost algoritmus a gradient boosting technikáját alkalmazza a modell továbbfejlesztésére és finomhangolására. A gradient boosting során iteratívan hozzáadunk újabb döntési fákat a modellhez. Minden iterációban a hibákra koncentrál, és azokra a területekre fókuszál, ahol a modell rosszul teljesít. Ez a fokozatos finomhangolás lehetővé teszi a CatBoost számára, hogy jobb és pontosabb predikciókat hozzon létre.
- Regularizáció és tuning: A CatBoost számos beállítással és paraméterrel rendelkezik, amelyek lehetővé teszik a modell finomhangolását és a túltanulás elkerülését. Például a paraméterek segítségével beállíthatjuk a fák mélységét, a tanulási sebességet vagy a regularizációs tényezőket.

A CatBoost algoritmus jellemzői és előnyei:

- Kategóriák kezelése: A CatBoost algoritmus kifejezetten hatékonyan kezeli a kategorikus változókat. Nem szükséges előzetesen kódolni vagy előfeldolgozni őket, mivel az algoritmus automatikusan megoldja a kategóriák kezelését.
- Robusztus a hiányzó adatokkal: A CatBoost képes kezelni a hiányzó adatokat. Nem szükséges külön stratégiát alkalmazni a hiányzó adatok pótlására vagy figyelmen kívül hagyására.
- Nagy teljesítmény és skálázhatóság: A CatBoost rendkívül gyors és skálázható. Jól kezel nagy adatkészleteket és támogatja a többszálú feldolgozást, így gyorsan tanul és prediktál, még nagyobb adatkörnyezetekben is.
- Robusztus a zajos adatokkal: A CatBoost ellenálló a zajos adatokkal szemben, amelyek gyakran előfordulnak a valós világban. Az algoritmusnak beépített zajtűrő

mechanizmusa van, amely segít csökkenteni a zaj hatását és javítja a predikciók megbízhatóságát.

- Interpretálhatóság: A CatBoost algoritmus lehetővé teszi a modell interpretálhatóságát és az attribútumok fontosságának meghatározását. Ez segít megérteni, hogy mely változók befolyásolják leginkább a predikciókat.

Összességében a CatBoost egy hatékony és sokoldalú gépi tanulási algoritmus, amely kategóriai változók kezelésére specializálódik. A kategóriai változók hatékony kezelése, a pontosság és a zajállóság miatt a CatBoost népszerű választás a gépi tanulási feladatok megoldására [99].

Az egyes minták jellemzőinek numerikus átalakításakor először a minta célértékét számítja ki a minta előtt, majd hozzáadja a megfelelő súlyt és prioritást,

$$x_k^i = \frac{\sum_{j=1}^{n} \left\{ x_j^i = x_k^i \right\} \cdot y_i + ap}{\sum_{j=1}^{n} \left\{ x_j^k = x_k^i \right\} + a}, \tag{25}$$

ahol p a hozzáadott előzetes értéket és a nullánál nagyobb súlykoefficienst jelenti. Egy a prioritás értéket adunk hozzá, hogy jelentősen csökkentsük az alacsony frekvenciájú jellemzők által okozott zajpontokat, hogy hatékonyan minimalizáljuk a modell túlillesztését és javítsuk az általánosítási képességet [100].

9.2. Megvalósítás

Az algoritmus tanítási és tesztelési folyamata Python program segítségével történt meg. A 25. képletben szereplő értékek (x, a, p) az algoritmus három fontos paramétere:

- **x**: A bemeneti tulajdonságok (features) száma.
- **a**: A fák számának beállítása. Ez a paraméter meghatározza, hány döntési fa alkotja majd a modellt.
- **p**: A döntési fák mélységének beállítása. Ez megadja, hogy az egyes döntési fák mekkora mélységig nőhetnek.

Ezeket az értékeket a CatBoost paraméterekkel állíthatjuk be a Python programban, amikor létrehozzuk és betanítjuk a modellt. A CatBoostClassifier osztály az 'iterations', 'learning_rate' és 'depth' paraméterekkel inicializálódik. Az 'iterations' paraméter az iterációk számát, a

'learning_rate' a képzési sebességet, a 'depth' pedig a fa mélységét adja meg. A 'fit()' függvény a modellt a képzési adatokon képzi ki, majd a 'predict()' függvény a tesztadatokra vonatkozó előrejelzések elkészítésére szolgál.

Minden támadás típus esetében a CatBoost osztályozóval ugyanazokat a tanító- és tesztadathalmazokat használtam, valamint olyan jellemzőket, amelyeket korábban az egyes jellemzők pontszámainak súlyozott átlaggal történő összesítésével azonosítottam. Az így kapott helyes és tévesen besorolt esetek számát bemutató tévesztési mátrixok a 23 - 27. táblázatok tartalmazzák. A tévesztési mátrix (más néven konfúziós mátrix) egy olyan eszköz, amelyet osztályozó algoritmusok teljesítményének értékelésére használnak. A 2x2-es tévesztési mátrix az osztályozás során előforduló különböző eredményeket és hibákat összegzi.

23. táblázat Tévesztési Mátrixok az FTP halmaz vizsgálatánál

n= 171433	*1*	*0*	
1	99,999%	0,001%	*132762*
0	0,000%	100,000%	*38671*
	132761	*38672*	

tanító halmaz

n= 85716	*1*	*0*	
1	99,998%	0,002%	*66381*
0	0,000%	100,000%	*19335*
	66380	*19336*	

teszt halmaz

24. táblázat Tévesztési Mátrixok az SSH halmaz vizsgálatánál

n= 170280	*1*	*0*	
1	99,999%	0,001%	*132762*
0	0,000%	100,000%	*37518*
	132761	*37519*	

tanító halmaz

n= 85140	*1*	*0*	
1	99,994%	0,006%	*66381*
0	0,000%	100,000%	*18759*
	66377	*18763*	

teszt halmaz

25. táblázat Tévesztési Mátrixok az SQL halmaz vizsgálatánál

n= 417068	*1*	*0*	
1	99,999%	0,001%	*416981*
0	2,299%	97,701%	*87*
	416979	*89*	

tanító halmaz

n= 208577	*1*	*0*	
1	100,000%	0,000%	*208490*
0	2,299%	97,701%	*87*
	208491	*86*	

teszt halmaz

26. táblázat Tévesztési Mátrixok az XSS halmaz vizsgálatánál

n= 417211	*1*	*0*	
1	99,987%	0,013%	*416981*
0	6,957%	93,043%	*230*
	416941	*270*	

tanító halmaz

n= 208720	*1*	*0*	
1	99,991%	0,009%	*208490*
0	6,957%	93,043%	*230*
	208488	*232*	

teszt halmaz

27. táblázat Tévesztési Mátrixok a WEB halmaz vizsgálatánál

n= 417592	*1*	*0*	
1	100,000%	0,000%	*416981*
0	25,368%	74,632%	*611*
	417135	*457*	

tanító halmaz

n= 209101	*1*	*0*	
1	100,000%	0,000%	*208490*
0	25,368%	74,632%	*611*
	208644	*457*	

teszt halmaz

9.3. Eredmények Catboost algoritmussal

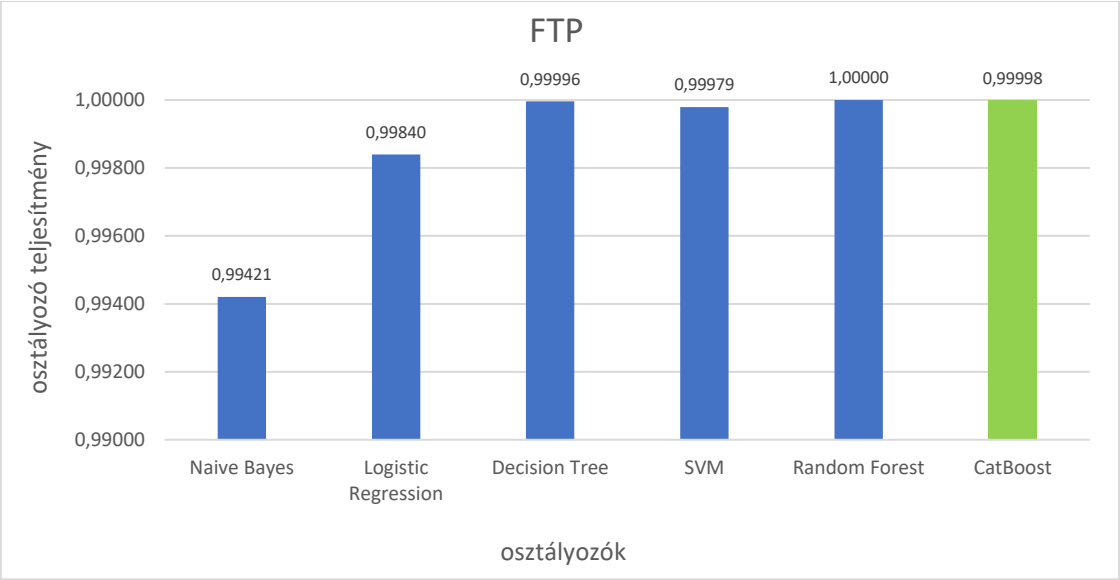
Mindegyik adathalmaz esetében a tanító és teszt halmazok vizsgálatával létrejött Accuracy, Precision, Recall, F1 teljesítmény értékek számtani átlagát figyelembe véve határoztam meg egy átlagos osztályozási teljesítmény számot, amely 0-1 közé eső szám, ahol az 1 a legjobb teljesítményt mutatja. Ez alapján létrejött egy összehasonlítás (lásd 26. táblázat) a Naive Bayes, Logisztikus regresszió, Tartóvektor-gép, Döntési fa, Véletlen erdő osztályozó algoritmusok és a CatBoost algoritmus között.

28. Táblázat A Catboost összehasonlítása

Adathalmaz	Jellemzők száma	Osztályozó	Osztályozó teljesítmények átlaga
FTP	5	Naive Bayes	0,9942
		Logisztikus regresszió	0,9984
		Döntési fa	1,0000
		Tartóvektor-gép	0,9998
		Véletlen erdő	1,0000
		CatBoost	**1,0000**
SSH	6	Naive Bayes	0,9999
		Logisztikus regresszió	0,9940
		Döntési fa	1,0000
		Tartóvektor-gép	0,9999
		Véletlen erdő	1,0000
		CatBoost	**1,0000**
SQL	7	Naive Bayes	0,2499
		Logisztikus regresszió	0,5252
		Döntési fa	0,9826
		Tartóvektor-gép	0,7323
		Véletlen erdő	0,9913
		CatBoost	**0,9694**
XSS	2	Naive Bayes	0,2498
		Logisztikus regresszió	0,2498
		Döntési fa	0,9657
		Tartóvektor-gép	0,3183
		Véletlen erdő	0,9697
		CatBoost	**0,9197**
WEB	13	Naive Bayes	0,5017
		Logisztikus regresszió	0,2494
		Döntési fa	0,9363
		Tartóvektor-gép	0,2071
		Véletlen erdő	0,8994
		CatBoost	**0,8994**

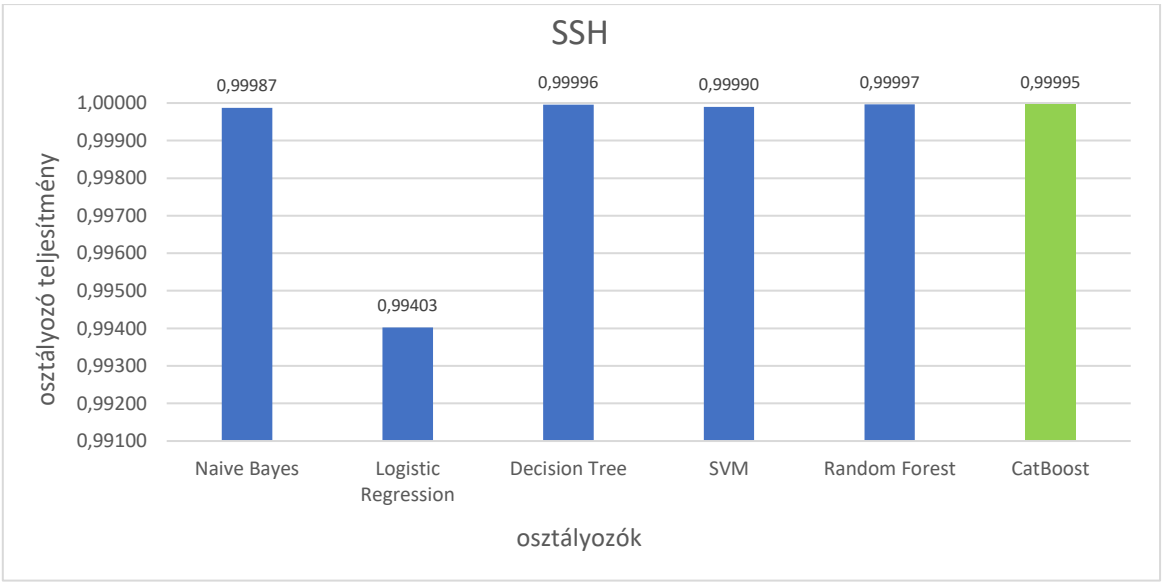
Az osztályozók részletes teljesítmény értékei megtalálhatóak a Melléklet A.17-es táblázatában.

Az FTP halmaz esetében jól látható, hogy a Catboost algoritmus közel azonos eredményt ért el a legjobban teljesítő (véletlen erdő) osztályozó algoritmussal. Minden egyéb esetben jobban teljesített. Három osztályozónál (Logistikis regresszió, Döntési fa, Tartóvektor-gép) minimálisan jobb eredményt ért el, egy esetben (Naive Bayes) nagyobb mértékben jobban teljesített.



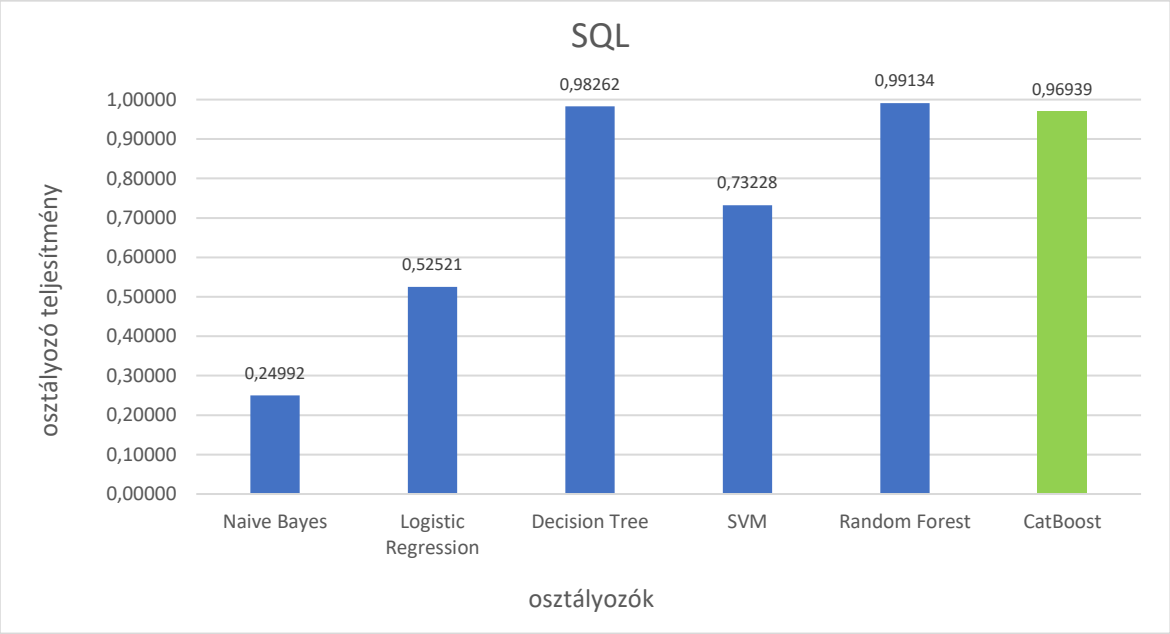
36. ábra Catboost összehasonlítása az FTP halmaznál

Az SSH adathalmaz vizsgálatánál az az eredmény született, hogy egy osztályozónál (Logistic Regression) nagyobb mértékben jobb volt a teljesítménye, a többi algoritmushoz képest minimális eltérésű a Catboost teljesítménye.



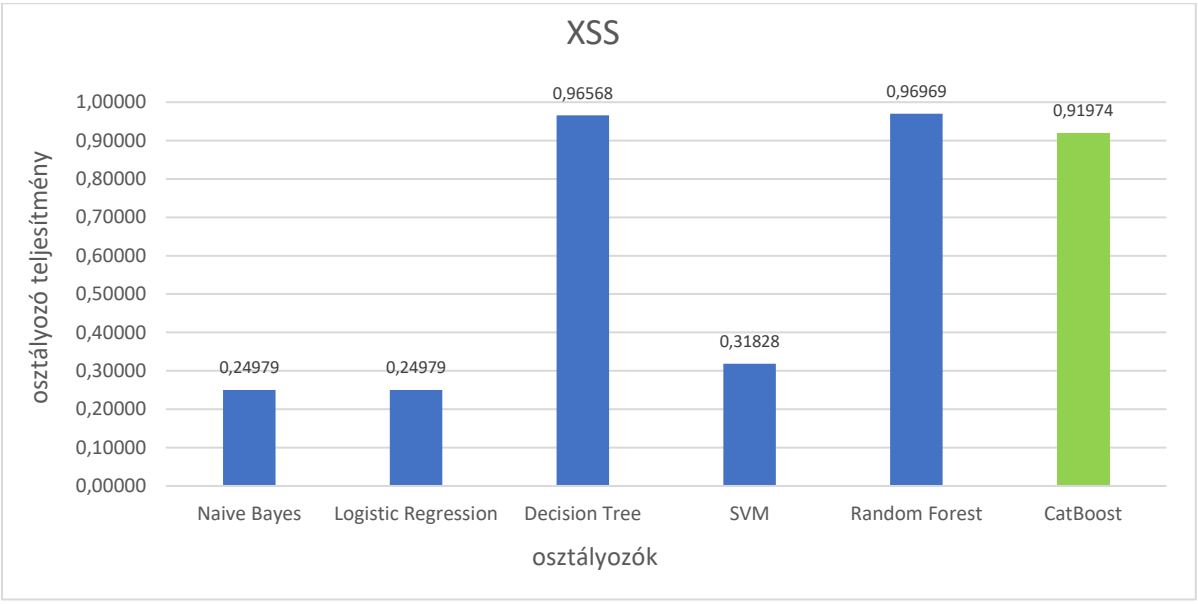
37. ábra Catboost összehasonlítása az SSH halmaznál

Az SQL halmaznál látható, hogy a 2 legjobb eredményhez (DT, RF) képest csak miminálisan csökkent a teljesítménye a Catboost algoritmusnak, és 3 másik osztályozónál (NB, LR, SVM) magasabb eredményt ér el.

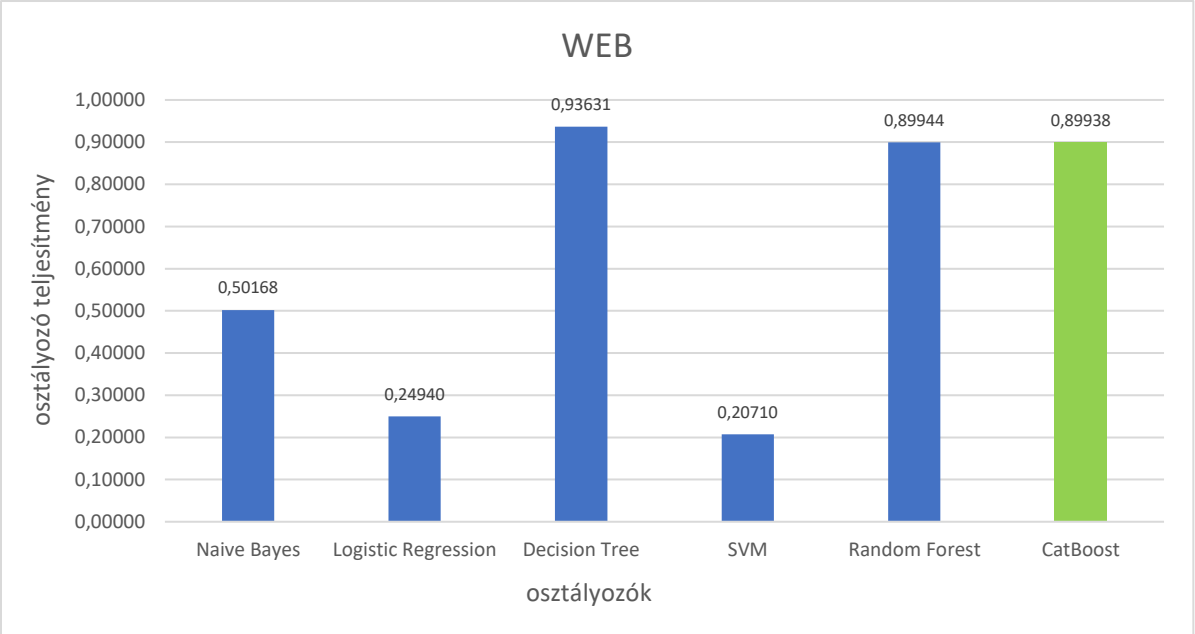


38. ábra Catboost összehasonlítása az SQL halmaznál

Az XSS és WEB halmaz esetén 3 osztályozóhoz képest (NB, LR, SVM) nagymértékben javult a teljesítménye, és 2 osztályozóhoz képest (DT, RF) minimálisan teljesített alul.



39. ábra Catboost összehasonlítása az XSS halmaznál



40. ábra Catboost összehasonlítása a WEB halmaznál

A vizsgálat kimutatta, hogy a legtöbb esetben a CatBoost algoritmus alkalmazása legalább ugyanolyan, esetenként jobb osztályozási teljesítményt eredményezett a kiválasztott támadások esetében, mint a korábban tesztelt alapértelmezett osztályozó típusok. Bár a CatBoost osztályozási teljesítménymutatói nem mindig voltak jobbak, de a CatBoost nagy előnye volt a jelentősen gyorsabb betanítási idő, amely átlagosan egy tizede volt az alapszintű osztályozókénak. Így elmondható, hogy a CatBoost algoritmus hatékony és gyors módja lehet az anomália alapú IDS-rendszerek fejlesztésének és működtetésének.

4. TÉZIS

Megmutattam, hogy a CatBoost algoritmus alkalmazásával legalább olyan jó osztályozási eredmények érhetőek el bizonyos hálózati támadások esetében, mint a Logisztikus regresszió, Naive Bayes, Tartóvektor-gép, Döntési fa, Véletlen erdő osztályozótípusok használata mellett.

A tézisemhez tartozó publikációm a következő: [S5]

10. Új tudományos eredmények összefoglalása

Kutatásom kezdeti szakaszában a különböző IDS-rendszerek alapvető jellemzőit és funkcióit vizsgáltam [S15] [S11]. Ezt követően figyelmemet az anomália alapú IDS-rendszerekre irányítottam, különös tekintettel az osztályozó moduljuk tanítási folyamatára. Ez a folyamat jellemzően nagy adatminták felhasználását jelenti, amelyek mind jóindulatú, mind rosszindulatú forgalmi adatokat tartalmaznak. Kutatásom során több adathalmazt is megvizsgáltam, végül találtam egy megfelelőt (CSE-CIC-IDS2018 az AWS-en), amely nemcsak megfelelt a kritériumoknak, hanem friss is volt, így ideális a tanításhoz.

Az adatkészlet kiválasztása és számos előfeldolgozási lépés végrehajtása után a vizsgálatom a jellemzőválasztásimódszerek köré összpontosult. Itt elsődleges eredményem a hat különböző módszerrel kapott normalizált jellemzőpontszámok számtani átlaga alapján a jellemzők rangsorolása volt (ld. 1. tézis). A továbbiakban a kutatásom a jellemzőkiválasztást célozta meg, azzal a céllal, hogy küszöbértékeket határozzak meg a számtani átlagon alapuló többtényzős (ensemble) módszerrel kapott pontszámok számára. A cél az volt, hogy olyan releváns jellemzőkészletet határozzak meg, amely elegendő információt szolgáltat az osztályozó modul számára (ld. 2. tézis).

Ezt követően azzal a feltételezéssel folytattam a kutatást, hogy az egyes jellemzők pontszámainak súlyozott átlagon alapuló többtényezős módszer alkalmazása potenciálisan javíthatja az osztályozási teljesítményt, vagy legalábbis csökkentheti a szükséges jellemzők számát. Ezen hipotézis igazolása érdekében egy Taguchi típusú kísérlettervet alkalmaztam a szükséges kísérletek számának alacsony szinten tartása érdekében. A kísérleti eredmények megerősítették a hipotézist (ld. a 3. tézis).

Kutatásomat azzal a hipotézissel folytattam, hogy a korábban használt öt közismert osztályozó algoritmus által elért osztályozási teljesítményt a viszonylag új CatBoost algoritmus alkalmazásával felül lehet múlni, vagy legalábbis meg lehet közelíteni. A kísérleti eredmények ezt a hipotézist is megerősítették (ld. 4. tézis).

1. TÉZIS

Egy IDS rendszer tanítására használt adatbázis segítségével olyan módszert dolgoztam ki, ami alkalmas a beazonosításhoz szükséges jellemzők fontossági sorrendjének meghatározására az Információnyereség, Nyereségarány, Szimmetrikus bizonytalanság, Relief, Khi-négyzet próba és a Varianciaanalízis módszerek normalizált értékszámainak átlagai alapján végzett rangsorolás segítségével.

A tézisemhez tartozó publikációm a következő: [S3]

2.TÉZIS

Átlagolással kapott jellemző-értékszámokhoz kapcsolódóan meghatároztam azokat a küszöbértékeket, amelyek segítségével beazonosítható a jellemzők azon minimális darabszámú csoportja, amely mellett jó teljesítményű osztályozók taníthatók be.

A tézisemhez tartozó publikációm a következő: [S3]

3. TÉZIS

Megmutattam, hogy a különböző módszerekkel előállított jellemzőértékszámok súlyozásával képzett általános mérőszám alkalmazásával ugyanolyan vagy jobb osztályozási eredmények érhetőek el csökkentett számú figyelembe vett jellemző mellett.

A tézisemhez tartozó publikációm a következő: [S4]

4. TÉZIS

Megmutattam, hogy a CatBoost algoritmus alkalmazásával legalább olyan jó osztályozási eredmények érhetőek el bizonyos hálózati támadások esetében, mint a Logisztikus regresszió, Naive Bayes, Tartóvektor-gép, Döntési fa, Véletlen erdő osztályozótípusok használata mellett.

A tézisemhez tartozó publikációm a következő: [S5]

11. További kutatási irányok

Az IDS-rendszerekkel kapcsolatos kutatásomat az adatfúzión alapuló megoldások kialakítása irányában kívánom folytatni. Itt az adatfúzió olyan folyamatot jelent, amely során több forrásból vagy szenzorból származó adatokat kombinálnak és elemeznek annak érdekében, hogy növeljék a hálózati behatolások észlelésének pontosságát. Ide tartozik a tűzfalak, behatolásérzékelő-rendszerek (IDS), naplófájlok, hálózati forgalomadatok és más releváns források információinak integrálása. Az adatfúzió célja az, hogy kihasználja a különböző adatforrások előnyeit, és javítsa a behatolásérzékelő-rendszerek észlelési képességeit. Az egyes szenzorok vagy észlelési módszerek által esetleg fel nem ismert mintázatok és anomáliák valószínűleg könnyebben azonosíthatók több adatfolyam kombinálásával, ami megbízhatóbb behatolásérzékelést eredményezhet.

A kutatás előkészítéseként munkahelyemen, a kecskeméti Neumann János Egyetemen előkészítés alatt áll egy Informatikai Adatlabor kialakítása, melynek része lesz egy Szakértői és Kiberbiztonsági labor, ahol kialakításra kerül egy komplett informatikai laborkörnyezet, a kanadai laborhoz hasonlóan.

Egy teljes informatikai infrastruktúra lenne modellezve, normál és támadási kommunikációkkal. Egy ilyen rendszer kialakításával és az adatfúzió megvalósításával egy új, saját mintaadathalmazt tudnék létrehozni, amit az IDS-rendszereket kutatók szabadon használhatnának. Ezen adathalmaz felhasználásával a disszertációban bemutatott osztályozókon kívül más megoldások vizsgálatát is tervezem.

12. Summary

During the initial phase of my investigation, I delved into the fundamental characteristics and functionalities of various IDS systems [S15] [S11]. Subsequently, I directed my attention towards Anomaly-based IDS systems, specifically focusing on the training process of their classification module. Typically, this process involves utilizing big data samples that describe both benign and malicious traffic scenarios. Throughout my research, I explored multiple datasets, eventually discovering a suitable one (CSE-CIC-IDS2018 on AWS) that not only met the criteria but was also recent, making it ideal for training purposes.

Once the dataset was chosen and several preprocessing steps were executed, my investigation centered around feature selection methods. The primary outcome here was the ranking of features based on the arithmetic mean of normalized feature scores obtained from six distinct methods (refer to Thesis statement 1). Moving forward, my research targeted feature selection, aiming to establish threshold values for the scores obtained through the arithmetic mean based ensemble method. The objective was to identify a relevant set of features that would furnish sufficient information for the classifier module (refer to Thesis statement 2).

Subsequently, I pursued my investigation with the assumption that utilizing an ensemble method based on weighted aggregation of individual feature scores could potentially enhance classification performance or, at the very least, reduce the number of required features. To validate this hypothesis, I adopted a Taguchi-type DoE design, conducting a low number of trials. The experimental results confirmed the hypothesis (refer to Thesis statement 3).

Continuing my research, I operated under the hypothesis that the classification performance achieved by the five well-known classification algorithms used previously could be surpassed, or at least matched, by employing the relatively new CatBoost algorithm. The experimental results also confirmed this hypothesis (refer to Thesis statement 4).

Hivatkozott irodalom

[1] András K., ‘Hálózatbiztonság’, *TÁMOP-4.1.2-A/1-11/1-2011-0021*, 2013.

[2] Á. Bodlaki, A. Csernay, P. Mátyás, L. Muha, G. D. Papp, and D. Vadász, *Informatikai Tárcaközi Bizottság ajánlásai, Informatikai rendszerek biztonsági követelményei, 12. sz. ajánlás*. Budapest: Miniszterelnöki Hivatal Informatikai Koordinációs Iroda, 1996.

[3] C. Menyhárt, ‘Kockázatelemzés az informatikában’, Feb. 13, 2018. https://snt.hu/blog/kockazatelemzes-az-informatikaban/

[4] L. Göcs and Z. C. Johanyák, ‘SURVEY ON INTRUSION DETECTION SYSTEMS’, in *7th International Scientific and Expert Conference TEAM 2015 Technique, Education, Agriculture & Management*, 2015.

[5] L. Göcs, Z. C. Johanyák, and S. Kovács, ‘Review of Anomaly-Based IDS algorithms’, in *8th International Scientific and Expert Conference TEAM 2016 Technique, Education, Agriculture & Management.*, 2016.

[6] MTA SZTAKI, *Az informatikai hálózati infrastruktúra biztonsági kockázatai és kontrolljai*. Budapest, 2006.

[7] H.-J. Liao, C.-H. Richard Lin, Y.-C. Lin, and K.-Y. Tung, ‘Intrusion detection system: A comprehensive review’, *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 16–24, Jan. 2013, doi: 10.1016/j.jnca.2012.09.004.

[8] J. McHugh, A. Christie, and J. Allen, ‘Defending Yourself: The Role of Intrusion Detection Systems’, *IEEE Softw.*, vol. 17, no. 5, pp. 42–51, Sep. 2000, doi: 10.1109/52.877859.

[9] K. Dhangar, ‘A Proposed Intrusion Detection System’, *International Journal of Computer Applications*, vol. 65.

[10] D. Bolzoni and S. Etalle, ‘Approaches in anomaly-based intrusion detection systems’.

[11] C. Krügel, T. Toth, and E. Kirda, ‘Service specific anomaly detection for network intrusion detection’, in *Proceedings of the 2002 ACM symposium on Applied computing*, Madrid Spain: ACM, Mar. 2002, pp. 201–208. doi: 10.1145/508791.508835.

[12] K. Wang and S. J. Stolfo, ‘Anomalous Payload-Based Network Intrusion Detection’, in *Recent Advances in Intrusion Detection*, E. Jonsson, A. Valdes, and M. Almgren, Eds., in Lecture Notes in Computer Science, vol. 3224. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 203–222. doi: 10.1007/978-3-540-30143-1_11.

[13] V. Jyothsna, V. V. Rama Prasad, and K. Munivara Prasad, ‘A Review of Anomaly based Intrusion Detection Systems’, *IJCA*, vol. 28, no. 7, pp. 26–35, Aug. 2011, doi: 10.5120/3399-4730.

[14] A. Karami and M. Guerrero-Zapata, ‘A fuzzy anomaly detection system based on hybrid PSO-Kmeans algorithm in content-centric networks’, *Neurocomputing*, vol. 149, pp. 1253–1269, Feb. 2015, doi: 10.1016/j.neucom.2014.08.070.

[15] V. Jaiganesh, S. Mangayarkarasi, and D. P. Sumathi, ‘Intrusion Detection Systems: A Survey and Analysis of Classification Techniques’, vol. 2, no. 4, 2013.

[16] A. Pal Singh and M. Deep Singh, ‘Analysis of Host-Based and Network-Based Intrusion Detection System’, *IJCNIS*, vol. 6, no. 8, pp. 41–47, Jul. 2014, doi: 10.5815/ijcnis.2014.08.06.

[17] K. Dhangar, ‘A Proposed Intrusion Detection System’, *International Journal of Computer Applications*, vol. 65.

[18] Q. Ibrahim and S. Lazim, ‘Applying an Efficient Searching Algorithm for Intrusion Detection on Ubicom Network Processor’, vol. 2, no. 2, 2011.

[19] J. McHugh, A. Christie, and J. Allen, ‘Defending Yourself: The Role of Intrusion Detection Systems’, *IEEE Softw.*, vol. 17, no. 5, pp. 42–51, Sep. 2000, doi: 10.1109/52.877859.

[20] S. M. Othman, N. T. Alsohybe, F. M. Ba-Alwi, and A. T. Zahary, ‘Survey on Intrusion Detection System Types’, 2018.
[21] C.-M. Ou, ‘Host-based intrusion detection systems adapted from agent-based artificial immune systems’, *Neurocomputing*, vol. 88, pp. 78–86, Jul. 2012, doi: 10.1016/j.neucom.2011.07.031.
[22] C. Krügel, T. Toth, and E. Kirda, ‘Service specific anomaly detection for network intrusion detection’, in *Proceedings of the 2002 ACM symposium on Applied computing*, Madrid Spain: ACM, Mar. 2002, pp. 201–208. doi: 10.1145/508791.508835.
[23] A. H. Farooqi and F. A. Khan, ‘Intrusion Detection Systems for Wireless Sensor Networks: A Survey’, in *Communication and Networking*, D. Ślęzak, T. Kim, A. C.-C. Chang, T. Vasilakos, M. Li, and K. Sakurai, Eds., in Communications in Computer and Information Science, vol. 56. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 234–241. doi: 10.1007/978-3-642-10844-0_29.
[24] N. Einwechter, ‘An introduction to distributed intrusion detection systems’. Security Focus, 2001. [Online]. Available: https://tinyurl.hu/HXAd
[25] E. Biermann, E. Cloete, and L. M. Venter, ‘A comparison of Intrusion Detection systems’, *Computers & Security*, vol. 20, no. 8, pp. 676–683, Dec. 2001, doi: 10.1016/S0167-4048(01)00806-9.
[26] A. S. Ashoor and S. Gore, ‘Difference between Intrusion Detection System (IDS) and Intrusion Prevention System (IPS)’, in *Advances in Network Security and Applications*, D. C. Wyld, M. Wozniak, N. Chaki, N. Meghanathan, and D. Nagamalai, Eds., in Communications in Computer and Information Science, vol. 196. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 497–501. doi: 10.1007/978-3-642-22540-6_48.
[27] P. Joshi, C. Jindal, M. Chowkwale, R. Shethia, S. A. Shaikh, and D. Ved, ‘Protego: A passive intrusion detection system for Android smartphones’, in *2016 International Conference on Computing, Analytics and Security Trends (CAST)*, Pune, India: IEEE, Dec. 2016, pp. 232–237. doi: 10.1109/CAST.2016.7914972.
[28] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, ‘Anomaly-based network intrusion detection: Techniques, systems and challenges’, *Computers & Security*, vol. 28, no. 1–2, pp. 18–28, Feb. 2009, doi: 10.1016/j.cose.2008.08.003.
[29] L. Vokorokos, A. Baláž, and M. Chovanec, ‘INTRUSION DETECTION SYSTEM USING SELF ORGANIZING MAP’, vol. 6, no. 1, 2006.
[30] D. Kheyri and M. Karami, ‘A Comprehensive Survey on Anomaly-Based Intrusion Detection in MANET’, *CIS*, vol. 5, no. 4, p. p132, Jun. 2012, doi: 10.5539/cis.v5n4p132.
[31] V. Kumar and D. O. P. Sangwan, ‘Signature Based Intrusion Detection System Using SNORT’, *International Journal of Computer Applications*, vol. Vol. I, no. Issue III, pp. 35–41, Nov. 2012.
[32] C. Elkan, ‘Results of the KDD’99 classifier learning’, *SIGKDD Explor. Newsl.*, vol. 1, no. 2, pp. 63–64, Jan. 2000, doi: 10.1145/846183.846199.
[33] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, ‘Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization’:, in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, Funchal, Madeira, Portugal: SCITEPRESS - Science and Technology Publications, 2018, pp. 108–116. doi: 10.5220/0006639801080116.
[34] R. B. Basnet, R. Shash, C. Johnson, L. Walgren, and T. Doleck, ‘Towards Detecting and Classifying Network Intrusion Traffic Using Deep Learning Frameworks’, *J. Internet Serv. Inf. Secur.*, no. 9, pp. 1–17, 2019.
[35] R. H. = Lashkari, Canadian Institute for Cybersecurity (CIC), University of New Brunswick (UNB) Fredericton, Canada, M. Chen, Canadian Institute for Cybersecurity (CIC), University of New Brunswick (UNB) Fredericton, Canada, A. A. Ghorbani, and

Canadian Institute for Cybersecurity (CIC), University of New Brunswick (UNB) Fredericton, Canada, 'A Survey on User Profiling Model forAnomaly Detection in Cyberspace', *JCSM*, vol. 8, no. 1, pp. 75–112, 2018, doi: 10.13052/jcsm2245-1439.814.

[36] I. Sharafaldin *et al.*, 'Towards a Reliable Intrusion Detection Benchmark Dataset', *JSN*, vol. 2017, no. 1, pp. 177–200, 2017, doi: 10.13052/jsn2445-9739.2017.009.

[37] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, and A. Hotho, 'Flow-based benchmark data sets for intrusion detection', *ACPI*, vol. Proceedings of the 16th European Conference on Cyber Warfare and Security (ECCWS), pp. 361–369, 2017.

[38] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, and A. Hotho, 'Creation of Flow-Based Data Sets for Intrusion Detection', *Peregrine Technical Solutions*, vol. 4, no. 16, pp. 41–54, 2017.

[39] G. Maciá-Fernández, J. Camacho, R. Magán-Carrión, P. García-Teodoro, and R. Therón, 'UGR'16: A new dataset for the evaluation of cyclostationarity-based network IDSs', *Computers & Security*, vol. 73, pp. 411–424, Mar. 2018, doi: 10.1016/j.cose.2017.11.004.

[40] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, 'Towards Generating Real-life Datasets for Network Intrusion Detection', 2015.

[41] N. Moustafa and J. Slay, 'UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)', in *2015 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia: IEEE, Nov. 2015, pp. 1–6. doi: 10.1109/MilCIS.2015.7348942.

[42] G. Creech, 'Developing a high-accuracy cross platform Host-Based Intrusion Detection System capable of reliably detecting zero-day attacks', UNSW Sydney, 2014. doi: 10.26190/UNSWORKS/16615.

[43] G. Creech and J. Hu, 'Generation of a new IDS test dataset: Time to retire the KDD collection', in *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, Shanghai, Shanghai, China: IEEE, Apr. 2013, pp. 4487–4492. doi: 10.1109/WCNC.2013.6555301.

[44] M. Xie, J. Hu, and J. Slay, 'Evaluating Host-based Anomaly Detection Systems: Application of the One-class SVM Algorithm to'.

[45] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, 'Toward developing a systematic approach to generate benchmark datasets for intrusion detection', *Computers & Security*, vol. 31, no. 3, pp. 357–374, May 2012, doi: 10.1016/j.cose.2011.12.012.

[46] T. J. Lucas, C. A. C. Tojeiro, R. G. Pires, K. A. P. Da Costa, and J. P. Papa, 'Machine Learning for Web Intrusion Detection: A Comparative Analysis of Feature Selection Methods mRMR and PFI', in *Artificial Intelligence and Soft Computing*, L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz, and J. M. Zurada, Eds., in Lecture Notes in Computer Science, vol. 12415. Cham: Springer International Publishing, 2020, pp. 535–546. doi: 10.1007/978-3-030-61401-0_50.

[47] J. Song, H. Takakura, Y. Okabe, M. Eto, D. Inoue, and K. Nakao, 'Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation', in *Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security*, Salzburg Austria: ACM, Apr. 2011, pp. 29–36. doi: 10.1145/1978672.1978676.

[48] B. Sangster *et al.*, 'Toward Instrumenting Network Warfare Competitions to Generate Labeled Datasets'.

[49] J. O. Nehinbe, 'Critical analyses of alerts swamping and intrusion redundancy', in *2009 International Conference for Internet Technology and Secured Transactions, (ICITST)*, London: IEEE, Nov. 2009, pp. 1–8. doi: 10.1109/ICITST.2009.5402617.

[50] A. A. Olusola, A. S. Oladele, and D. O. Abosede, 'Analysis of KDD '99 Intrusion Detection Dataset for Selection of Relevance Features', 2010.

[51] A. Habibi Lashkari, G. Draper Gil, M. S. I. Mamun, and A. A. Ghorbani, 'Characterization of Tor Traffic using Time based Features':, in *Proceedings of the 3rd International Conference on Information Systems Security and Privacy*, Porto, Portugal: SCITEPRESS - Science and Technology Publications, 2017, pp. 253–262. doi: 10.5220/0006105602530262.
[52] J. Bernard, *Python Recipes Handbook: A Problem-Solution Approach*, 1st ed. 2016. Berkeley, CA: Apress : Imprint: Apress, 2016. doi: 10.1007/978-1-4842-0241-8.
[53] W. McKinney, 'Data Structures for Statistical Computing in Python', presented at the Python in Science Conference, Austin, Texas, 2010, pp. 56–61. doi: 10.25080/Majora-92bf1922-00a.
[54] D. Stiawan, M. Y. B. Idris, A. M. Bamhdi, R. Budiarto, and others, 'CICIDS-2017 dataset feature analysis with information gain for anomaly detection', *IEEE Access*, vol. 8, pp. 132911–132921, 2020, doi: 10.1109/ACCESS.2020.3009843.
[55] M. A. Rahman, A. T. Asyhari, O. W. Wen, H. Ajra, Y. Ahmed, and F. Anwar, 'Effective combining of feature selection techniques for machine learning-enabled IoT intrusion detection', *Multimedia Tools and Applications*, vol. 80, no. 20, pp. 31381–31399, 2021, doi: http://dx.doi.org/10.1007/s11042-021-10567-y.
[56] A. Javadpour, S. K. Abharian, and G. Wang, 'Feature selection and intrusion detection in cloud environment based on machine learning algorithms', in *2017 IEEE international symposium on parallel and distributed processing with applications and 2017 IEEE international conference on ubiquitous computing and communications (ISPA/IUCC)*, IEEE, 2017, pp. 1417–1421. doi: https://doi.org/10.1109/ISPA/IUCC.2017.00215.
[57] K. A. Taher, B. M. Y. Jisan, and M. M. Rahman, 'Network intrusion detection using supervised machine learning technique with feature selection', in *2019 International conference on robotics, electrical and signal processing techniques (ICREST)*, IEEE, 2019, pp. 643–646. doi: https://doi.org/10.1109/icrest.2019.8644161.
[58] G. Kocher and G. Kumar, 'Analysis of machine learning algorithms with feature selection for intrusion detection using UNSW-NB15 dataset', *Available at SSRN 3784406*, 2021, doi: https://doi.org/10.5121/ijnsa.2021.13102.
[59] M. Alkasassbeh, 'An empirical evaluation for the intrusion detection features based on machine learning and feature selection methods', *arXiv preprint arXiv:1712.09623*, 2017, doi: https://doi.org/10.48550/arXiv.1712.09623.
[60] I. S. Thaseen and C. A. Kumar, 'Intrusion detection model using fusion of chi-square feature selection and multi class SVM', *Journal of King Saud University-Computer and Information Sciences*, vol. 29, no. 4, pp. 462–472, 2017, doi: 10.1016/j.jksuci.2015.12.004.
[61] J. B. Awotunde, C. Chakraborty, and A. E. Adeniyi, 'Intrusion detection in industrial internet of things network-based on deep learning model with rule-based feature selection', *Wireless communications and mobile computing*, vol. 2021, 2021, doi: 10.1155/2021/7154587.
[62] H. P. S. Sasan and M. Sharma, 'Intrusion detection using feature selection and machine learning algorithm with misuse detection', *International Journal of Computer Science and Information Technology*, vol. 8, no. 1, pp. 17–25, 2016, doi: 10.5121/ijcsit.2016.8102.
[63] S. K. Biswas and others, 'Intrusion detection using machine learning: A comparison study', *International Journal of pure and applied mathematics*, vol. 118, no. 19, pp. 101–114, 2018.
[64] A. Ali *et al.*, 'Network intrusion detection leveraging machine learning and feature selection', in *2020 IEEE 17th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)*, IEEE, 2020, pp. 49–53. doi: 10.1109/honet50430.2020.9322813.

[65] H. Malhotra, P. Sharma, and others, ‘Intrusion detection using machine learning and feature selection’, *International Journal of Computer Network and Information security*, vol. 11, no. 4, p. 43, 2019, doi: 10.5815/ijcnis.2019.04.06.
[66] S. Krishnaveni, S. Sivamohan, S. Sridhar, and S. Prabakaran, ‘Efficient feature selection and classification through ensemble method for network intrusion detection on cloud computing’, *Cluster Computing*, vol. 24, no. 3, pp. 1761–1779, 2021, doi: https://doi.org/10.1007/s10586-020-03222-y.
[67] K. Kumar and J. S. Batth, ‘Network intrusion detection with feature selection techniques using machine-learning algorithms’, *International Journal of Computer Applications*, vol. 150, no. 12, 2016, doi: 10.5120/ijca2016910764.
[68] A. Pattawaro and C. Polprasert, ‘Anomaly-based network intrusion detection system through feature selection and hybrid machine learning technique’, in *2018 16th International Conference on ICT and Knowledge Engineering (ICT&KE)*, IEEE, 2018, pp. 1–6. doi: 10.1109/ictke.2018.8612331.
[69] T. Ahmad and M. N. Aziz, ‘Data preprocessing and feature selection for machine learning intrusion detection systems’, *ICIC Express Lett*, vol. 13, no. 2, pp. 93–101, 2019, doi: 10.24507/icicel.13.02.93.
[70] M. Manonmani and S. Balakrishnan, ‘An Ensemble Feature Selection Method for Prediction of CKD’, in *2020 International Conference on Computer Communication and Informatics (ICCCI)*, Coimbatore, India: IEEE, Jan. 2020, pp. 1–6. doi: 10.1109/ICCCI48352.2020.9104137.
[71] A. Hashemi, M. B. Dowlatshahi, and H. Nezamabadi-pour, ‘Ensemble of feature selection algorithms: a multi-criteria decision-making approach’, *Int. J. Mach. Learn. & Cyber.*, vol. 13, no. 1, pp. 49–69, Jan. 2022, doi: 10.1007/s13042-021-01347-z.
[72] N. Hoque, M. Singh, and D. K. Bhattacharyya, ‘EFS-MI: an ensemble feature selection method for classification: An ensemble feature selection method’, *Complex Intell. Syst.*, vol. 4, no. 2, pp. 105–118, Jun. 2018, doi: 10.1007/s40747-017-0060-x.
[73] A. S. Sumant and D. Patil, ‘Ensemble Feature Subset Selection: Integration of Symmetric Uncertainty and Chi-Square techniques with RReliefF’, *J. Inst. Eng. India Ser. B*, vol. 103, no. 3, pp. 831–844, Jun. 2022, doi: 10.1007/s40031-021-00684-5.
[74] C.-F. Tsai and Y.-T. Sung, ‘Ensemble feature selection in high dimension, low sample size datasets: Parallel and serial combination approaches’, *Knowledge-Based Systems*, vol. 203, p. 106097, Sep. 2020, doi: 10.1016/j.knosys.2020.106097.
[75] J. Wang, J. Xu, C. Zhao, Y. Peng, and H. Wang, ‘An ensemble feature selection method for high-dimensional data based on sort aggregation’, *Systems Science & Control Engineering*, vol. 7, no. 2, pp. 32–39, Nov. 2019, doi: 10.1080/21642583.2019.1620658.
[76] Z. J. Viharos, K. B. Kis, Á. Fodor, and M. I. Büki, ‘Adaptive, hybrid feature selection (AHFS)’, *Pattern Recognition*, vol. 116, p. 107932, 2021, doi: 10.1016/j.patcog.2021.107932.
[77] K. Muhi and Z. C. Johanyák, ‘Dimensionality reduction methods used in Machine Learning’, *M\Huszaki Tudományos Közlemények*, vol. 13, no. 1, pp. 148–151, 2020, doi: 10.33894/mtk-2020.13.27.
[78] T. Dobján and E. D. Antal, ‘Modern feature extraction methods and learning algorithms in the field of industrial acoustic signal processing’, in *2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY)*, IEEE, 2017, pp. 000065–000070. doi: 10.1109/sisy.2017.8080589.
[79] N. S. Chauhan, ‘Decision Tree Algorithm—Explained’. 2020. [Online]. Available: https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html/https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html/

[80] V. Bolón-Canedo and A. Alonso-Betanzos, ‘Ensembles for feature selection: A review and future trends’, *Information Fusion*, vol. 52, pp. 1–12, Dec. 2019, doi: 10.1016/j.inffus.2018.11.008.
[81] G. Ayyappan, D. C. Nalini, and D. A. Kumaravel, ‘Efficient mining for social networks using Information Gain Ratio based on Academic dataset’, *International Journal of Civil Engineering and Technology*, vol. 8, no. 1, 2017.
[82] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, *Feature selection for high-dimensional data*. Springer, 2015. doi: 10.1007/978-3-319-21858-8.
[83] A. G. Karegowda, A. S. Manjunath, and M. A. Jayaram, ‘COMPARATIVE STUDY OF ATTRIBUTE SELECTION USING GAIN RATIO AND CORRELATION BASED FEATURE SELECTION’.
[84] R. P. Priyadarsini, M. Valarmathi, and S. Sivakumari, ‘Gain ratio based feature selection method for privacy preservation’, *ICTACT Journal on soft computing*, vol. 1, no. 4, pp. 201–205, 2011, doi: 10.21917/ijsc.2011.0031.
[85] S. J. Pasha and E. S. Mohamed, ‘Ensemble Gain Ratio Feature Selection (EGFS) Model with Machine Learning and Data Mining Algorithms for Disease Risk Prediction’, in *2020 International Conference on Inventive Computation Technologies (ICICT)*, Coimbatore, India: IEEE, Feb. 2020, pp. 590–596. doi: 10.1109/ICICT48043.2020.9112406.
[86] R. J. Urbanowicz, M. Meeker, W. La Cava, R. S. Olson, and J. H. Moore, ‘Relief-based feature selection: Introduction and review’, *Journal of biomedical informatics*, vol. 85, pp. 189–203, 2018, doi: 10.1016/j.jbi.2018.07.014.
[87] B. Singh, N. Kushwaha, O. P. Vyas, and others, ‘A feature subset selection technique for high dimensional data using symmetric uncertainty’, *Journal of Data Analysis and Information Processing*, vol. 2, no. 04, p. 95, 2014, doi: 10.4236/jdaip.2014.24012.
[88] S. Bakhshandeh, R. Azmi, and M. Teshnehlab, ‘Symmetric uncertainty class-feature association map for feature selection in microarray dataset’, *Int. J. Mach. Learn. & Cyber.*, vol. 11, no. 1, pp. 15–32, Jan. 2020, doi: 10.1007/s13042-019-00932-7.
[89] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, *Feature selection for high-dimensional data*. Springer, 2015.
[90] M. Kumar, N. K. Rath, A. Swain, and S. K. Rath, ‘Feature selection and classification of microarray data using MapReduce based ANOVA and K-nearest neighbor’, *Procedia Computer Science*, vol. 54, pp. 301–310, 2015, doi: 10.1016/j.procs.2015.06.035.
[91] M. Héder *et al.*, ‘The Past, Present and Future of the ELKH Cloud’, *InfTars*, vol. 22, no. 2, p. 128, Aug. 2022, doi: 10.22503/inftars.XXII.2022.2.8.
[92] M. Maalouf, ‘Logistic regression in data analysis: an overview’, *IJDATS*, vol. 3, no. 3, p. 281, 2011, doi: 10.1504/IJDATS.2011.041335.
[93] I. Steinwart and A. Christmann, *Support vector machines*, 1st ed. in Information science and statistics. New York: Springer, 2008.
[94] B. Charbuty and A. Abdulazeez, ‘Classification based on decision tree algorithm for machine learning’, *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, 2021, doi: 10.38094/jastt20165.
[95] R. Davidson, ‘Reliable inference for the Gini index’, *Journal of Econometrics*, vol. 150, no. 1, pp. 30–40, May 2009, doi: 10.1016/j.jeconom.2008.11.004.
[96] L. Breiman, ‘Random Forests’, *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
[97] A. Freddi and M. Salmon, *Design Principles and Methodologies: From Conceptualization to First Prototyping with Examples and Case Studies*, 1st ed. 2019. in Springer Tracts in Mechanical Engineering. Cham: Springer International Publishing : Imprint: Springer, 2019. doi: 10.1007/978-3-319-95342-7.

[98] S. R, A. Raj, K. Saivenu, M. I. Ahmed, S. B, and A. Kanavalli, ‘Detection and mitigation of botnet based DDoS attacks using catboost machine learning algorithm in SDN environment’, *IJATEE*, vol. 8, no. 76, pp. 445–461, Mar. 2021, doi: 10.19101/IJATEE.2021.874021.
[99] D. Chepenko, ‘Introduction to gradient boosting on decision trees with Catboost’, *Towards Data Science*, Feb. 13, 2019. https://towardsdatascience.com/introduction-to-gradient-boosting-on-decision-trees-with-catboost-d511a9ccbd14
[100] F. Zhou *et al.*, ‘Fire Prediction Based on CatBoost Algorithm’, *Mathematical Problems in Engineering*, vol. 2021, pp. 1–9, Jul. 2021, doi: 10.1155/2021/1929137.

Saját publikációk

Folyóiratcikkek

S1. **László, Göcs**; Attila, Pásztor; Zsolt, Csaba Johanyák: Computer network solutions in modern industrial environment ANNALS OF FACULTY OF ENGINEERING HUNEDOARA - INTERNATIONAL JOURNAL OF ENGINEERING 10: 1 pp. 75-80., 6 p. (2022)
S2. **László, Göcs**; Zsolt, Csaba Johanyák; Péter, András Agg: Protection of Computer Laboratories in Educational Institutions ACTA TECHNICA CORVINIENSIS – BULLETIN OF ENGINEERING 9: 2 pp. 93-98., 6 p. (2016)
S3. **László, Göcs**; Zsolt, Csaba Johanyák: Identifying Relevant Features of CSE-CIC-IDS2018 Dataset for the Development of an Intrusion Detection System, 2023 http://gocslaszlo.hu/phd/tezis_1_2.pdf (publikálás alatt)
S4. **L. Göcs** and Z. C. Johanyák, ‘Feature Selection with Weighted Ensemble Ranking for Improved Classification Performance on the CSE-CIC-IDS2018 Dataset’, Computers, vol. 12, no. 8, p. 147, Jul. 2023, doi: 10.3390/computers12080147.
S5. **László, Göcs**; Zsolt, Csaba Johanyák: Catboost algorithm based IDS cassification module for brute force attacks", ANNALS OF FACULTY OF ENGINEERING HUNEDOARA: INTERNATIONAL JOURNAL OF ENGINEERING (2023) http://gocslaszlo.hu/phd/tezis_4.pdf (megjelenés alatt)

Konferenciaközlemények

S6. **Göcs, László**; Johanyák, Zsolt Csaba: Adatbázis feldolgozása IDS rendszerek tanításához Kutatás és innováció 2021: GAMF Közlemények tanulmánykötete Kecskemét, Magyarország: Neumann János Egyetem GAMF Műszaki és Informatikai Kar (2021) pp. 401-406., 6 p.
S7. **Göcs, László**; Pásztor, Attila; Johanyák, Zsolt Csaba: Modern ipari környezet informatikai hálózati lehetőségei a rendelkezésre állás biztosítása érdekében GRADUS 8: 3 pp. 147-156., 10 p. (2021)
S8. **Göcs, László**; Johanyák, Csaba; Kovács, Szilveszter: IDS rendszerek fuzzy logikával In: Keresztes, Gábor; Kohus, Zsolt; Szabó P., Katalin; Tokody, Dániel (szerk.) Tavaszi Szél 2017 Konferencia. Nemzetközi Multidiszciplináris Konferencia: Absztraktkötet Budapest, Magyarország: Doktoranduszok Országos Szövetsége (DOSZ) (2017) 477 p. p. 311
S9. Agg, Péter András; Johanyák, Zsolt Csaba; **Göcs, László**: Szoftver által definiált hálózatok áttekintése In: Bitay, Enikő (szerk.) A XXI. Fiatal Műszakiak Tudományos Ülésszaka előadásai Kolozsvár, Románia: Erdélyi Múzeum Egyesület (EME) (2016) 452 p. pp. 57-60., 4 p.

S10. **Göcs, László**; Johanyák, Zsolt Csaba; Kovács, Szilveszter: Csapda a hálózaton GRADUS 3: 2 pp. 55-60., 6 p. (2016)
S11. **László, Göcs**; Zsolt, Csaba Johanyák; Szilveszter, Kovács: Review of Anomaly-Based IDS algorithms In: AlumniPress - AlumniPress (szerk.) TEAM 2016: Proceedings of the 8th International Scientific and Expert Conference Trnava, Szlovákia: Alumni Press (2016) 360 p. pp. 58-63., 6 p.
S12. **László, Göcs**; Zsolt, Csaba Johanyák: Virtualization in Network Administration Education In: Kucsinka, Katalin; Kiss, Alexandra; Veres, Erika (szerk.) Matematikát oktatók és kutatók nemzetközi tudományos konferenciája Beregszász, Ukrajna: II. Rákóczi Ferenc Kárpátaljai Magyar Főiskola (2016) 78 p. p. 52
S13. Agg, P; **Göcs, L**; Johanyák, Zs Cs; Borza, Z: Csomagszűrés CISCO routereken ACL-ek segítségével GRADUS 2: 2 pp. 104-111., 8 p. (2015)
S14. **Göcs, László**; Johanyák, Zsolt Csaba: Vállalati informatikai biztonság szerepe napjainkban In: Bitay, Enikő (szerk.) A XX. Fiatal Műszakiak Tudományos Ülésszaka előadásai: Proceedings of the XX-th International Sciencific Conference of Young Engineers Kolozsvár, Románia: Erdélyi Múzeum Egyesület (EME) (2015) 356 p. pp. 155-158., 4 p.
S15. **László, Göcs**; Zsolt, Csaba Johanyák: Survey on intrusion detection systems In: Prof, Aleksandar Sedmak; Zoran, Radakovic; Simon, Sedmak; Snezana, Kirin (szerk.) Proceedings of TEAM 2015: 7th International Scientific and Expert Conference of the International TEAM Society Beograd, Szerbia: University of Belgrade, Faculty of Mechanical Engineering (2015) 650 p. pp. 167-170., 4 p.
S16. Zsolt, Csaba Johanyák; Piroska, Gyöngyi Ailer; **László, Göcs**: A simple fuzzy control design for series hybrid electric vehicle In: Andrea, Ádámné Major; Lóránt, Kovács; Zsolt, Csaba Johanyák; Róbert, Pap-Szigeti (szerk.) Proceedings of TEAM 2014: 6th International Scientific and Expert Conference of the International TEAM Society Kecskemét, Magyarország: Kecskeméti Főiskola Gépipari és Automatizálási Műszaki Főiskolai Kar (2014) 499 p. pp. 159-164., 6 p.
S17. **Göcs, László**: Informatikai biztonság In: Ferencz, Árpád; Borsné, Pető Judit; Lipócziné, Csabai Sarolta; Kovács, Lóránt (szerk.) AGTEDU 2011: a Magyar Tudomány Ünnepe alkalmából rendezett 12. tudományos konferencia Kecskemét, Magyarország: Kecskeméti Főiskola (2011) 406 p. pp. 143-148., 6 p.

Egyéb publikációk

S18. **Göcs, László**: Covid19 hatása az informatikai rendszerekre (2020) AGTEDU 2020, 2020. november 12., Előadás,
S19. **Göcs, László**: Adataink és az okos eszközök - kémek a lakásban? (2019) AGTEDU 2019, 2019. november 13., Előadás,
S20. **Göcs, László**; Johanyák, Zsolt Csaba: Címkézett adatbázis IDS rendszerekhez (2018) AGTEDU 2018, Kecskemét: 2018. november 15., Előadás,
S21. **Göcs, László**: A digitális világ biztonságos használata: Internet és informatikai biztonság (2018) Hírös Szabadegyetem, Kecskemét, 2018. április 11., Előadás,
S22. **László, Göcs**: Importance of passwords in IT security (2018) WCNCI 2018 (Workshop on Computer Networks and Computational Intelligence), 2018. október 16., Előadás,
S23. **Göcs, László**; Johanyák, Zsolt Csaba; Bors, Ádám: A blokklánc technológia (2017) AGTEDU 2017: Magyar Tudomány Ünnepe: 2017. november 16., Neumann János Egyetem Gazdálkodási Kar, Szolnok, Előadás, Megjelenés: Magyarország,

Oktatási anyagok

S24. **Göcs, László**: Szerveroldali megoldások Linux környezetben (Ubuntu 20.04 LTS) Kecskemét, Magyarország: Neumann János Egyetem (2021) ISBN: 9786155817915

S25. Johanyák, Zsolt Csaba; **Göcs, László**: Windows hálózati adminisztráció a gyakorlatban, 153 p. (2014)

S26. Johanyák, Zsolt Csaba; Kovács, Péter; **Göcs, László**: Linux hálózati adminisztráció a gyakorlatban, 113 p. (2013)

Mellékletek

A.1. táblázat: Jellemzőkiválasztási eredmények az FTP adathalmazhoz

Features	IG	SU	GR	Chi²	Relief	ANOVA	Átlag
00 Dst Port	1.0000	0.4848	0.3632	0.0339	0.2564	0.0000	0.3600
01 Protocol	0.1426	0.2024	0.2261	0.1002	0.3465	0.0000	0.1700
02 Flow Duration	0.8921	0.1560	0.0756	0.0000	0.0000	1.0000	0.3500
03 Tot Fwd Pkts	0.4072	0.2647	0.1959	0.0001	0.0010	0.0195	0.1500
04 Tot Bwd Pkts	0.3648	0.2420	0.1772	0.0002	0.0005	0.0008	0.1300
05 TotLen Fwd Pkts	0.4562	0.1787	0.1036	0.0000	0.0001	1.0000	0.2900
06 TotLen Bwd Pkts	0.3824	0.1490	0.0798	0.0002	0.0004	0.0013	0.1000
07 Fwd Pkt Len Max	0.4562	0.1985	0.1223	0.0000	0.0040	1.0000	0.3000
08 Fwd Pkt Len Min	0.1358	0.1123	0.0694	0.0009	0.0205	0.0000	0.0600
09 Fwd Pkt Len Mean	0.4562	0.1724	0.0979	0.0000	0.0075	0.1426	0.1500
10 Fwd Pkt Len Std	0.2041	0.1056	0.0497	0.0000	0.0046	0.1909	0.0900
11 Bwd Pkt Len Max	0.3824	0.1808	0.1102	0.1347	0.3325	0.0000	0.1900
12 Bwd Pkt Len Min	0.1284	0.0929	0.0464	0.0579	0.0495	0.0000	0.0600
13 Bwd Pkt Len Mean	0.3824	0.1484	0.0792	0.0855	0.1292	0.0000	0.1400
14 Bwd Pkt Len Std	0.1684	0.1024	0.0505	0.1089	0.2837	0.0000	0.1200
15 Flow Byts/s	0.4575	0.0933	0.0298	0.0004	0.0002	0.0000	0.1000
16 Flow Pkts/s	0.8994	0.1558	0.0753	0.8117	0.0166	0.0000	0.3300
17 Flow IAT Mean	0.8974	0.1560	0.0755	0.0000	0.0000	1.0000	0.3500
18 Flow IAT Std	0.2700	0.0724	0.0156	0.0084	0.0000	0.0000	0.0600
19 Flow IAT Max	0.8862	0.1602	0.0789	0.0000	0.0000	1.0000	0.3500
20 Flow IAT Min	0.5574	0.1429	0.0691	0.0000	0.0000	1.0000	0.2900
21 Fwd IAT Tot	0.4072	0.0944	0.0314	0.0000	0.0000	1.0000	0.2600
22 Fwd IAT Mean	0.4072	0.0943	0.0313	0.0000	0.0000	1.0000	0.2600
23 Fwd IAT Std	0.2211	0.0691	0.0141	0.0085	0.0000	0.0000	0.0500
24 Fwd IAT Max	0.4072	0.0954	0.0322	0.0000	0.0000	1.0000	0.2600
25 Fwd IAT Min	0.3974	0.1242	0.0569	0.0000	0.0000	1.0000	0.2600
26 Bwd IAT Tot	0.2029	0.0673	0.0131	0.0225	0.1070	0.0000	0.0700
27 Bwd IAT Mean	0.2029	0.0673	0.0130	0.0008	0.0106	0.0000	0.0500
28 Bwd IAT Std	0.1698	0.0606	0.0083	0.0057	0.0230	0.0000	0.0400
29 Bwd IAT Max	0.2029	0.0707	0.0161	0.0073	0.0534	0.0000	0.0600
30 Bwd IAT Min	0.1741	0.0764	0.0230	0.0004	0.0010	0.0000	0.0500
31 Fwd PSH Flags	0.0133	0.0316	0.0824	0.0092	0.0652	0.0000	0.0300
32 Fwd Header Len	0.5691	0.3017	0.2158	0.0001	0.0008	0.0286	0.1900
33 Bwd Header Len	0.7297	0.3911	0.2920	0.0002	0.0008	0.0008	0.2400
34 Fwd Pkts/s	0.9286	0.1606	0.0789	0.7342	0.0071	0.0000	0.3200
35 Bwd Pkts/s	0.9534	0.1905	0.1025	0.8485	0.0165	0.0000	0.3500
36 Pkt Len Min	0.1336	0.1121	0.0695	0.0010	0.0236	0.0000	0.0600
37 Pkt Len Max	0.4575	0.2005	0.1242	0.0000	0.0111	1.0000	0.3000
38 Pkt Len Mean	0.4575	0.1496	0.0774	0.0148	0.0426	0.0000	0.1200
39 Pkt Len Std	0.4434	0.1558	0.0834	0.0071	0.0236	0.0000	0.1200
40 Pkt Len Var	0.4434	0.1558	0.0834	0.0000	0.0012	1.0000	0.2800
41 FIN Flag Cnt	0.0000	0.0000	0.0403	0.0011	0.0000	0.0000	0.0100
42 SYN Flag Cnt	0.0133	0.0316	0.0824	0.0092	0.0652	0.0000	0.0300
43 RST Flag Cnt	0.0229	0.0495	0.0991	0.0150	0.0870	0.0000	0.0500
44 PSH Flag Cnt	0.3175	0.4268	0.4923	0.2550	1.0000	0.0000	0.4200
45 ACK Flag Cnt	0.1133	0.1758	0.2147	0.0771	0.4348	0.0000	0.1700
46 URG Flag Cnt	0.0177	0.0398	0.0902	0.0118	0.1522	0.0000	0.0500
47 ECE Flag Cnt	0.0229	0.0495	0.0991	0.0150	0.0870	0.0000	0.0500
48 Down/Up Ratio	0.2373	0.3113	0.3429	0.0001	0.0109	0.0206	0.1500
49 Pkt Size Avg	0.4575	0.1508	0.0784	0.0160	0.0679	0.0000	0.1300
50 Fwd Seg Size Avg	0.4562	0.1724	0.0979	0.0000	0.0071	0.1426	0.1500
51 Bwd Seg Size Avg	0.3824	0.1484	0.0792	0.0855	0.1263	0.0000	0.1400
52 Subflow Fwd Pkts	0.4072	0.2647	0.1959	0.0001	0.0015	0.0195	0.1500
53 Subflow Fwd Byts	0.4562	0.1787	0.1036	0.0000	0.0001	1.0000	0.2900
54 Subflow Bwd Pkts	0.3648	0.2420	0.1772	0.0002	0.0038	0.0008	0.1300
55 Subflow Bwd Byts	0.3824	0.1490	0.0798	0.0002	0.0001	0.0013	0.1000
56 Init Fwd Win Byts	0.9947	0.5830	0.4647	0.9827	0.7044	0.0000	0.6200
57 Init Bwd Win Byts	0.8070	0.4417	0.3366	0.0465	0.2011	0.0000	0.3100
58 Fwd Act Data Pkts	0.2357	0.1962	0.1499	0.0000	0.0051	0.7391	0.2200
59 Fwd Seg Size Min	0.9932	1.0000	1.0000	1.0000	0.9149	0.0000	0.8200
60 Active Mean	0.0419	0.0382	0.0007	0.0002	0.0008	0.0001	0.0100
61 Active Std	0.0298	0.0333	0.0000	0.0003	0.0005	0.0000	0.0100
62 Active Max	0.0419	0.0383	0.0007	0.0003	0.0031	0.0000	0.0100
63 Active Min	0.0419	0.0393	0.0024	0.0002	0.0017	0.0001	0.0100
64 Idle Mean	0.0487	0.0403	0.0010	0.0117	0.0000	0.0000	0.0200
65 Idle Std	0.0364	0.0363	0.0011	0.0009	0.0000	0.0000	0.0100
66 Idle Max	0.0487	0.0408	0.0017	0.0119	0.0000	0.0000	0.0200
67 Idle Min	0.0487	0.0406	0.0014	0.0110	0.0001	0.0000	0.0200

A.2. táblázat: Jellemzőkiválasztási eredmények az SSH adathalmazhoz

Features	IG	SU	GR	Chi²	Relief	ANOVA	Átlag
00 Dst Port	*0.9966*	*0.4882*	*0.3850*	*0.0334*	*0.2960*	*0.0000*	*0.3700*
01 Protocol	*0.1431*	*0.2049*	*0.2512*	*0.1003*	*0.5702*	*0.0000*	*0.2100*
02 Flow Duration	*0.9799*	*0.1532*	*0.1013*	*0.0000*	*0.0000*	*1.0000*	*0.3700*
03 Tot Fwd Pkts	*0.5243*	*0.3012*	*0.2430*	*0.0001*	*0.0010*	*0.0116*	*0.1800*
04 Tot Bwd Pkts	*0.5290*	*0.3128*	*0.2543*	*0.0002*	*0.0003*	*0.0010*	*0.1800*
05 TotLen Fwd Pkts	*0.6226*	*0.2229*	*0.1631*	*0.0000*	*0.0001*	*0.7893*	*0.3000*
06 TotLen Bwd Pkts	*0.5797*	*0.2151*	*0.1579*	*0.0002*	*0.0001*	*0.0016*	*0.1600*
07 Fwd Pkt Len Max	*0.6233*	*0.2567*	*0.1929*	*0.0000*	*0.0025*	*0.7893*	*0.3100*
08 Fwd Pkt Len Min	*0.1351*	*0.1128*	*0.0981*	*0.0004*	*0.0278*	*0.0000*	*0.0600*
09 Fwd Pkt Len Mean	*0.6218*	*0.2094*	*0.1515*	*0.0000*	*0.0064*	*0.0786*	*0.1800*
10 Fwd Pkt Len Std	*0.4795*	*0.2126*	*0.1606*	*0.0000*	*0.0026*	*0.7893*	*0.2700*
11 Bwd Pkt Len Max	*0.5797*	*0.2580*	*0.1965*	*0.4811*	*0.3560*	*0.0000*	*0.3100*
12 Bwd Pkt Len Min	*0.1298*	*0.0931*	*0.0756*	*0.0788*	*0.1573*	*0.0000*	*0.0900*
13 Bwd Pkt Len Mean	*0.5798*	*0.2099*	*0.1534*	*0.0848*	*0.0733*	*0.0000*	*0.1800*
14 Bwd Pkt Len Std	*0.4598*	*0.2500*	*0.1980*	*0.3311*	*0.1181*	*0.0000*	*0.2300*
15 Flow Byts/s	*0.6236*	*0.1094*	*0.0706*	*0.0010*	*0.0006*	*0.0000*	*0.1300*
16 Flow Pkts/s	*0.9839*	*0.1517*	*0.1001*	*0.2412*	*0.0025*	*0.0000*	*0.2500*
17 Flow IAT Mean	*0.9819*	*0.1524*	*0.1007*	*0.0000*	*0.0000*	*1.0000*	*0.3700*
18 Flow IAT Std	*0.5156*	*0.1143*	*0.0755*	*0.0000*	*0.0000*	*0.5998*	*0.2200*
19 Flow IAT Max	*0.9789*	*0.1549*	*0.1027*	*0.0000*	*0.0000*	*1.0000*	*0.3700*
20 Flow IAT Min	*0.7438*	*0.1877*	*0.1306*	*0.0000*	*0.0000*	*0.5998*	*0.2800*
21 Fwd IAT Tot	*0.5926*	*0.1153*	*0.0754*	*0.0000*	*0.0000*	*1.0000*	*0.3000*
22 Fwd IAT Mean	*0.5927*	*0.1152*	*0.0753*	*0.0000*	*0.0000*	*1.0000*	*0.3000*
23 Fwd IAT Std	*0.4883*	*0.1235*	*0.0832*	*0.0000*	*0.0000*	*0.5998*	*0.2200*
24 Fwd IAT Max	*0.5927*	*0.1135*	*0.0742*	*0.0000*	*0.0000*	*1.0000*	*0.3000*
25 Fwd IAT Min	*0.3049*	*0.0812*	*0.0524*	*0.0000*	*0.0000*	*0.5998*	*0.1700*
26 Bwd IAT Tot	*0.4795*	*0.1265*	*0.0859*	*0.0229*	*0.2796*	*0.0000*	*0.1700*
27 Bwd IAT Mean	*0.4795*	*0.1263*	*0.0857*	*0.0008*	*0.0363*	*0.0000*	*0.1200*
28 Bwd IAT Std	*0.4608*	*0.1298*	*0.0890*	*0.0059*	*0.0422*	*0.0000*	*0.1200*
29 Bwd IAT Max	*0.4795*	*0.1292*	*0.0882*	*0.0076*	*0.0605*	*0.0000*	*0.1300*
30 Bwd IAT Min	*0.4040*	*0.1598*	*0.1164*	*0.0004*	*0.0013*	*0.0000*	*0.1100*
31 Fwd PSH Flags	*0.0131*	*0.0316*	*0.1101*	*0.0091*	*0.0782*	*0.0000*	*0.0400*
32 Fwd Header Len	*0.9806*	*0.4520*	*0.3519*	*0.0001*	*0.0008*	*0.0654*	*0.3100*
33 Bwd Header Len	*0.9023*	*0.4388*	*0.3449*	*0.0002*	*0.0070*	*0.0010*	*0.2800*

Features	IG	SU	GR	Chi²	Relief	ANOVA	Átlag
34 Fwd Pkts/s	*0.9767*	*0.1508*	*0.0995*	*0.0032*	*0.0101*	*0.0000*	*0.2100*
35 Bwd Pkts/s	*0.9841*	*0.1730*	*0.1163*	*0.0494*	*0.0045*	*0.0000*	*0.2200*
36 Pkt Len Min	*0.1350*	*0.1127*	*0.0982*	*0.0009*	*0.0550*	*0.0000*	*0.0700*
37 Pkt Len Max	*0.6180*	*0.2568*	*0.1933*	*0.0000*	*0.0077*	*0.7893*	*0.3100*
38 Pkt Len Mean	*0.6183*	*0.1827*	*0.1291*	*0.0144*	*0.0649*	*0.0000*	*0.1700*
39 Pkt Len Std	*0.6102*	*0.1923*	*0.1373*	*0.0046*	*0.0166*	*0.0000*	*0.1600*
40 Pkt Len Var	*0.6102*	*0.1923*	*0.1373*	*0.0000*	*0.0008*	*0.7893*	*0.2900*
41 FIN Flag Cnt	*0.0000*	*0.0000*	*0.0701*	*0.0011*	*0.0391*	*0.0000*	*0.0200*
42 SYN Flag Cnt	*0.0131*	*0.0316*	*0.1101*	*0.0091*	*0.1173*	*0.0000*	*0.0500*
43 RST Flag Cnt	*0.0204*	*0.0490*	*0.1257*	*0.0146*	*0.1954*	*0.0000*	*0.0700*
44 PSH Flag Cnt	*0.0054*	*0.0046*	*0.0000*	*0.0080*	*0.0000*	*0.0000*	*0.0000*
45 ACK Flag Cnt	*0.0349*	*0.0487*	*0.0556*	*0.0420*	*0.0000*	*0.0000*	*0.0300*
46 URG Flag Cnt	*0.2112*	*0.3713*	*0.5644*	*0.2753*	*0.0000*	*0.0000*	*0.2400*
47 ECE Flag Cnt	*0.0204*	*0.0490*	*0.1257*	*0.0146*	*0.1564*	*0.0000*	*0.0600*
48 Down/Up Ratio	*0.0439*	*0.0546*	*0.0578*	*0.0001*	*0.0005*	*0.0485*	*0.0300*
49 Pkt Size Avg	*0.6138*	*0.1826*	*0.1291*	*0.0155*	*0.0543*	*0.0000*	*0.1700*
50 Fwd Seg Size Avg	*0.6218*	*0.2094*	*0.1515*	*0.0000*	*0.0078*	*0.0786*	*0.1800*
51 Bwd Seg Size Avg	*0.5798*	*0.2099*	*0.1534*	*0.0848*	*0.0985*	*0.0000*	*0.1900*
52 Subflow Fwd Pkts	*0.5243*	*0.3012*	*0.2430*	*0.0001*	*0.0010*	*0.0116*	*0.1800*
53 Subflow Fwd Byts	*0.6226*	*0.2229*	*0.1631*	*0.0000*	*0.0001*	*0.7893*	*0.3000*
54 Subflow Bwd Pkts	*0.5290*	*0.3128*	*0.2543*	*0.0002*	*0.0004*	*0.0010*	*0.1800*
55 Subflow Bwd Byts	*0.5797*	*0.2151*	*0.1579*	*0.0002*	*0.0016*	*0.0016*	*0.1600*
56 Init Fwd Win Byts	*0.9964*	*0.5596*	*0.4550*	*0.4722*	*0.2168*	*0.0000*	*0.4500*
57 Init Bwd Win Byts	*1.0000*	*0.5338*	*0.4290*	*0.0457*	*0.3257*	*0.0000*	*0.3900*
58 Fwd Act Data Pkts	*0.4757*	*0.3512*	*0.3047*	*0.0000*	*0.0020*	*0.6899*	*0.3000*
59 Fwd Seg Size Min	*0.9734*	*1.0000*	*1.0000*	*1.0000*	*1.0000*	*0.0000*	*0.8300*
60 Active Mean	*0.0422*	*0.0383*	*0.0312*	*0.0008*	*0.0010*	*0.0000*	*0.0200*
61 Active Std	*0.0303*	*0.0336*	*0.0305*	*0.0006*	*0.0015*	*0.0000*	*0.0200*
62 Active Max	*0.0422*	*0.0384*	*0.0312*	*0.0014*	*0.0041*	*0.0000*	*0.0200*
63 Active Min	*0.0422*	*0.0395*	*0.0328*	*0.0007*	*0.0027*	*0.0000*	*0.0200*
64 Idle Mean	*0.0479*	*0.0404*	*0.0315*	*0.0000*	*0.0000*	*1.0000*	*0.1900*
65 Idle Std	*0.0357*	*0.0364*	*0.0315*	*0.0000*	*0.0000*	*1.0000*	*0.1800*
66 Idle Max	*0.0479*	*0.0408*	*0.0322*	*0.0000*	*0.0000*	*1.0000*	*0.1900*
67 Idle Min	*0.0479*	*0.0406*	*0.0319*	*0.0000*	*0.0003*	*1.0000*	*0.1900*

A.3. táblázat: Jellemzőkiválasztási eredmények a WEB adathalmazhoz

Features	IG	SU	GR	Chi²	Relief	ANOVA	Átlag
00 Dst Port	*0.3503*	*0.4714*	*0.4701*	*0.0016*	*0.7010*	*0.0000*	*0.3300*
01 Protocol	*0.0145*	*0.0261*	*0.0264*	*0.0003*	*0.0000*	*0.0000*	*0.0100*
02 Flow Duration	*0.9397*	*0.4138*	*0.4119*	*0.0000*	*0.0000*	*0.9978*	*0.4600*
03 Tot Fwd Pkts	*0.3682*	*0.8426*	*0.8419*	*0.0000*	*0.0000*	*0.9987*	*0.5100*
04 Tot Bwd Pkts	*0.3591*	*0.8066*	*0.8060*	*0.0000*	*0.0000*	*0.9968*	*0.4900*
05 TotLen Fwd Pkts	*0.5910*	*0.7718*	*0.7694*	*0.8340*	*0.0002*	*0.0000*	*0.4900*
06 TotLen Bwd Pkts	*0.5301*	*0.6395*	*0.6375*	*0.0000*	*0.0004*	*0.9996*	*0.4700*
07 Fwd Pkt Len Max	*0.6410*	*0.9079*	*0.9054*	*0.0101*	*0.0716*	*0.0000*	*0.4200*
08 Fwd Pkt Len Min	*0.2197*	*0.5757*	*0.5764*	*0.2142*	*0.1009*	*0.0000*	*0.2800*
09 Fwd Pkt Len Mean	*0.6625*	*0.7844*	*0.7819*	*0.2686*	*0.1090*	*0.0000*	*0.4300*
10 Fwd Pkt Len Std	*0.4986*	*0.8150*	*0.8132*	*0.0127*	*0.0951*	*0.0000*	*0.3700*
11 Bwd Pkt Len Max	*0.5301*	*0.7796*	*0.7777*	*0.1285*	*0.3468*	*0.0000*	*0.4300*
12 Bwd Pkt Len Min	*0.0592*	*0.1399*	*0.1399*	*0.0004*	*0.1646*	*0.0000*	*0.0800*
13 Bwd Pkt Len Mean	*0.4835*	*0.5809*	*0.5791*	*0.0215*	*0.1125*	*0.0000*	*0.3000*
14 Bwd Pkt Len Std	*0.4868*	*1.0000*	*0.9986*	*0.0320*	*0.0732*	*0.0000*	*0.4300*
15 Flow Byts/s	*0.7146*	*0.4244*	*0.4225*	*0.0000*	*0.0003*	*0.9994*	*0.4300*
16 Flow Pkts/s	*0.9397*	*0.4113*	*0.4094*	*0.0001*	*0.0001*	*0.4185*	*0.3600*
17 Flow IAT Mean	*0.9251*	*0.4100*	*0.4081*	*0.0000*	*0.0000*	*0.9999*	*0.4600*
18 Flow IAT Std	*0.7315*	*0.5699*	*0.5676*	*0.0000*	*0.0000*	*0.9978*	*0.4800*
19 Flow IAT Max	*0.9336*	*0.4236*	*0.4216*	*0.0000*	*0.0000*	*0.9999*	*0.4600*
20 Flow IAT Min	*0.5419*	*0.3264*	*0.3251*	*0.0000*	*0.0000*	*0.9471*	*0.3600*
21 Fwd IAT Tot	*1.0000*	*0.5258*	*0.5235*	*0.0000*	*0.0000*	*0.9978*	*0.5100*
22 Fwd IAT Mean	*0.9975*	*0.5276*	*0.5252*	*0.0000*	*0.0000*	*0.9999*	*0.5100*
23 Fwd IAT Std	*0.7190*	*0.6672*	*0.6647*	*0.0000*	*0.0000*	*0.9978*	*0.5100*
24 Fwd IAT Max	*0.9990*	*0.5185*	*0.5162*	*0.0000*	*0.0000*	*0.9999*	*0.5100*
25 Fwd IAT Min	*0.8793*	*0.4167*	*0.4151*	*0.0000*	*0.0000*	*0.9471*	*0.4400*
26 Bwd IAT Tot	*0.4628*	*0.4790*	*0.4773*	*0.0808*	*0.5414*	*0.0000*	*0.3400*
27 Bwd IAT Mean	*0.4628*	*0.4759*	*0.4742*	*0.0000*	*0.0313*	*0.9947*	*0.4100*
28 Bwd IAT Std	*0.4887*	*0.5173*	*0.5155*	*0.0002*	*0.0728*	*0.0986*	*0.2800*
29 Bwd IAT Max	*0.4628*	*0.5011*	*0.4993*	*0.0002*	*0.2184*	*0.0314*	*0.2900*
30 Bwd IAT Min	*0.3182*	*0.3598*	*0.3588*	*0.0000*	*0.0020*	*0.9991*	*0.3400*
31 Fwd PSH Flags	*0.0372*	*0.1460*	*0.1554*	*0.0002*	*0.0000*	*0.0000*	*0.0600*
32 Fwd Header Len	*0.4563*	*0.8962*	*0.8945*	*0.0000*	*0.0001*	*0.9986*	*0.5400*
33 Bwd Header Len	*0.3893*	*0.7174*	*0.7163*	*0.0000*	*0.0000*	*0.9968*	*0.4700*

Features	IG	SU	GR	Chi²	Relief	ANOVA	Átlag
34 Fwd Pkts/s	*0.9878*	*0.4273*	*0.4253*	*0.0001*	*0.1131*	*0.4192*	*0.4000*
35 Bwd Pkts/s	*0.5607*	*0.3480*	*0.3465*	*0.0000*	*0.0094*	*0.9909*	*0.3800*
36 Pkt Len Min	*0.2174*	*0.5795*	*0.5802*	*0.2180*	*0.0688*	*0.0000*	*0.2800*
37 Pkt Len Max	*0.6627*	*0.9371*	*0.9345*	*0.1358*	*0.0962*	*0.0000*	*0.4600*
38 Pkt Len Mean	*0.6625*	*0.6686*	*0.6662*	*0.1615*	*0.1440*	*0.0000*	*0.3800*
39 Pkt Len Std	*0.5292*	*0.5962*	*0.5942*	*0.0286*	*0.1879*	*0.0000*	*0.3200*
40 Pkt Len Var	*0.5292*	*0.5952*	*0.5932*	*0.0244*	*0.0427*	*0.0000*	*0.3000*
41 FIN Flag Cnt	*0.0055*	*0.0944*	*0.1272*	*0.0000*	*0.1042*	*0.0821*	*0.0700*
42 SYN Flag Cnt	*0.0372*	*0.1460*	*0.1554*	*0.0002*	*0.0000*	*0.0000*	*0.0600*
43 RST Flag Cnt	*0.0355*	*0.6946*	*0.7116*	*0.0071*	*0.0000*	*0.0000*	*0.2400*
44 PSH Flag Cnt	*0.0000*	*0.0094*	*0.0095*	*0.0001*	*0.0000*	*0.0017*	*0.0000*
45 ACK Flag Cnt	*0.0000*	*0.0000*	*0.0000*	*0.0000*	*0.0000*	*0.9728*	*0.1600*
46 URG Flag Cnt	*0.0474*	*0.1874*	*0.1980*	*0.0003*	*0.0000*	*0.0000*	*0.0700*
47 ECE Flag Cnt	*0.0355*	*0.6946*	*0.7116*	*0.0071*	*0.0000*	*0.0000*	*0.2400*
48 Down/Up Ratio	*0.0548*	*0.3000*	*0.3027*	*0.0000*	*0.0000*	*1.0000*	*0.2800*
49 Pkt Size Avg	*0.6623*	*0.6788*	*0.6764*	*0.0877*	*0.1267*	*0.0000*	*0.3700*
50 Fwd Seg Size Avg	*0.6625*	*0.7844*	*0.7819*	*0.2686*	*0.0964*	*0.0000*	*0.4300*
51 Bwd Seg Size Avg	*0.4835*	*0.5809*	*0.5791*	*0.0215*	*0.1027*	*0.0000*	*0.2900*
52 Subflow Fwd Pkts	*0.3682*	*0.8426*	*0.8419*	*0.0000*	*0.0001*	*0.9987*	*0.5100*
53 Subflow Fwd Byts	*0.5910*	*0.7718*	*0.7694*	*0.8340*	*0.0002*	*0.0000*	*0.4900*
54 Subflow Bwd Pkts	*0.3591*	*0.8066*	*0.8060*	*0.0000*	*0.0000*	*0.9968*	*0.4900*
55 Subflow Bwd Byts	*0.5301*	*0.6395*	*0.6375*	*0.0000*	*0.0001*	*0.9996*	*0.4700*
56 Init Fwd Win Byts	*0.3630*	*0.7474*	*0.7465*	*0.0008*	*0.4160*	*0.0000*	*0.3800*
57 Init Bwd Win Byts	*0.4967*	*0.9266*	*0.9251*	*0.0016*	*1.0000*	*0.0000*	*0.5600*
58 Fwd Act Data Pkts	*0.2601*	*0.7957*	*0.7963*	*1.0000*	*0.0005*	*0.0000*	*0.4800*
59 Fwd Seg Size Min	*0.0177*	*0.0271*	*0.0274*	*0.0004*	*0.0000*	*0.0003*	*0.0100*
60 Active Mean	*0.2342*	*0.7128*	*0.7144*	*0.0000*	*0.0034*	*0.9999*	*0.4400*
61 Active Std	*0.0225*	*0.0598*	*0.0600*	*0.0000*	*0.0047*	*1.0000*	*0.1900*
62 Active Max	*0.2342*	*0.7141*	*0.7158*	*0.0000*	*0.0047*	*0.9997*	*0.4400*
63 Active Min	*0.2342*	*0.7626*	*0.7648*	*0.0000*	*0.0005*	*1.0000*	*0.4600*
64 Idle Mean	*0.2631*	*0.5090*	*0.5083*	*0.0000*	*0.0001*	*0.9978*	*0.3800*
65 Idle Std	*0.2255*	*0.9968*	*1.0000*	*0.0000*	*0.0000*	*0.9978*	*0.5400*
66 Idle Max	*0.2631*	*0.4767*	*0.4760*	*0.0000*	*0.0000*	*0.9999*	*0.3700*
67 Idle Min	*0.2631*	*0.4756*	*0.4750*	*0.0000*	*0.0002*	*0.9999*	*0.3700*

A.4. táblázat: Jellemzőkiválasztási eredmények az XSS adathalmazhoz

Features	IG	SU	GR	Chi²	Relief	ANOVA	Átlag
00 Dst Port	*0.2524*	*0.1869*	*0.1865*	*0.0003*	*1.0000*	*0.0000*	*0.2700*
01 Protocol	*0.0618*	*0.1611*	*0.1618*	*0.0004*	*0.0889*	*0.0000*	*0.0800*
02 Flow Duration	*0.9600*	*0.2131*	*0.2125*	*0.0000*	*0.0000*	*0.9994*	*0.4000*
03 Tot Fwd Pkts	*0.5510*	*0.7613*	*0.7607*	*0.0000*	*0.0002*	*0.9995*	*0.5100*
04 Tot Bwd Pkts	*0.5578*	*0.7743*	*0.7737*	*0.0000*	*0.0001*	*1.0000*	*0.5200*
05 TotLen Fwd Pkts	*0.5677*	*0.4366*	*0.4358*	*0.0000*	*0.0004*	*0.9813*	*0.4000*
06 TotLen Bwd Pkts	*0.5801*	*0.4317*	*0.4309*	*0.0000*	*0.0001*	*1.0000*	*0.4100*
07 Fwd Pkt Len Max	*0.5687*	*0.4804*	*0.4796*	*0.0009*	*0.0565*	*0.0000*	*0.2600*
08 Fwd Pkt Len Min	*0.0785*	*0.0884*	*0.0883*	*0.0003*	*0.0771*	*0.0000*	*0.0600*
09 Fwd Pkt Len Mean	*0.5677*	*0.4169*	*0.4161*	*0.0216*	*0.1155*	*0.0000*	*0.2600*
10 Fwd Pkt Len Std	*0.5412*	*0.5413*	*0.5405*	*0.0137*	*0.1020*	*0.0000*	*0.2900*
11 Bwd Pkt Len Max	*0.5801*	*0.5414*	*0.5406*	*1.0000*	*0.4163*	*0.0000*	*0.5100*
12 Bwd Pkt Len Min	*0.0729*	*0.0609*	*0.0608*	*0.0002*	*0.1262*	*0.0000*	*0.0500*
13 Bwd Pkt Len	*0.5684*	*0.4209*	*0.4202*	*1.0000*	*0.2215*	*0.0000*	*0.4400*
14 Bwd Pkt Len Std	*0.4906*	*0.6739*	*0.6733*	*0.0316*	*0.1848*	*0.0000*	*0.3400*
15 Flow Byts/s	*0.5807*	*0.2037*	*0.2031*	*0.0000*	*0.0052*	*0.9995*	*0.3300*
16 Flow Pkts/s	*0.9844*	*0.2263*	*0.2257*	*0.0000*	*0.0009*	*0.8949*	*0.3900*
17 Flow IAT Mean	*0.9479*	*0.2215*	*0.2209*	*0.0000*	*0.0001*	*0.9813*	*0.4000*
18 Flow IAT Std	*0.7825*	*0.3303*	*0.3295*	*0.0000*	*0.0000*	*0.9994*	*0.4100*
19 Flow IAT Max	*0.9296*	*0.2237*	*0.2231*	*0.0000*	*0.0000*	*0.9994*	*0.4000*
20 Flow IAT Min	*0.4017*	*0.1338*	*0.1334*	*0.0000*	*0.0001*	*0.9994*	*0.2800*
21 Fwd IAT Tot	*0.9924*	*0.2823*	*0.2816*	*0.0000*	*0.0000*	*0.9994*	*0.4300*
22 Fwd IAT Mean	*1.0000*	*0.2955*	*0.2948*	*0.0000*	*0.0001*	*0.9813*	*0.4300*
23 Fwd IAT Std	*0.8273*	*0.3797*	*0.3788*	*0.0000*	*0.0000*	*0.9994*	*0.4300*
24 Fwd IAT Max	*0.9909*	*0.2883*	*0.2875*	*0.0000*	*0.0000*	*0.9994*	*0.4300*
25 Fwd IAT Min	*0.7743*	*0.1976*	*0.1971*	*0.0000*	*0.0001*	*0.9994*	*0.3600*
26 Bwd IAT Tot	*0.5440*	*0.3113*	*0.3106*	*0.0656*	*0.1031*	*0.0000*	*0.2200*
27 Bwd IAT Mean	*0.5440*	*0.3110*	*0.3103*	*0.0000*	*0.0388*	*0.9999*	*0.3700*
28 Bwd IAT Std	*0.4925*	*0.3383*	*0.3376*	*0.0000*	*0.0557*	*0.7965*	*0.3400*
29 Bwd IAT Max	*0.5440*	*0.3250*	*0.3243*	*0.0000*	*0.1715*	*0.6581*	*0.3400*
30 Bwd IAT Min	*0.1920*	*0.1670*	*0.1666*	*0.0000*	*0.0012*	*1.0000*	*0.2500*
31 Fwd PSH Flags	*0.0350*	*0.0848*	*0.0874*	*0.0000*	*0.0000*	*0.0030*	*0.0400*
32 Fwd Header Len	*0.6111*	*0.6640*	*0.6631*	*0.0000*	*0.0003*	*1.0000*	*0.4900*
33 Bwd Header Len	*0.5525*	*0.6538*	*0.6531*	*0.0000*	*0.0001*	*1.0000*	*0.4800*

Features	IG	SU	GR	Chi²	Relief	ANOVA	Átlag
34 Fwd Pkts/s	*0.9624*	*0.2304*	*0.2298*	*0.0000*	*0.0781*	*0.8953*	*0.4000*
35 Bwd Pkts/s	*0.5796*	*0.2127*	*0.2122*	*0.0000*	*0.0052*	*0.9997*	*0.3300*
36 Pkt Len Min	*0.0757*	*0.0885*	*0.0885*	*0.0003*	*0.0617*	*0.0000*	*0.0500*
37 Pkt Len Max	*0.5695*	*0.5090*	*0.5082*	*0.9659*	*0.1688*	*0.0000*	*0.4500*
38 Pkt Len Mean	*0.5695*	*0.3529*	*0.3521*	*0.0673*	*0.1847*	*0.0000*	*0.2500*
39 Pkt Len Std	*0.5601*	*0.3808*	*0.3801*	*0.9478*	*0.3471*	*0.0000*	*0.4400*
40 Pkt Len Var	*0.5601*	*0.3801*	*0.3793*	*0.9523*	*0.0895*	*0.0000*	*0.3900*
41 FIN Flag Cnt	*0.0000*	*0.0436*	*0.0526*	*0.0000*	*0.0000*	*0.2880*	*0.0600*
42 SYN Flag Cnt	*0.0350*	*0.0848*	*0.0874*	*0.0000*	*0.2520*	*0.0030*	*0.0800*
43 RST Flag Cnt	*0.1129*	*0.4846*	*0.4896*	*0.0016*	*0.0000*	*0.0000*	*0.1800*
44 PSH Flag Cnt	*0.0320*	*0.0000*	*0.0000*	*0.0001*	*0.0000*	*0.0005*	*0.0100*
45 ACK Flag Cnt	*0.0609*	*0.0278*	*0.0279*	*0.0002*	*0.0000*	*0.0000*	*0.0200*
46 URG Flag Cnt	*0.0404*	*0.0887*	*0.0912*	*0.0001*	*0.3360*	*0.0014*	*0.0900*
47 ECE Flag Cnt	*0.1129*	*0.4846*	*0.4896*	*0.0016*	*0.0000*	*0.0000*	*0.1800*
48 Down/Up Ratio	*0.1600*	*0.2914*	*0.2924*	*0.0000*	*0.0020*	*1.0000*	*0.2900*
49 Pkt Size Avg	*0.5693*	*0.3566*	*0.3558*	*0.0536*	*0.1601*	*0.0000*	*0.2500*
50 Fwd Seg Size Avg	*0.5677*	*0.4169*	*0.4161*	*0.0216*	*0.1017*	*0.0000*	*0.2500*
51 Bwd Seg Size Avg	*0.5684*	*0.4209*	*0.4202*	*1.0000*	*0.1985*	*0.0000*	*0.4300*
52 Subflow Fwd Pkts	*0.5510*	*0.7613*	*0.7607*	*0.0000*	*0.0003*	*0.9995*	*0.5100*
53 Subflow Fwd Byts	*0.5677*	*0.4366*	*0.4358*	*0.0000*	*0.0006*	*0.9813*	*0.4000*
54 Subflow Bwd Pkts	*0.5578*	*0.7743*	*0.7737*	*0.0000*	*0.0001*	*1.0000*	*0.5200*
55 Subflow Bwd Byts	*0.5801*	*0.4317*	*0.4309*	*0.0000*	*0.0000*	*1.0000*	*0.4100*
56 Init Fwd Win Byts	*0.5095*	*0.7152*	*0.7145*	*0.0002*	*0.8432*	*0.0004*	*0.4600*
57 Init Bwd Win Byts	*0.5749*	*0.6102*	*0.6095*	*0.0003*	*0.8340*	*0.0000*	*0.4400*
58 Fwd Act Data Pkts	*0.5411*	*1.0000*	*1.0000*	*0.0000*	*0.0013*	*0.9813*	*0.5900*
59 Fwd Seg Size Min	*0.0674*	*0.1574*	*0.1581*	*0.0004*	*0.0000*	*0.0000*	*0.0600*
60 Active Mean	*0.1027*	*0.0576*	*0.0576*	*0.0000*	*0.0024*	*1.0000*	*0.2000*
61 Active Std	*0.0787*	*0.0159*	*0.0158*	*0.0000*	*0.0010*	*1.0000*	*0.1900*
62 Active Max	*0.1027*	*0.0577*	*0.0577*	*0.0000*	*0.0027*	*1.0000*	*0.2000*
63 Active Min	*0.1027*	*0.0627*	*0.0627*	*0.0000*	*0.0005*	*1.0000*	*0.2000*
64 Idle Mean	*0.0707*	*0.0400*	*0.0398*	*0.0000*	*0.0001*	*0.9994*	*0.1900*
65 Idle Std	*0.0918*	*0.0634*	*0.0634*	*0.0000*	*0.0000*	*0.9994*	*0.2000*
66 Idle Max	*0.0707*	*0.0410*	*0.0408*	*0.0000*	*0.0000*	*0.9994*	*0.1900*
67 Idle Min	*0.0707*	*0.0384*	*0.0383*	*0.0000*	*0.0002*	*0.9994*	*0.1900*

A.5. táblázat: Jellemzőkiválasztási eredmények az SQL adathalmazhoz

Features	IG	SU	GR	Chi²	Relief	ANO	Átlag
00 Dst Port	*0.2791*	*0.2310*	*0.2308*	*0.0001*	*0.8879*	*0.019*	*0.2700*
01 Protocol	*0.0773*	*0.2659*	*0.2665*	*0.0002*	*1.0000*	*0.000*	*0.2700*
02 Flow Duration	*0.8699*	*0.3206*	*0.3204*	*0.0000*	*0.0001*	*0.999*	*0.4200*
03 Tot Fwd Pkts	*0.1657*	*0.3191*	*0.3191*	*0.0000*	*0.0001*	*0.999*	*0.3000*
04 Tot Bwd Pkts	*0.2438*	*0.2528*	*0.2528*	*0.0000*	*0.0000*	*1.000*	*0.2900*
05 TotLen Fwd Pkts	*0.6314*	*0.3988*	*0.3987*	*0.0000*	*0.0001*	*0.999*	*0.4000*
06 TotLen Bwd Pkts	*0.7012*	*0.5136*	*0.5134*	*0.0000*	*0.0004*	*1.000*	*0.4500*
07 Fwd Pkt Len Max	*0.6351*	*0.4338*	*0.4336*	*0.0005*	*0.0231*	*0.000*	*0.2500*
08 Fwd Pkt Len Min	*0.0595*	*0.1020*	*0.1019*	*0.0000*	*0.0386*	*1.000*	*0.2200*
09 Fwd Pkt Len Mean	*0.6518*	*0.4908*	*0.4906*	*0.0013*	*0.0446*	*0.000*	*0.2800*
10 Fwd Pkt Len Std	*0.6815*	*0.7874*	*0.7873*	*0.0024*	*0.0486*	*0.000*	*0.3800*
11 Bwd Pkt Len Max	*0.7113*	*0.7422*	*0.7421*	*1.0000*	*0.1135*	*0.000*	*0.5500*
12 Bwd Pkt Len Min	*0.0416*	*0.0789*	*0.0788*	*0.0001*	*0.1195*	*0.080*	*0.0700*
13 Bwd Pkt Len Mean	*0.6551*	*0.5258*	*0.5256*	*0.0013*	*0.0520*	*0.000*	*0.2900*
14 Bwd Pkt Len Std	*0.6407*	*1.0000*	*1.0000*	*0.8184*	*0.0467*	*0.000*	*0.5800*
15 Flow Byts/s	*0.6748*	*0.2973*	*0.2971*	*0.0000*	*0.0002*	*0.999*	*0.3800*
16 Flow Pkts/s	*0.9090*	*0.3231*	*0.3229*	*0.0000*	*0.0455*	*0.997*	*0.4300*
17 Flow IAT Mean	*0.8997*	*0.3195*	*0.3193*	*0.0000*	*0.0000*	*0.999*	*0.4200*
18 Flow IAT Std	*0.7514*	*0.4409*	*0.4406*	*0.0000*	*0.0000*	*0.983*	*0.4400*
19 Flow IAT Max	*0.8722*	*0.3331*	*0.3328*	*0.0000*	*0.0000*	*0.983*	*0.4200*
20 Flow IAT Min	*0.3634*	*0.1682*	*0.1681*	*0.0000*	*0.0000*	*0.999*	*0.2800*
21 Fwd IAT Tot	*0.9870*	*0.4123*	*0.4121*	*0.0000*	*0.0001*	*0.999*	*0.4700*
22 Fwd IAT Mean	*1.0000*	*0.4195*	*0.4193*	*0.0000*	*0.0000*	*0.999*	*0.4700*
23 Fwd IAT Std	*0.7016*	*0.4946*	*0.4943*	*0.0000*	*0.0000*	*0.983*	*0.4500*
24 Fwd IAT Max	*0.9918*	*0.4151*	*0.4148*	*0.0000*	*0.0000*	*0.983*	*0.4700*
25 Fwd IAT Min	*0.8250*	*0.3029*	*0.3027*	*0.0000*	*0.0000*	*0.999*	*0.4100*
26 Bwd IAT Tot	*0.6830*	*0.4702*	*0.4700*	*0.0001*	*0.2368*	*0.512*	*0.4000*
27 Bwd IAT Mean	*0.6830*	*0.4704*	*0.4701*	*0.0000*	*0.0126*	*1.000*	*0.4400*
28 Bwd IAT Std	*0.6421*	*0.5120*	*0.5118*	*0.0000*	*0.0055*	*0.990*	*0.4400*
29 Bwd IAT Max	*0.6830*	*0.4921*	*0.4919*	*0.0000*	*0.1047*	*0.977*	*0.4600*
30 Bwd IAT Min	*0.6195*	*0.5528*	*0.5526*	*0.0000*	*0.0011*	*1.000*	*0.4500*
31 Fwd PSH Flags	*0.0412*	*0.1130*	*0.1143*	*0.0000*	*0.3429*	*0.066*	*0.1100*
32 Fwd Header Len	*0.2088*	*0.3250*	*0.3249*	*0.0000*	*0.0001*	*0.999*	*0.3100*
33 Bwd Header Len	*0.2527*	*0.2841*	*0.2841*	*0.0000*	*0.0014*	*1.000*	*0.3000*

Features	IG	SU	GR	Chi²	Relief	ANO	Átlag
34 Fwd Pkts/s	*0.9238*	*0.3359*	*0.3357*	*0.0000*	*0.0232*	*0.997*	*0.4400*
35 Bwd Pkts/s	*0.7971*	*0.3550*	*0.3547*	*0.0000*	*0.0036*	*1.000*	*0.4200*
36 Pkt Len Min	*0.0565*	*0.1020*	*0.1020*	*0.0000*	*0.0369*	*0.998*	*0.2200*
37 Pkt Len Max	*0.6748*	*0.6943*	*0.6941*	*0.9483*	*0.0650*	*0.000*	*0.5100*
38 Pkt Len Mean	*0.6748*	*0.4788*	*0.4786*	*0.0006*	*0.0651*	*0.000*	*0.2800*
39 Pkt Len Std	*0.6570*	*0.5581*	*0.5578*	*0.2200*	*0.0993*	*0.000*	*0.3500*
40 Pkt Len Var	*0.6570*	*0.5576*	*0.5574*	*0.2155*	*0.0192*	*0.000*	*0.3300*
41 FIN Flag Cnt	*0.0056*	*0.0687*	*0.0735*	*0.0000*	*0.0490*	*0.516*	*0.1200*
42 SYN Flag Cnt	*0.0412*	*0.1130*	*0.1143*	*0.0000*	*0.2450*	*0.066*	*0.1000*
43 RST Flag Cnt	*0.2609*	*0.9650*	*0.9696*	*0.0013*	*0.0000*	*0.000*	*0.3700*
44 PSH Flag Cnt	*0.0899*	*0.0765*	*0.0766*	*0.0001*	*0.0000*	*0.000*	*0.0400*
45 ACK Flag Cnt	*0.0000*	*0.0029*	*0.0028*	*0.0000*	*0.0000*	*0.134*	*0.0200*
46 URG Flag Cnt	*0.0000*	*0.0000*	*0.0000*	*0.0000*	*0.0000*	*0.496*	*0.0800*
47 ECE Flag Cnt	*0.2609*	*0.9650*	*0.9696*	*0.0013*	*0.0000*	*0.000*	*0.3700*
48 Down/Up Ratio	*0.0000*	*0.0458*	*0.0458*	*0.0000*	*0.0006*	*1.000*	*0.1800*
49 Pkt Size Avg	*0.6748*	*0.4718*	*0.4715*	*0.0016*	*0.0419*	*0.000*	*0.2800*
50 Fwd Seg Size Avg	*0.6518*	*0.4908*	*0.4906*	*0.0013*	*0.0496*	*0.000*	*0.2800*
51 Bwd Seg Size Avg	*0.6551*	*0.5258*	*0.5256*	*0.0013*	*0.0679*	*0.000*	*0.3000*
52 Subflow Fwd Pkts	*0.1657*	*0.3191*	*0.3191*	*0.0000*	*0.0001*	*0.999*	*0.3000*
53 Subflow Fwd Byts	*0.6314*	*0.3988*	*0.3987*	*0.0000*	*0.0001*	*0.999*	*0.4000*
54 Subflow Bwd Pkts	*0.2438*	*0.2528*	*0.2528*	*0.0000*	*0.0001*	*1.000*	*0.2900*
55 Subflow Bwd Byts	*0.7012*	*0.5136*	*0.5134*	*0.0000*	*0.0000*	*1.000*	*0.4500*
56 Init Fwd Win Byts	*0.5362*	*0.7463*	*0.7463*	*0.0002*	*0.3963*	*0.004*	*0.4000*
57 Init Bwd Win Byts	*0.6901*	*0.8788*	*0.8788*	*0.0004*	*0.5552*	*0.000*	*0.5000*
58 Fwd Act Data Pkts	*0.1605*	*0.1233*	*0.1233*	*0.0000*	*0.0003*	*0.999*	*0.2300*
59 Fwd Seg Size Min	*0.0832*	*0.2590*	*0.2595*	*0.0002*	*0.2531*	*0.000*	*0.1400*
60 Active Mean	*0.1104*	*0.0314*	*0.0314*	*0.0000*	*0.0007*	*1.000*	*0.2000*
61 Active Std	*0.0858*	*0.0309*	*0.0309*	*0.0000*	*0.0015*	*1.000*	*0.1900*
62 Active Max	*0.1104*	*0.0315*	*0.0315*	*0.0000*	*0.0036*	*1.000*	*0.2000*
63 Active Min	*0.1104*	*0.0342*	*0.0342*	*0.0000*	*0.0001*	*1.000*	*0.2000*
64 Idle Mean	*0.2085*	*0.0336*	*0.0336*	*0.0000*	*0.0000*	*0.983*	*0.2100*
65 Idle Std	*0.1003*	*0.0321*	*0.0320*	*0.0000*	*0.0000*	*0.983*	*0.1900*
66 Idle Max	*0.2085*	*0.0344*	*0.0343*	*0.0000*	*0.0000*	*0.983*	*0.2100*
67 Idle Min	*0.2085*	*0.0340*	*0.0339*	*0.0000*	*0.0002*	*0.988*	*0.2100*

A.6. táblázat: Jellemzőcsoportok a küszöbértékekhez

Rangsorolási küszöbérték	FTP	SSH	WEB	XSS	SQL
0,05	23, 27, 30, 43, 46, 47, 08, 12, 18, 29, 36, 26, 10, 06, 15, 55, 14, 38, 39, 04, 49, 54, 13, 51, 03, 09, 48, 50, 52, 01, 45, 11, 32, 58, 33, 21, 22, 24, 25, 40, 05, 20, 53, 07, 37, 57, 34, 16, 02, 17, 19, 35, 00, 44, 56, 59	42,08,47,36,43,12,30,27,28,15,29, 06,39,55,25,26,38,49,03,04,09,13, 50,52,54,65,51,64,66,67,01,34,18, 23,35,14,46,16,10,20,33,40,05,21, 22,24,53,58,07,11,32,37,00,02,17, 19,57,56,59	31,42,41,46,12,45,61,43,47,08,28, 36,48,29,51,13,40,39,00,26,30,16, 20,10,49,66,67,35,38,56,64,34,27, 07,09,11,14,15,50,25,60,62,02,17, 19,37,63,06,33,55,18,58,04,05,53, 54,03,21,22,23,24,52,32,65,57	12,36,08,41,59,01,42,46,43,47,61, 64,66,67,60,62,63,65,26,30,38,49, 50,07,09,00,20,10,48,15,35,14,28, 29,25,27,16,40,02,05,17,19,34,53, 06,18,55,21,22,23,24,51,13,39,57, 37,56,33,32,03,11,52,04,54,58	12,46,42,31,41,59,48,61,65,60,62,63 ,64,66,67,08,36,58,07,00,01,09,20,3 8,49,50,04,13,54,03,33,51,52,32,40, 39,43,47,10,15,05,26,53,56,25,02,17 ,19,35,16,18,27,28,34,06,23,30,55,2 9,21,22,24,57,37,11,14
0,10	06, 15, 55, 14, 38, 39, 04, 49, 54, 13, 51, 03, 09, 48, 50, 52, 01, 45, 11, 32, 58, 33, 21, 22, 24, 25, 40, 05, 20, 53, 07, 37, 57, 34, 16, 02, 17, 19, 35, 00, 44, 56, 59	30,27,28,15,29,06,39,55,25,26,38, 49,03,04,09,13,50,52,54,65,51,64, 66,67,01,34,18,23,35,14,46,16,10, 20,33,40,05,21,22,24,53,58,07,11, 32,37,00,02,17,19,57,56,59	45,61,43,47,08,28,36,48,29,51,13, 40,39,00,26,30,16,20,10,49,66,67, 35,38,56,64,34,27,07,09,11,14,15, 50,25,60,62,02,17,19,37,63,06,33, 55,18,58,04,05,53,54,03,21,22,23, 24,52,32,65,57	43,47,61,64,66,67,60,62,63,65,26, 30,38,49,50,07,09,00,20,10,48,15, 35,14,28,29,25,27,16,40,02,05,17, 19,34,53,06,18,55,21,22,23,24,51, 13,39,57,37,56,33,32,03,11,52,04, 54,58	42,31,41,59,48,61,65,60,62,63,64,66 ,67,08,36,58,07,00,01,09,20,38,49,5 0,04,13,54,03,33,51,52,32,40,39,43, 47,10,15,05,26,53,56,25,02,17,19,35 ,16,18,27,28,34,06,23,30,55,29,21,2 2,24,57,37,11,14
0,15	03, 09, 48, 50, 52 ,01 ,45, 11, 32, 58, 33, 21, 22, 24, 25, 40, 05, 20, 53, 07, 37, 57, 34, 16, 02, 17, 19, 35, 00, 44, 56, 59	06,39,55,25,26,38,49,03,04,09,13, 50,52,54,65,51,64,66,67,01,34,18, 23,35,14,46,16,10,20,33,40,05,21, 22,24,53,58,07,11,32,37,00,02,17, 19,57,56,59	45,61,43,47,08,28,36,48,29,51,13, 40,39,00,26,30,16,20,10,49,66,67, 35,38,56,64,34,27,07,09,11,14,15, 50,25,60,62,02,17,19,37,63,06,33, 55,18,58,04,05,53,54,03,21,22,23, 24,52,32,65,57	43,47,61,64,66,67,60,62,63,65,26, 30,38,49,50,07,09,00,20,10,48,15, 35,14,28,29,25,27,16,40,02,05,17, 19,34,53,06,18,55,21,22,23,24,51, 13,39,57,37,56,33,32,03,11,52,04, 54,58	48,61,65,60,62,63,64,66,67,08,36,58 ,07,00,01,09,20,38,49,50,04,13,54,0 3,33,51,52,32,40,39,43,47,10,15,05, 26,53,56,25,02,17,19,35,16,18,27,28 ,34,06,23,30,55,29,21,22,24,57,37,1 1,14
0,20	58, 33, 21, 22, 24, 25, 40, 05, 20, 53, 07, 37, 57, 34, 16, 02, 17, 19, 35, 00, 44, 56, 59	01,34,18,23,35,14,46,16,10,20,33, 40,05,21,22,24,53,58,07,11,32,37, 00,02,17,19,57,56,59	43,47,08,28,36,48,29,51,13,40,39, 00,26,30,16,20,10,49,66,67,35,38, 56,64,34,27,07,09,11,14,15,50,25, 60,62,02,17,19,37,63,06,33,55,18, 58,04,05,53,54,03,21,22,23,24,52, 32,65,57	60,62,63,65,26,30,38,49,50,07,09, 00,20,10,48,15,35,14,28,29,25,27, 16,40,02,05,17,19,34,53,06,18,55, 21,22,23,24,51,13,39,57,37,56,33, 32,03,11,52,04,54,58	60,62,63,64,66,67,08,36,58,07,00,01 ,09,20,38,49,50,04,13,54,03,33,51,5 2,32,40,39,43,47,10,15,05,26,53,56, 25,02,17,19,35,16,18,27,28,34,06,23 ,30,55,29,21,22,24,57,37,11,14

0,25	21, 22, 24, 25, 40, 05, 20, 53, 07, 37, 57, 34, 16, 02, 17, 19, 35, 00, 44, 56 ,59	16,10,20,33,40,05,21,22,24,53,58, 07,11,32,37,00,02,17,19,57,56,59	08,28,36,48,29,51,13,40,39,00,26, 30,16,20,10,49,66,67,35,38,56,64, 34,27,07,09,11,14,15,50,25,60,62, 02,17,19,37,63,06,33,55,18,58,04, 05,53,54,03,21,22,23,24,52,32,65, 57	30,38,49,50,07,09,00,20,10,48,15, 35,14,28,29,25,27,16,40,02,05,17, 19,34,53,06,18,55,21,22,23,24,51, 13,39,57,37,56,33,32,03,11,52,04, 54,58	07,00,01,09,20,38,49,50,04,13,54,03 ,33,51,52,32,40,39,43,47,10,15,05,2 6,53,56,25,02,17,19,35,16,18,27,28, 34,06,23,30,55,29,21,22,24,57,37,11 ,14
0,30	07, 37, 57, 34, 16, 02, 17, 19, 35, 00, 44, 56, 59	05,21,22,24,53,58,07,11,32,37,00, 02,17,19,57,56,59	13,40,39,00,26,30,16,20,10,49,66, 67,35,38,56,64,34,27,07,09,11,14, 15,50,25,60,62,02,17,19,37,63,06, 33,55,18,58,04,05,53,54,03,21,22, 23,24,52,32,65,57	15,35,14,28,29,25,27,16,40,02,05, 17,19,34,53,06,18,55,21,22,23,24, 51,13,39,57,37,56,33,32,03,11,52, 04,54,58	03,33,51,52,32,40,39,43,47,10,15,05 ,26,53,56,25,02,17,19,35,16,18,27,2 8,34,06,23,30,55,29,21,22,24,57,37, 11,14
0,35	02,17,19,35, 00,44,56,59	00,02,17, 19,57, 56, 59	16, 20, 10, 49, 66, 67, 35, 38, 56, 64, 34, 27, 07, 09, 11, 14, 15, 50, 25, 60, 62, 02, 17, 19, 37, 63, 06, 33, 55, 18, 58, 04, 05, 53, 54, 03, 21, 22, 23, 24, 52, 32, 65, 57	25, 27, 16, 40, 02, 05, 17, 19, 34, 53, 06, 18, 55, 21, 22, 23, 24, 51, 13, 39, 57, 37, 56, 33, 32, 03, 11, 52, 04, 54, 58	39, 43, 47, 10, 15, 05, 26, 53, 56, 25, 02, 17, 19, 35, 16, 18, 27, 28, 34, 06, 23, 30, 55, 29, 21, 22, 24, 57, 37, 11, 14
0,40	44,56,59	56, 59	34, 27, 07, 09, 11, 14, 15, 50, 25, 60, 62, 02, 17, 19, 37, 63, 06, 33, 55, 18, 58, 04, 05, 53, 54, 03, 21, 22, 23, 24, 52, 32, 65, 57	02, 05, 17, 19, 34, 53, 06, 18, 55, 21, 22, 23, 24, 51, 13, 39, 57, 37, 56, 33, 32, 03, 11, 52, 04, 54, 58	05, 26, 53, 56, 25, 02, 17, 19, 35, 16, 18, 27, 28, 34, 06, 23, 30, 55, 29, 21, 22, 24, 57, 37, 11, 14
0,45	56, 59	56, 59	02, 17, 19, 37, 63, 06, 33, 55, 18, 58, 04, 05, 53, 54, 03, 21, 22, 23, 24, 52, 32, 65, 57	37, 56, 33, 32, 03, 11, 52, 04, 54, 58	06, 23, 30, 55, 29, 21, 22, 24, 57, 37, 11, 14
0,50	56, 59	59	03, 21 ,22, 23, 24, 52, 32, 65, 57	03, 11, 52, 04, 54, 58	57, 37, 11, 14
0,55	56, 59	59	57	58	11, 14

A.7. táblázat: Osztályozó eredmények az FTP adathalmazhoz

Features		Train Dataset				Test Dataset			
	Models	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
02,17,19,35,00,44,56,59	Naive Bayes	*0.99974*	*0.99912*	*0.99972*	*0.99942*	*0.99977*	*0.99928*	*0.99969*	*0.99948*
	Logistic Regression	*0.99943*	*0.99747*	*1.00000*	*0.99873*	*0.99945*	*0.99758*	*1.00000*	*0.99879*
	Tree	*1.00000*	*1.00000*	*1.00000*	*1.00000*	*0.99999*	*0.99995*	*1.00000*	*0.99997*
	SVM	*0.99973*	*0.99881*	*1.00000*	*0.99941*	*0.99973*	*0.99881*	*1.00000*	*0.99941*
	Random Forest	*1.00000*	*1.00000*	*1.00000*	*1.00000*	*1.00000*	*1.00000*	*1.00000*	*1.00000*
44,56,59	Naive Bayes	*0.99875*	*0.99450*	*1.00000*	*0.99724*	*0.99881*	*0.99475*	*1.00000*	*0.99737*
	Logistic Regression	*0.99871*	*0.99432*	*1.00000*	*0.99715*	*0.99875*	*0.99450*	*1.00000*	*0.99724*
	Tree	*0.99999*	*0.99997*	*1.00000*	*0.99999*	*0.99998*	*0.99990*	*1.00000*	*0.99995*
	SVM	*0.99971*	*0.99873*	*1.00000*	*0.99937*	*0.99973*	*0.99881*	*1.00000*	*0.99941*
	Random Forest	*0.99999*	*0.99997*	*1.00000*	*0.99999*	*0.99998*	*0.99990*	*1.00000*	*0.99995*
56,59	Naive Bayes	*0.99875*	*0.99447*	*1.00000*	*0.99723*	*0.99881*	*0.99475*	*1.00000*	*0.99737*
	Logistic Regression	*0.99914*	*0.99619*	*1.00000*	*0.99809*	*0.99918*	*0.99639*	*1.00000*	*0.99819*
	Tree	*0.99999*	*0.99997*	*1.00000*	*0.99999*	*0.99998*	*0.99990*	*1.00000*	*0.99995*
	SVM	*0.99971*	*0.99873*	*1.00000*	*0.99937*	*0.99973*	*0.99881*	*1.00000*	*0.99941*
	Random Forest	*0.99999*	*0.99997*	*1.00000*	*0.99999*	*0.99998*	*0.99990*	*1.00000*	*0.99995*

A.8. táblázat: Osztályozó eredmények az SSH adathalmazhoz

Features		Train Dataset				Test Dataset			
	Models	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
00,02,17,19,57,56,59	Naive Bayes	*0.99998*	*0.99992*	*0.99997*	*0.99995*	*0.99993*	*0.99979*	*0.99989*	*0.99984*
	Logistic Regression	*0.99691*	*0.98615*	*1.00000*	*0.99303*	*0.99691*	*0.98617*	*1.00000*	*0.99304*
	Tree	*0.99999*	*0.99997*	*1.00000*	*0.99999*	*1.00000*	*1.00000*	*1.00000*	*1.00000*
	SVM	*0.99979*	*0.99928*	*0.99979*	*0.99953*	*0.99985*	*0.99947*	*0.99984*	*0.99965*
	Random Forest	*0.99999*	*0.99997*	*1.00000*	*0.99999*	*0.99999*	*0.99995*	*1.00000*	*0.99997*
56,59	Naive Bayes	*0.99804*	*0.99118*	*1.00000*	*0.99557*	*0.99811*	*0.99149*	*1.00000*	*0.99573*
	Logistic Regression	*0.99544*	*0.97971*	*1.00000*	*0.98975*	*0.99541*	*0.97958*	*1.00000*	*0.98969*
	Tree	*0.99955*	*0.99798*	*1.00000*	*0.99899*	*0.99957*	*0.99803*	*1.00000*	*0.99901*
	SVM	*0.88854*	*0.99340*	*0.49744*	*0.66293*	*0.88968*	*0.99336*	*0.50264*	*0.66752*
	Random Forest	*0.99955*	*0.99798*	*1.00000*	*0.99899*	*0.99959*	*0.99814*	*1.00000*	*0.99907*
59	Naive Bayes	*0.99535*	*0.97933*	*1.00000*	*0.98956*	*0.99529*	*0.97907*	*1.00000*	*0.98942*
	Logistic Regression	*0.99544*	*0.97971*	*1.00000*	*0.98975*	*0.99541*	*0.97958*	*1.00000*	*0.98969*
	Tree	*0.99720*	*0.98765*	*0.99979*	*0.99368*	*0.99700*	*0.98674*	*0.99984*	*0.99325*
	SVM	*0.99717*	*0.98752*	*0.99979*	*0.99362*	*0.99698*	*0.98664*	*0.99984*	*0.99320*
	Random Forest	*0.99720*	*0.98765*	*0.99979*	*0.99368*	*0.99700*	*0.98674*	*0.99984*	*0.99325*

A.9. táblázat: Osztályozó eredmények a WEB adathalmazhoz

Features		Train Dataset				Test Dataset			
	Models	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
16, 20, 10, 49, 66, 67, 35, 38, 56, 64, 34, 27, 07, 09, 11, 14, 15, 50, 25, 60, 62, 02, 17, 19, 37, 63, 06, 33, 55, 18, 58, 04, 05, 53, 54, 03, 21, 22, 23, 24, 52, 32, 65, 57	Naive Bayes	*0.924800*	*0.013001*	*0.672668*	*0.025508*	*0.924926*	*0.025834*	*0.672668*	*0.049758*
	Logistic Regression	*0.998324*	*0.000000*	*0.000000*	*0.000000*	*0.996877*	*0.000000*	*0.000000*	*0.000000*
	Tree	*0.999940*	*0.989967*	*0.968903*	*0.979322*	*0.999723*	*0.938193*	*0.968903*	*0.953301*
	SVM	*0.327255*	*0.000773*	*0.355155*	*0.001542*	*0.326541*	*0.001543*	*0.355155*	*0.003072*
	Random Forest	*0.999634*	*0.991416*	*0.756137*	*0.857939*	*0.999283*	*0.997840*	*0.756137*	*0.860335*
34, 27, 07, 09, 11, 14, 15, 50, 25, 60, 62, 02, 17, 19, 37, 63, 06, 33, 55, 18, 58, 04, 05, 53, 54, 03, 21, 22, 23, 24, 52, 32, 65, 57	Naive Bayes	*0.874013*	*0.007258*	*0.626841*	*0.014351*	*0.872588*	*0.014293*	*0.626841*	*0.027948*
	Logistic Regression	*0.998355*	*0.000000*	*0.000000*	*0.000000*	*0.996887*	*0.000000*	*0.000000*	*0.000000*
	Tree	*0.999497*	*0.983133*	*0.667758*	*0.795322*	*0.998847*	*0.914798*	*0.667758*	*0.771996*
	SVM	*0.317951*	*0.001676*	*0.782324*	*0.003345*	*0.317053*	*0.003339*	*0.782324*	*0.006650*
	Random Forest	*0.999193*	*0.979021*	*0.458265*	*0.624303*	*0.998398*	*0.985915*	*0.458265*	*0.625698*
02, 17, 19, 37, 63, 06, 33, 55, 18, 58, 04, 05, 53, 54, 03, 21, 22, 23, 24, 52, 32, 65, 57	Naive Bayes	*0.965335*	*0.024813*	*0.592471*	*0.047632*	*0.964175*	*0.047607*	*0.592471*	*0.088131*
	Logistic Regression	*0.998537*	*0.000000*	*0.000000*	*0.000000*	*0.997078*	*0.000000*	*0.000000*	*0.000000*
	Tree	*0.999485*	*0.975962*	*0.664484*	*0.790652*	*0.998857*	*0.922727*	*0.664484*	*0.772598*
	SVM	*0.040525*	*0.000871*	*0.571195*	*0.001739*	*0.045547*	*0.001748*	*0.571195*	*0.003485*
	Random Forest	*0.999191*	*0.985765*	*0.453355*	*0.621076*	*0.998393*	*0.996390*	*0.451718*	*0.621622*
03, 21, 22, 23, 24, 52, 32, 65, 57	Naive Bayes	*0.996944*	*0.137405*	*0.206219*	*0.164921*	*0.995705*	*0.233766*	*0.206219*	*0.219130*
	Logistic Regression	*0.998537*	*0.000000*	*0.000000*	*0.000000*	*0.997078*	*0.000000*	*0.000000*	*0.000000*
	Tree	*0.999461*	*0.968447*	*0.653028*	*0.780059*	*0.998785*	*0.904762*	*0.653028*	*0.758555*
	SVM	*0.281356*	*0.001525*	*0.749591*	*0.003043*	*0.281582*	*0.003043*	*0.749591*	*0.006061*
	Random Forest	*0.999155*	*0.996154*	*0.423895*	*0.594719*	*0.998317*	*1.000000*	*0.423895*	*0.595402*
57	Naive Bayes	*0.998537*	*0.000000*	*0.000000*	*0.000000*	*0.997078*	*0.000000*	*0.000000*	*0.000000*
	Logistic Regression	*0.998537*	*0.000000*	*0.000000*	*0.000000*	*0.997078*	*0.000000*	*0.000000*	*0.000000*
	Tree	*0.998824*	*0.961538*	*0.204583*	*0.337382*	*0.997676*	*1.000000*	*0.204583*	*0.339674*
	SVM	*0.217976*	*0.001462*	*0.782324*	*0.002919*	*0.218296*	*0.002918*	*0.782324*	*0.005815*
	Random Forest	*0.999116*	*0.958333*	*0.414075*	*0.578286*	*0.998264*	*0.980620*	*0.414075*	*0.582278*

A.10. táblázat: Osztályozó eredmények az XSS adathalmazhoz

Features		Train Dataset				Test Dataset			
	Models	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
25, 27, 16, 40, 02, 05, 17, 19, 34, 53, 06, 18, 55, 21, 22, 23, 24, 51, 13, 39, 57, 37, 56, 33, 32, 03, 11, 52, 04, 54, 58	Naive Bayes	*0.88881*	*0.00444*	*0.90000*	*0.00885*	*0.88975*	*0.00892*	*0.90000*	*0.01767*
	Logistic Regression	*0.99945*	*0.00000*	*0.00000*	*0.00000*	*0.99890*	*0.00000*	*0.00000*	*0.00000*
	Tree	*0.99999*	*0.99559*	*0.98261*	*0.98906*	*0.99996*	*0.97835*	*0.98261*	*0.98048*
	SVM	*0.34396*	*0.00043*	*0.50870*	*0.00085*	*0.34348*	*0.00085*	*0.50870*	*0.00170*
	Random Forest	*0.99999*	*1.00000*	*0.97391*	*0.98678*	*0.99997*	*1.00000*	*0.97391*	*0.98678*
02, 05, 17, 19, 34, 53, 06, 18, 55, 21, 22, 23, 24, 51, 13, 39, 57, 37, 56, 33, 32, 03, 11, 52, 04, 54, 58	Naive Bayes	*0.87397*	*0.00370*	*0.84783*	*0.00736*	*0.87443*	*0.00740*	*0.84783*	*0.01466*
	Logistic Regression	*0.99945*	*0.00000*	*0.00000*	*0.00000*	*0.99890*	*0.00000*	*0.00000*	*0.00000*
	Tree	*0.99999*	*1.00000*	*0.98261*	*0.99123*	*0.99998*	*0.99559*	*0.98261*	*0.98906*
	SVM	*0.33647*	*0.00042*	*0.50870*	*0.00084*	*0.33644*	*0.00084*	*0.50870*	*0.00169*
	Random Forest	*0.99988*	*1.00000*	*0.77391*	*0.87255*	*0.99975*	*1.00000*	*0.77391*	*0.87255*
37, 56, 33, 32, 03, 11, 52, 04, 54, 58	Naive Bayes	*0.99870*	*0.00000*	*0.00000*	*0.00000*	*0.99812*	*0.00000*	*0.00000*	*0.00000*
	Logistic Regression	*0.99945*	*0.00000*	*0.00000*	*0.00000*	*0.99890*	*0.00000*	*0.00000*	*0.00000*
	Tree	*0.99998*	*1.00000*	*0.96957*	*0.98455*	*0.99996*	*0.99554*	*0.96957*	*0.98238*
	SVM	*0.37911*	*0.00046*	*0.51304*	*0.00091*	*0.37972*	*0.00091*	*0.51304*	*0.00182*
	Random Forest	*0.99999*	*1.00000*	*0.97391*	*0.98678*	*0.99997*	*0.99556*	*0.97391*	*0.98462*
03, 11, 52, 04, 54, 58	Naive Bayes	*0.99945*	*0.00000*	*0.00000*	*0.00000*	*0.99890*	*0.00000*	*0.00000*	*0.00000*
	Logistic Regression	*0.99945*	*0.00000*	*0.00000*	*0.00000*	*0.99890*	*0.00000*	*0.00000*	*0.00000*
	Tree	*0.99972*	*1.00000*	*0.48696*	*0.65497*	*0.99943*	*1.00000*	*0.48696*	*0.65497*
	SVM	*0.24808*	*0.00038*	*0.51304*	*0.00075*	*0.24765*	*0.00075*	*0.51304*	*0.00150*
	Random Forest	*0.99972*	*1.00000*	*0.48696*	*0.65497*	*0.99943*	*1.00000*	*0.48696*	*0.65497*
58	Naive Bayes	*0.99945*	*0.00000*	*0.00000*	*0.00000*	*0.99890*	*0.00000*	*0.00000*	*0.00000*
	Logistic Regression	*0.99945*	*0.00000*	*0.00000*	*0.00000*	*0.99890*	*0.00000*	*0.00000*	*0.00000*
	Tree	*0.99971*	*0.99083*	*0.46957*	*0.63717*	*0.99941*	*0.99083*	*0.46957*	*0.63717*
	SVM	*0.28919*	*0.00039*	*0.50870*	*0.00079*	*0.28816*	*0.00079*	*0.50870*	*0.00157*
	Random Forest	*0.99971*	*0.99083*	*0.46957*	*0.63717*	*0.99941*	*0.98182*	*0.46957*	*0.63529*

A.11. táblázat: Osztályozó eredmények az SQL adathalmazhoz

Features		Train Dataset				Test Dataset			
	Models	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
39, 43, 47, 10 ,15, 05, 26, 53, 56, 25, 02, 17, 19, 35, 16, 18, 27, 28, 34, 06, 23, 30, 55, 29, 21, 22, 24, 57, 37, 11, 14	Naive Bayes	*0.936794*	*0.001934*	*0.586207*	*0.003854*	*0.937682*	*0.003919*	*0.586207*	*0.007786*
	Logistic Regression	*0.999806*	*1.000000*	*0.068966*	*0.129032*	*0.999612*	*1.000000*	*0.068966*	*0.129032*
	Tree	*0.999993*	*1.000000*	*0.965517*	*0.982456*	*0.999986*	*1.000000*	*0.965517*	*0.982456*
	SVM	*0.999871*	*1.000000*	*0.379310*	*0.550000*	*0.999741*	*1.000000*	*0.379310*	*0.550000*
	Random Forest	*0.999998*	*1.000000*	*0.988506*	*0.994220*	*0.999995*	*1.000000*	*0.988506*	*0.994220*
05, 26, 53, 56, 25, 02, 17, 19, 35, 16, 18, 27, 28, 34, 06, 23, 30, 55, 29, 21, 22, 24, 57, 37, 11, 14	Naive Bayes	*0.915656*	*0.001449*	*0.586207*	*0.002891*	*0.916741*	*0.002934*	*0.586207*	*0.005839*
	Logistic Regression	*0.999794*	*1.000000*	*0.011494*	*0.022727*	*0.999588*	*1.000000*	*0.011494*	*0.022727*
	Tree	*0.999990*	*1.000000*	*0.954023*	*0.976471*	*0.999981*	*1.000000*	*0.954023*	*0.976471*
	SVM	*0.999871*	*1.000000*	*0.379310*	*0.550000*	*0.999741*	*1.000000*	*0.379310*	*0.550000*
	Random Forest	*0.999983*	*1.000000*	*0.919540*	*0.958084*	*0.999966*	*1.000000*	*0.919540*	*0.958084*
06, 23, 30, 55, 29, 21, 22, 24, 57, 37, 11, 14	Naive Bayes	*0.999779*	*0.407407*	*0.126437*	*0.192982*	*0.999607*	*0.647059*	*0.126437*	*0.211538*
	Logistic Regression	*0.999791*	*0.000000*	*0.000000*	*0.000000*	*0.999583*	*0.000000*	*0.000000*	*0.000000*
	Tree	*0.999926*	*0.982759*	*0.655172*	*0.786207*	*0.999842*	*0.950000*	*0.655172*	*0.775510*
	SVM	*0.999871*	*1.000000*	*0.379310*	*0.550000*	*0.999741*	*1.000000*	*0.379310*	*0.550000*
	Random Forest	*0.999902*	*1.000000*	*0.528736*	*0.691729*	*0.999803*	*1.000000*	*0.528736*	*0.691729*
57, 37, 11, 14	Naive Bayes	*0.999791*	*0.000000*	*0.000000*	*0.000000*	*0.999583*	*0.000000*	*0.000000*	*0.000000*
	Logistic Regression	*0.999791*	*0.000000*	*0.000000*	*0.000000*	*0.999583*	*0.000000*	*0.000000*	*0.000000*
	Tree	*0.999904*	*1.000000*	*0.540230*	*0.701493*	*0.999803*	*0.979167*	*0.540230*	*0.696296*
	SVM	*0.999871*	*1.000000*	*0.379310*	*0.550000*	*0.999741*	*1.000000*	*0.379310*	*0.550000*
	Random Forest	*0.999897*	*1.000000*	*0.505747*	*0.671756*	*0.999794*	*1.000000*	*0.505747*	*0.671756*
11, 14	Naive Bayes	*0.999791*	*0.000000*	*0.000000*	*0.000000*	*0.999583*	*0.000000*	*0.000000*	*0.000000*
	Logistic Regression	*0.999791*	*0.000000*	*0.000000*	*0.000000*	*0.999583*	*0.000000*	*0.000000*	*0.000000*
	Tree	*0.999897*	*0.978261*	*0.517241*	*0.676692*	*0.999789*	*0.957447*	*0.517241*	*0.671642*
	SVM	*0.999871*	*1.000000*	*0.379310*	*0.550000*	*0.999741*	*1.000000*	*0.379310*	*0.550000*
	Random Forest	*0.999897*	*0.978261*	*0.517241*	*0.676692*	*0.999789*	*0.957447*	*0.517241*	*0.671642*

A.12. táblázat Az FTP-adatkészlet súlyozott átlagának számítási eredményei

Features	Weighted Average							
	1	2	3	4	5	6	7	8
00 Dst Port	0.3600	0.1105	0.2272	0.2880	0.3887	0.3421	0.4141	0.3676
01 Protocol	0.1700	0.1172	0.1134	0.2069	0.1734	0.1218	0.1684	0.1168
02 Flow Duration	0.3500	0.3292	0.3776	0.0984	0.2590	0.4668	0.4836	0.2758
03 Tot Fwd Pkts	0.1500	0.0264	0.1225	0.1173	0.1472	0.1511	0.1657	0.1615
04 Tot Bwd Pkts	0.1300	0.0186	0.1062	0.1062	0.1318	0.1319	0.1456	0.1453
05 TotLen Fwd Pkts	0.2900	0.3202	0.3793	0.1001	0.1647	0.3725	0.3882	0.1804
06 TotLen Bwd Pkts	0.1000	0.0147	0.0625	0.0623	0.1111	0.1113	0.1258	0.1255
07 Fwd Pkt Len Max	0.3000	0.3220	0.3883	0.1100	0.1704	0.3773	0.3941	0.1855
08 Fwd Pkt Len Min	0.0600	0.0124	0.0459	0.0504	0.0552	0.0510	0.0641	0.0600
09 Fwd Pkt Len Mean	0.1500	0.0619	0.1169	0.0786	0.1389	0.1670	0.1842	0.1530
10 Fwd Pkt Len Std	0.0900	0.0673	0.0988	0.0465	0.0683	0.1070	0.1197	0.0791
11 Bwd Pkt Len Max	0.1900	0.1243	0.0874	0.1852	0.1992	0.1578	0.2140	0.1726
12 Bwd Pkt Len Min	0.0600	0.0312	0.0379	0.0604	0.0557	0.0574	0.0654	0.0672
13 Bwd Pkt Len Mean	0.1400	0.0641	0.0668	0.1118	0.1428	0.1337	0.1573	0.1482
14 Bwd Pkt Len Std	0.1200	0.0988	0.0486	0.1308	0.1218	0.0852	0.1326	0.0961
15 Flow Byts/s	0.1000	0.0137	0.0393	0.0394	0.1156	0.1156	0.1289	0.1289
16 Flow Pkts/s	0.3300	0.2189	0.0939	0.2673	0.2530	0.4195	0.2699	0.4363
17 Flow IAT Mean	0.3500	0.3293	0.3777	0.0985	0.2602	0.4680	0.4848	0.2770
18 Flow IAT Std	0.0600	0.0103	0.0269	0.0287	0.0683	0.0700	0.0802	0.0819
19 Flow IAT Max	0.3500	0.3292	0.3792	0.1000	0.2585	0.4663	0.4833	0.2755
20 Flow IAT Min	0.2900	0.3209	0.3653	0.0861	0.1793	0.3871	0.4026	0.1948
21 Fwd IAT Tot	0.2600	0.3154	0.3418	0.0625	0.1345	0.3423	0.3555	0.1477
22 Fwd IAT Mean	0.2600	0.3154	0.3417	0.0625	0.1345	0.3423	0.3555	0.1477
23 Fwd IAT Std	0.0500	0.0091	0.0247	0.0265	0.0565	0.0583	0.0680	0.0698
24 Fwd IAT Max	0.2600	0.3155	0.3422	0.0629	0.1347	0.3425	0.3557	0.1479
25 Fwd IAT Min	0.2600	0.3165	0.3544	0.0752	0.1388	0.3466	0.3607	0.1529
26 Bwd IAT Tot	0.0700	0.0367	0.0264	0.0535	0.0772	0.0595	0.0885	0.0709
27 Bwd IAT Mean	0.0500	0.0092	0.0237	0.0260	0.0542	0.0522	0.0656	0.0636
28 Bwd IAT Std	0.0400	0.0122	0.0206	0.0266	0.0483	0.0447	0.0593	0.0556
29 Bwd IAT Max	0.0600	0.0209	0.0263	0.0390	0.0652	0.0555	0.0766	0.0669
30 Bwd IAT Min	0.0500	0.0067	0.0272	0.0275	0.0479	0.0477	0.0590	0.0589
31 Fwd PSH Flags	0.0300	0.0203	0.0286	0.0441	0.0384	0.0266	0.0277	0.0160
32 Fwd Header Len	0.1900	0.0342	0.1423	0.1345	0.1906	0.1964	0.2145	0.2084
33 Bwd Header Len	0.2400	0.0333	0.1761	0.1761	0.2469	0.2470	0.2678	0.2675
34 Fwd Pkts/s	0.3200	0.1996	0.0946	0.2497	0.2568	0.4090	0.2739	0.4261
35 Bwd Pkts/s	0.3500	0.2301	0.1104	0.2915	0.2736	0.4477	0.2920	0.4661
36 Pkt Len Min	0.0600	0.0130	0.0459	0.0511	0.0553	0.0506	0.0643	0.0595
37 Pkt Len Max	0.3000	0.3238	0.3894	0.1125	0.1728	0.3783	0.3966	0.1865
38 Pkt Len Mean	0.1200	0.0293	0.0648	0.0768	0.1381	0.1323	0.1532	0.1474
39 Pkt Len Std	0.1200	0.0230	0.0667	0.0731	0.1318	0.1283	0.1469	0.1435
40 Pkt Len Var	0.2800	0.3192	0.3690	0.0900	0.1567	0.3643	0.3797	0.1716
41 FIN Flag Cnt	0.0100	0.0012	0.0094	0.0096	0.0094	0.0096	0.0010	0.0012
42 SYN Flag Cnt	0.0300	0.0203	0.0286	0.0441	0.0384	0.0266	0.0277	0.0160
43 RST Flag Cnt	0.0500	0.0277	0.0375	0.0588	0.0501	0.0350	0.0397	0.0247
44 PSH Flag Cnt	0.4200	0.3206	0.2503	0.5130	0.4367	0.2808	0.4230	0.2671
45 ACK Flag Cnt	0.1700	0.1308	0.1054	0.2125	0.1833	0.1084	0.1751	0.1003
46 URG Flag Cnt	0.0500	0.0416	0.0345	0.0688	0.0617	0.0323	0.0511	0.0218
47 ECE Flag Cnt	0.0500	0.0277	0.0375	0.0588	0.0501	0.0350	0.0397	0.0247
48 Down/Up Ratio	0.1500	0.0295	0.1642	0.1607	0.1453	0.1474	0.1430	0.1365
49 Pkt Size Avg	0.1300	0.0355	0.0659	0.0835	0.1443	0.1335	0.1595	0.1486
50 Fwd Seg Size Avg	0.1500	0.0618	0.1169	0.0785	0.1388	0.1670	0.1841	0.1530
51 Bwd Seg Size Avg	0.1400	0.0634	0.0668	0.1111	0.1422	0.1336	0.1566	0.1481
52 Subflow Fwd Pkts	0.1500	0.0265	0.1225	0.1174	0.1473	0.1511	0.1658	0.1615
53 Subflow Fwd Byts	0.2900	0.3202	0.3793	0.1001	0.1647	0.3725	0.3882	0.1804
54 Subflow Bwd Pkts	0.1300	0.0194	0.1063	0.1069	0.1326	0.1320	0.1463	0.1454
55 Subflow Bwd Byts	0.1000	0.0147	0.0625	0.0622	0.1110	0.1113	0.1258	0.1255
56 Init Fwd Win Byts	0.6200	0.4398	0.3060	0.6591	0.5396	0.5979	0.5644	0.6226
57 Init Bwd Win Byts	0.3100	0.0944	0.2055	0.2573	0.3241	0.2917	0.3461	0.3137
58 Fwd Act Data Pkts	0.2200	0.2387	0.3100	0.1048	0.1178	0.2703	0.2811	0.1264
59 Fwd Seg Size Min	0.8200	0.5149	0.5327	0.9335	0.7228	0.7406	0.7228	0.7406
60 Active Mean	0.0100	0.0021	0.0101	0.0102	0.0110	0.0109	0.0189	0.0187
61 Active Std	0.0100	0.0016	0.0085	0.0086	0.0078	0.0078	0.0148	0.0148
62 Active Max	0.0100	0.0027	0.0101	0.0108	0.0115	0.0109	0.0194	0.0188
63 Active Min	0.0100	0.0024	0.0108	0.0111	0.0116	0.0113	0.0194	0.0190
64 Idle Mean	0.0200	0.0048	0.0110	0.0135	0.0128	0.0152	0.0210	0.0234
65 Idle Std	0.0100	0.0019	0.0096	0.0098	0.0096	0.0098	0.0170	0.0172
66 Idle Max	0.0200	0.0049	0.0113	0.0138	0.0129	0.0154	0.0211	0.0236
67 Idle Min	0.0200	0.0047	0.0112	0.0135	0.0129	0.0151	0.0211	0.0233

A.13. táblázat Az SSH adatkészlet súlyozott átlagának számítási eredményei

Features	Weighted Average							
	1	2	3	4	5	6	7	8
00 Dst Port	0.3700	0.1201	0.2339	0.3028	0.4023	0.3473	0.4239	0.3689
01 Protocol	0.2100	0.1699	0.1250	0.2653	0.2314	0.1330	0.2217	0.1233
02 Flow Duration	0.3700	0.3317	0.3850	0.1058	0.2853	0.4931	0.5040	0.2962
03 Tot Fwd Pkts	0.1800	0.0286	0.1423	0.1393	0.1860	0.1883	0.2006	0.1980
04 Tot Bwd Pkts	0.1800	0.0259	0.1445	0.1443	0.1896	0.1897	0.2020	0.2018
05 TotLen Fwd Pkts	0.3000	0.2627	0.3434	0.1230	0.2118	0.3758	0.3884	0.2243
06 TotLen Bwd Pkts	0.1600	0.0227	0.1007	0.1003	0.1766	0.1770	0.1889	0.1886
07 Fwd Pkt Len Max	0.3100	0.2647	0.3583	0.1384	0.2203	0.3838	0.3977	0.2331
08 Fwd Pkt Len Min	0.0600	0.0146	0.0529	0.0588	0.0634	0.0576	0.0664	0.0607
09 Fwd Pkt Len Mean	0.1800	0.0482	0.1223	0.1017	0.1886	0.2036	0.2170	0.1994
10 Fwd Pkt Len Std	0.2700	0.2596	0.3372	0.1173	0.1783	0.3418	0.3532	0.1887
11 Bwd Pkt Len Max	0.3100	0.2187	0.1387	0.3139	0.2805	0.3067	0.2934	0.3195
12 Bwd Pkt Len Min	0.0900	0.0618	0.0477	0.0971	0.0883	0.0719	0.0920	0.0756
13 Bwd Pkt Len Mean	0.1800	0.0587	0.1016	0.1347	0.1944	0.1968	0.2062	0.2087
14 Bwd Pkt Len Std	0.2300	0.1256	0.1253	0.2194	0.1940	0.2385	0.2048	0.2494
15 Flow Byts/s	0.1300	0.0191	0.0564	0.0567	0.1642	0.1642	0.1723	0.1724
16 Flow Pkts/s	0.2500	0.0854	0.0871	0.1381	0.2618	0.3118	0.2726	0.3226
17 Flow IAT Mean	0.3700	0.3318	0.3847	0.1055	0.2856	0.4934	0.5042	0.2964
18 Flow IAT Std	0.2200	0.1982	0.2379	0.0704	0.1583	0.2829	0.2911	0.1664
19 Flow IAT Max	0.3700	0.3318	0.3857	0.1065	0.2854	0.4932	0.5042	0.2964
20 Flow IAT Min	0.2800	0.2065	0.2731	0.1056	0.2259	0.3505	0.3625	0.2378
21 Fwd IAT Tot	0.3000	0.3213	0.3612	0.0819	0.1883	0.3961	0.4045	0.1967
22 Fwd IAT Mean	0.3000	0.3212	0.3611	0.0819	0.1883	0.3961	0.4045	0.1967
23 Fwd IAT Std	0.2200	0.1979	0.2412	0.0737	0.1540	0.2786	0.2870	0.1624
24 Fwd IAT Max	0.3000	0.3212	0.3605	0.0813	0.1880	0.3958	0.4041	0.1963
25 Fwd IAT Min	0.1700	0.1920	0.2199	0.0524	0.1032	0.2278	0.2338	0.1092
26 Bwd IAT Tot	0.1700	0.0864	0.0676	0.1309	0.2000	0.1462	0.2085	0.1547
27 Bwd IAT Mean	0.1200	0.0247	0.0613	0.0691	0.1428	0.1354	0.1513	0.1439
28 Bwd IAT Std	0.1200	0.0270	0.0627	0.0728	0.1408	0.1332	0.1494	0.1418
29 Bwd IAT Max	0.1300	0.0320	0.0633	0.0775	0.1493	0.1382	0.1578	0.1468
30 Bwd IAT Min	0.1100	0.0162	0.0737	0.0740	0.1251	0.1249	0.1341	0.1340
31 Fwd PSH Flags	0.0400	0.0239	0.0353	0.0536	0.0478	0.0333	0.0313	0.0169
32 Fwd Header Len	0.3100	0.0615	0.2296	0.2115	0.3226	0.3360	0.3571	0.3434
33 Bwd Header Len	0.2800	0.0412	0.2037	0.2049	0.3019	0.3007	0.3218	0.3202
34 Fwd Pkts/s	0.2100	0.0316	0.0812	0.0840	0.2562	0.2548	0.2669	0.2655
35 Bwd Pkts/s	0.2200	0.0422	0.0914	0.1027	0.2621	0.2715	0.2740	0.2834
36 Pkt Len Min	0.0700	0.0211	0.0535	0.0652	0.0697	0.0583	0.0727	0.0614
37 Pkt Len Max	0.3100	0.2658	0.3584	0.1396	0.2204	0.3828	0.3977	0.2320
38 Pkt Len Mean	0.1700	0.0401	0.0887	0.1053	0.1935	0.1829	0.2047	0.1942
39 Pkt Len Std	0.1600	0.0268	0.0913	0.0958	0.1823	0.1798	0.1938	0.1913
40 Pkt Len Var	0.2900	0.2612	0.3300	0.1098	0.2024	0.3663	0.3779	0.2138
41 FIN Flag Cnt	0.0200	0.0110	0.0172	0.0257	0.0254	0.0175	0.0107	0.0028
42 SYN Flag Cnt	0.0500	0.0330	0.0362	0.0627	0.0569	0.0342	0.0404	0.0178
43 RST Flag Cnt	0.0700	0.0534	0.0460	0.0900	0.0809	0.0431	0.0649	0.0270
44 PSH Flag Cnt	0.0000	0.0021	0.0014	0.0031	0.0016	0.0032	0.0025	0.0042
45 ACK Flag Cnt	0.0300	0.0130	0.0261	0.0349	0.0232	0.0320	0.0217	0.0305
46 URG Flag Cnt	0.2400	0.0907	0.2289	0.2865	0.1954	0.2530	0.1550	0.2126
47 ECE Flag Cnt	0.0600	0.0443	0.0451	0.0809	0.0718	0.0422	0.0558	0.0261
48 Down/Up Ratio	0.0300	0.0185	0.0419	0.0285	0.0265	0.0365	0.0359	0.0257
49 Pkt Size Avg	0.1700	0.0378	0.0884	0.1030	0.1900	0.1819	0.2012	0.1931
50 Fwd Seg Size Avg	0.1800	0.0485	0.1224	0.1021	0.1889	0.2036	0.2173	0.1994
51 Bwd Seg Size Avg	0.1900	0.0646	0.1022	0.1406	0.2003	0.1974	0.2121	0.2092
52 Subflow Fwd Pkts	0.1800	0.0286	0.1423	0.1393	0.1861	0.1883	0.2006	0.1980
53 Subflow Fwd Byts	0.3000	0.2627	0.3434	0.1231	0.2118	0.3758	0.3884	0.2243
54 Subflow Bwd Pkts	0.1800	0.0259	0.1445	0.1444	0.1896	0.1897	0.2020	0.2018
55 Subflow Bwd Byts	0.1600	0.0230	0.1008	0.1007	0.1770	0.1770	0.1893	0.1886
56 Init Fwd Win Byts	0.4500	0.2070	0.2752	0.4194	0.4120	0.4654	0.4339	0.4873
57 Init Bwd Win Byts	0.3900	0.1320	0.2558	0.3335	0.4216	0.3630	0.4435	0.3849
58 Fwd Act Data Pkts	0.3000	0.2358	0.3727	0.1805	0.2110	0.3540	0.3641	0.2204
59 Fwd Seg Size Min	0.8300	0.5343	0.5343	0.9529	0.7380	0.7380	0.7380	0.7380
60 Active Mean	0.0200	0.0030	0.0172	0.0176	0.0182	0.0182	0.0197	0.0197
61 Active Std	0.0200	0.0027	0.0157	0.0161	0.0153	0.0151	0.0159	0.0157
62 Active Max	0.0200	0.0039	0.0173	0.0185	0.0190	0.0184	0.0205	0.0199
63 Active Min	0.0200	0.0035	0.0179	0.0186	0.0190	0.0186	0.0204	0.0200
64 Idle Mean	0.1900	0.3058	0.3209	0.0417	0.0497	0.2575	0.2594	0.0516
65 Idle Std	0.1800	0.3054	0.3197	0.0404	0.0468	0.2546	0.2556	0.0478
66 Idle Max	0.1900	0.3058	0.3211	0.0419	0.0499	0.2577	0.2595	0.0517
67 Idle Min	0.1900	0.3059	0.3210	0.0418	0.0499	0.2576	0.2595	0.0516

A.14. táblázat A WEB adatkészlet súlyozott átlagának számítási eredményei

Features	Weighted Average							
	1	2	3	4	5	6	7	8
00 Dst Port	0.3300	0.1934	0.2434	0.3905	0.3648	0.2184	0.3651	0.2187
01 Protocol	0.0100	0.0016	0.0125	0.0126	0.0101	0.0102	0.0101	0.0101
02 Flow Duration	0.4600	0.3434	0.5163	0.2376	0.3542	0.5615	0.5619	0.3546
03 Tot Fwd Pkts	0.5100	0.3504	0.7029	0.4241	0.3313	0.5388	0.5390	0.3314
04 Tot Bwd Pkts	0.4900	0.3479	0.6854	0.4071	0.3199	0.5270	0.5272	0.3200
05 TotLen Fwd Pkts	0.4900	0.2436	0.3916	0.5662	0.3538	0.5283	0.3543	0.5288
06 TotLen Bwd Pkts	0.4700	0.3450	0.6122	0.3332	0.3168	0.5244	0.5249	0.3171
07 Fwd Pkt Len Max	0.4200	0.0761	0.4385	0.4556	0.3976	0.3848	0.3982	0.3853
08 Fwd Pkt Len Min	0.2800	0.1052	0.2804	0.3463	0.2270	0.2507	0.2268	0.2506
09 Fwd Pkt Len Mean	0.4300	0.1397	0.3884	0.4675	0.3857	0.4191	0.3863	0.4197
10 Fwd Pkt Len Std	0.3700	0.0745	0.3927	0.4153	0.3464	0.3292	0.3468	0.3296
11 Bwd Pkt Len Max	0.4300	0.1591	0.3856	0.4850	0.4059	0.3602	0.4063	0.3606
12 Bwd Pkt Len Min	0.0800	0.0463	0.0703	0.1048	0.0878	0.0535	0.0878	0.0535
13 Bwd Pkt Len Mean	0.3000	0.0694	0.2841	0.3122	0.2873	0.2682	0.2877	0.2686
14 Bwd Pkt Len Std	0.4300	0.0823	0.4786	0.5006	0.3865	0.3778	0.3867	0.3781
15 Flow Byts/s	0.4300	0.3392	0.5164	0.2374	0.3047	0.5123	0.5127	0.3050
16 Flow Pkts/s	0.3600	0.1678	0.3396	0.2227	0.3360	0.4230	0.4234	0.3364
17 Flow IAT Mean	0.4600	0.3435	0.5148	0.2356	0.3499	0.5577	0.5581	0.3503
18 Flow IAT Std	0.4800	0.3458	0.5839	0.3053	0.3456	0.5529	0.5534	0.3461
19 Flow IAT Max	0.4600	0.3444	0.5213	0.2421	0.3553	0.5631	0.5635	0.3557
20 Flow IAT Min	0.3600	0.3148	0.4511	0.1867	0.2379	0.4347	0.4350	0.2382
21 Fwd IAT Tot	0.5100	0.3500	0.5697	0.2910	0.3968	0.6041	0.6046	0.3973
22 Fwd IAT Mean	0.5100	0.3507	0.5710	0.2918	0.3967	0.6045	0.6050	0.3972
23 Fwd IAT Std	0.5100	0.3501	0.6288	0.3502	0.3675	0.5749	0.5754	0.3681
24 Fwd IAT Max	0.5100	0.3503	0.5669	0.2877	0.3947	0.6025	0.6030	0.3952
25 Fwd IAT Min	0.4400	0.3268	0.5009	0.2365	0.3394	0.5362	0.5366	0.3398
26 Bwd IAT Tot	0.3400	0.1777	0.2476	0.3778	0.3575	0.2611	0.3579	0.2615
27 Bwd IAT Mean	0.4100	0.3416	0.5339	0.2627	0.2664	0.4665	0.4735	0.2602
28 Bwd IAT Std	0.2800	0.0822	0.2831	0.2709	0.2655	0.2708	0.2864	0.2507
29 Bwd IAT Max	0.2900	0.0944	0.2580	0.2950	0.2872	0.2480	0.2941	0.2419
30 Bwd IAT Min	0.3400	0.3273	0.4773	0.1988	0.1965	0.4037	0.4044	0.1963
31 Fwd PSH Flags	0.0600	0.0079	0.0710	0.0710	0.0482	0.0482	0.0462	0.0463
32 Fwd Header Len	0.5400	0.3549	0.7297	0.4509	0.3653	0.5728	0.5731	0.3656
33 Bwd Header Len	0.4700	0.3445	0.6445	0.3662	0.3040	0.5111	0.5114	0.3042
34 Fwd Pkts/s	0.4000	0.1962	0.3509	0.2576	0.3776	0.4410	0.4651	0.3543
35 Bwd Pkts/s	0.3800	0.3316	0.4750	0.2003	0.2513	0.4552	0.4575	0.2496
36 Pkt Len Min	0.2800	0.0987	0.2814	0.3414	0.2200	0.2512	0.2199	0.2511
37 Pkt Len Max	0.4600	0.1129	0.4561	0.5046	0.4188	0.4270	0.4193	0.4276
38 Pkt Len Mean	0.3800	0.1175	0.3329	0.3969	0.3618	0.3654	0.3623	0.3660
39 Pkt Len Std	0.3200	0.0903	0.2942	0.3395	0.3195	0.2861	0.3199	0.2866
40 Pkt Len Var	0.3000	0.0556	0.2902	0.3043	0.2854	0.2815	0.2858	0.2820
41 FIN Flag Cnt	0.0700	0.0544	0.0790	0.0779	0.0598	0.0550	0.0700	0.0311
42 SYN Flag Cnt	0.0600	0.0079	0.0710	0.0710	0.0482	0.0482	0.0462	0.0463
43 RST Flag Cnt	0.2400	0.0352	0.3280	0.3295	0.1901	0.1915	0.1865	0.1880
44 PSH Flag Cnt	0.0000	0.0010	0.0049	0.0045	0.0025	0.0029	0.0028	0.0025
45 ACK Flag Cnt	0.1600	0.2948	0.2948	0.0232	0.0295	0.2316	0.2316	0.0295
46 URG Flag Cnt	0.0700	0.0101	0.0907	0.0908	0.0614	0.0615	0.0592	0.0593
47 ECE Flag Cnt	0.2400	0.0352	0.3280	0.3295	0.1901	0.1915	0.1865	0.1880
48 Down/Up Ratio	0.2800	0.3183	0.4445	0.1652	0.1204	0.3282	0.3276	0.1198
49 Pkt Size Avg	0.3700	0.0968	0.3355	0.3804	0.3586	0.3504	0.3591	0.3509
50 Fwd Seg Size Avg	0.4300	0.1367	0.3881	0.4646	0.3828	0.4189	0.3833	0.4194
51 Bwd Seg Size Avg	0.2900	0.0671	0.2839	0.3099	0.2850	0.2680	0.2854	0.2684
52 Subflow Fwd Pkts	0.5100	0.3504	0.7029	0.4241	0.3313	0.5388	0.5390	0.3314
53 Subflow Fwd Byts	0.4900	0.2436	0.3916	0.5661	0.3538	0.5283	0.3543	0.5288
54 Subflow Bwd Pkts	0.4900	0.3479	0.6854	0.4071	0.3199	0.5270	0.5272	0.3200
55 Subflow Bwd Byts	0.4700	0.3450	0.6122	0.3331	0.3167	0.5244	0.5248	0.3171
56 Init Fwd Win Byts	0.3800	0.1401	0.3656	0.4528	0.3722	0.2853	0.3724	0.2855
57 Init Bwd Win Byts	0.5600	0.2875	0.4655	0.6751	0.5848	0.3758	0.5851	0.3761
58 Fwd Act Data Pkts	0.4800	0.2757	0.3995	0.6090	0.2876	0.4968	0.2874	0.4966
59 Fwd Seg Size Min	0.0100	0.0019	0.0132	0.0132	0.0111	0.0113	0.0112	0.0112
60 Active Mean	0.4400	0.3424	0.6404	0.3619	0.2683	0.4753	0.4757	0.2672
61 Active Std	0.1900	0.3074	0.3315	0.0533	0.0520	0.2588	0.2597	0.0509
62 Active Max	0.4400	0.3427	0.6410	0.3629	0.2689	0.4757	0.4763	0.2676
63 Active Min	0.4600	0.3441	0.6637	0.3846	0.2805	0.4882	0.4878	0.2799
64 Idle Mean	0.3800	0.3322	0.5451	0.2665	0.2215	0.4288	0.4290	0.2216
65 Idle Std	0.5400	0.3540	0.7720	0.4934	0.3384	0.5458	0.5451	0.3378
66 Idle Max	0.3700	0.3313	0.5307	0.2515	0.2133	0.4211	0.4212	0.2134
67 Idle Min	0.3700	0.3313	0.5302	0.2510	0.2130	0.4208	0.4210	0.2131

A.15. táblázat Az XSS adatkészlet súlyozott átlagának számítási eredményei

Features	Weighted Average							
	1	2	3	4	5	6	7	8
00 Dst Port	*0.2700*	*0.2472*	*0.1160*	*0.3253*	*0.3390*	*0.1297*	*0.3391*	*0.1298*
01 Protocol	*0.0800*	*0.0297*	*0.0786*	*0.0973*	*0.0764*	*0.0579*	*0.0763*	*0.0578*
02 Flow Duration	*0.4000*	*0.3351*	*0.4242*	*0.1451*	*0.3079*	*0.5156*	*0.5157*	*0.3081*
03 Tot Fwd Pkts	*0.5100*	*0.3511*	*0.6696*	*0.3906*	*0.3531*	*0.5607*	*0.5609*	*0.3532*
04 Tot Bwd Pkts	*0.5200*	*0.3520*	*0.6760*	*0.3968*	*0.3580*	*0.5658*	*0.5659*	*0.3581*
05 TotLen Fwd Pkts	*0.4000*	*0.3309*	*0.5134*	*0.2395*	*0.2734*	*0.4772*	*0.4774*	*0.2734*
06 TotLen Bwd Pkts	*0.4100*	*0.3366*	*0.5171*	*0.2380*	*0.2755*	*0.4833*	*0.4834*	*0.2756*
07 Fwd Pkt Len Max	*0.2600*	*0.0489*	*0.2378*	*0.2498*	*0.2681*	*0.2565*	*0.2683*	*0.2566*
08 Fwd Pkt Len Min	*0.0600*	*0.0239*	*0.0447*	*0.0609*	*0.0588*	*0.0427*	*0.0588*	*0.0427*
09 Fwd Pkt Len Mean	*0.2600*	*0.0645*	*0.2101*	*0.2388*	*0.2659*	*0.2462*	*0.2660*	*0.2464*
10 Fwd Pkt Len Std	*0.2900*	*0.0646*	*0.2669*	*0.2911*	*0.2882*	*0.2697*	*0.2883*	*0.2699*
11 Bwd Pkt Len Max	*0.5100*	*0.3680*	*0.2981*	*0.5945*	*0.3933*	*0.5155*	*0.3935*	*0.5156*
12 Bwd Pkt Len Min	*0.0500*	*0.0339*	*0.0330*	*0.0594*	*0.0619*	*0.0355*	*0.0619*	*0.0355*
13 Bwd Pkt Len Mean	*0.4400*	*0.3168*	*0.2372*	*0.4929*	*0.3145*	*0.4774*	*0.3146*	*0.4775*
14 Bwd Pkt Len Std	*0.3400*	*0.0931*	*0.3297*	*0.3750*	*0.3301*	*0.2980*	*0.3302*	*0.2981*
15 Flow Byts/s	*0.3300*	*0.3271*	*0.4111*	*0.1331*	*0.2185*	*0.4251*	*0.4263*	*0.2176*
16 Flow Pkts/s	*0.3900*	*0.3048*	*0.3992*	*0.1495*	*0.3140*	*0.4998*	*0.5001*	*0.3140*
17 Flow IAT Mean	*0.4000*	*0.3297*	*0.4223*	*0.1483*	*0.3067*	*0.5106*	*0.5107*	*0.3068*
18 Flow IAT Std	*0.4100*	*0.3364*	*0.4745*	*0.1955*	*0.2966*	*0.5043*	*0.5044*	*0.2968*
19 Flow IAT Max	*0.4000*	*0.3349*	*0.4284*	*0.1493*	*0.3036*	*0.5112*	*0.5114*	*0.3037*
20 Flow IAT Min	*0.2800*	*0.3184*	*0.3744*	*0.0953*	*0.1579*	*0.3655*	*0.3656*	*0.1579*
21 Fwd IAT Tot	*0.4300*	*0.3391*	*0.4571*	*0.1780*	*0.3331*	*0.5408*	*0.5410*	*0.3333*
22 Fwd IAT Mean	*0.4300*	*0.3344*	*0.4579*	*0.1839*	*0.3377*	*0.5416*	*0.5418*	*0.3379*
23 Fwd IAT Std	*0.4300*	*0.3397*	*0.4985*	*0.2194*	*0.3196*	*0.5273*	*0.5275*	*0.3198*
24 Fwd IAT Max	*0.4300*	*0.3393*	*0.4598*	*0.1808*	*0.3343*	*0.5420*	*0.5422*	*0.3345*
25 Fwd IAT Min	*0.3600*	*0.3301*	*0.4127*	*0.1336*	*0.2608*	*0.4685*	*0.4686*	*0.2609*
26 Bwd IAT Tot	*0.2200*	*0.0664*	*0.1612*	*0.1965*	*0.2315*	*0.2237*	*0.2316*	*0.2238*
27 Bwd IAT Mean	*0.3700*	*0.3391*	*0.4610*	*0.1900*	*0.2452*	*0.4449*	*0.4532*	*0.2373*
28 Bwd IAT Std	*0.3400*	*0.2815*	*0.4113*	*0.2006*	*0.2380*	*0.3919*	*0.4037*	*0.2265*
29 Bwd IAT Max	*0.3400*	*0.2671*	*0.3671*	*0.2192*	*0.2693*	*0.3702*	*0.4062*	*0.2336*
30 Bwd IAT Min	*0.2500*	*0.3155*	*0.3851*	*0.1061*	*0.1179*	*0.3254*	*0.3257*	*0.1177*
31 Fwd PSH Flags	*0.0400*	*0.0057*	*0.0417*	*0.0409*	*0.0305*	*0.0311*	*0.0306*	*0.0300*
32 Fwd Header Len	*0.4900*	*0.3482*	*0.6259*	*0.3467*	*0.3422*	*0.5499*	*0.5501*	*0.3423*
33 Bwd Header Len	*0.4800*	*0.3463*	*0.6198*	*0.3406*	*0.3259*	*0.5337*	*0.5338*	*0.3260*
34 Fwd Pkts/s	*0.4000*	*0.3226*	*0.4025*	*0.1689*	*0.3279*	*0.4976*	*0.5141*	*0.3117*
35 Bwd Pkts/s	*0.3300*	*0.3275*	*0.4154*	*0.1373*	*0.2206*	*0.4272*	*0.4285*	*0.2196*
36 Pkt Len Min	*0.0500*	*0.0203*	*0.0444*	*0.0573*	*0.0546*	*0.0417*	*0.0546*	*0.0418*
37 Pkt Len Max	*0.4500*	*0.3008*	*0.2762*	*0.5137*	*0.3242*	*0.4910*	*0.3243*	*0.4912*
38 Pkt Len Mean	*0.2500*	*0.0882*	*0.1831*	*0.2358*	*0.2670*	*0.2425*	*0.2672*	*0.2426*
39 Pkt Len Std	*0.4400*	*0.3319*	*0.2201*	*0.4911*	*0.3303*	*0.4560*	*0.3304*	*0.4561*
40 Pkt Len Var	*0.3900*	*0.2730*	*0.2139*	*0.4319*	*0.2703*	*0.4509*	*0.2704*	*0.4510*
41 FIN Flag Cnt	*0.0600*	*0.0895*	*0.1096*	*0.0292*	*0.0220*	*0.0818*	*0.0799*	*0.0201*
42 SYN Flag Cnt	*0.0800*	*0.0643*	*0.0476*	*0.0995*	*0.0891*	*0.0370*	*0.0892*	*0.0358*
43 RST Flag Cnt	*0.1800*	*0.0256*	*0.2292*	*0.2296*	*0.1514*	*0.1517*	*0.1504*	*0.1507*
44 PSH Flag Cnt	*0.0100*	*0.0009*	*0.0009*	*0.0008*	*0.0075*	*0.0076*	*0.0076*	*0.0075*
45 ACK Flag Cnt	*0.0200*	*0.0027*	*0.0144*	*0.0144*	*0.0213*	*0.0213*	*0.0213*	*0.0213*
46 URG Flag Cnt	*0.0900*	*0.0837*	*0.0510*	*0.1210*	*0.1108*	*0.0408*	*0.1106*	*0.0400*
47 ECE Flag Cnt	*0.1800*	*0.0256*	*0.2292*	*0.2296*	*0.1514*	*0.1517*	*0.1504*	*0.1507*
48 Down/Up Ratio	*0.2900*	*0.3208*	*0.4426*	*0.1638*	*0.1428*	*0.3501*	*0.3503*	*0.1421*
49 Pkt Size Avg	*0.2500*	*0.0795*	*0.1839*	*0.2286*	*0.2619*	*0.2396*	*0.2621*	*0.2398*
50 Fwd Seg Size Avg	*0.2500*	*0.0612*	*0.2098*	*0.2356*	*0.2626*	*0.2459*	*0.2628*	*0.2461*
51 Bwd Seg Size Avg	*0.4300*	*0.3115*	*0.2367*	*0.4875*	*0.3091*	*0.4768*	*0.3093*	*0.4770*
52 Subflow Fwd Pkts	*0.5100*	*0.3512*	*0.6696*	*0.3906*	*0.3531*	*0.5607*	*0.5609*	*0.3532*
53 Subflow Fwd Byts	*0.4000*	*0.3310*	*0.5134*	*0.2396*	*0.2734*	*0.4772*	*0.4775*	*0.2734*
54 Subflow Bwd Pkts	*0.5200*	*0.3520*	*0.6760*	*0.3968*	*0.3580*	*0.5658*	*0.5659*	*0.3581*
55 Subflow Bwd Byts	*0.4100*	*0.3366*	*0.5171*	*0.2379*	*0.2755*	*0.4833*	*0.4834*	*0.2756*
56 Init Fwd Win Byts	*0.4600*	*0.2414*	*0.3641*	*0.5405*	*0.4974*	*0.3210*	*0.4976*	*0.3211*
57 Init Bwd Win Byts	*0.4400*	*0.2357*	*0.3164*	*0.4910*	*0.4836*	*0.3091*	*0.4837*	*0.3092*
58 Fwd Act Data Pkts	*0.5900*	*0.3567*	*0.7751*	*0.5014*	*0.4117*	*0.6153*	*0.6156*	*0.4114*
59 Fwd Seg Size Min	*0.0600*	*0.0090*	*0.0749*	*0.0750*	*0.0561*	*0.0562*	*0.0560*	*0.0561*
60 Active Mean	*0.2000*	*0.3086*	*0.3322*	*0.0535*	*0.0695*	*0.2768*	*0.2773*	*0.0690*
61 Active Std	*0.1900*	*0.3058*	*0.3123*	*0.0332*	*0.0529*	*0.2605*	*0.2607*	*0.0527*
62 Active Max	*0.2000*	*0.3087*	*0.3323*	*0.0537*	*0.0696*	*0.2768*	*0.2774*	*0.0690*
63 Active Min	*0.2000*	*0.3085*	*0.3346*	*0.0555*	*0.0704*	*0.2780*	*0.2781*	*0.0703*
64 Idle Mean	*0.1900*	*0.3064*	*0.3231*	*0.0440*	*0.0569*	*0.2646*	*0.2647*	*0.0570*
65 Idle Std	*0.2000*	*0.3079*	*0.3345*	*0.0554*	*0.0678*	*0.2755*	*0.2755*	*0.0678*
66 Idle Max	*0.1900*	*0.3064*	*0.3235*	*0.0445*	*0.0572*	*0.2649*	*0.2649*	*0.0572*
67 Idle Min	*0.1900*	*0.3063*	*0.3223*	*0.0433*	*0.0566*	*0.2642*	*0.2643*	*0.0566*

A.16. táblázat Az SQL adatkészlet súlyozott átlagának számítási eredményei

Features	Weighted Average							
	1	2	3	4	5	6	7	8
00 Dst Port	0.2700	0.2295	0.1403	0.3208	0.3310	0.1492	0.3350	0.1452
01 Protocol	0.2700	0.2468	0.1489	0.3582	0.3187	0.1094	0.3186	0.1093
02 Flow Duration	0.4200	0.3381	0.4723	0.1931	0.3146	0.5223	0.5224	0.3146
03 Tot Fwd Pkts	0.3000	0.3217	0.4553	0.1761	0.1505	0.3582	0.3583	0.1505
04 Tot Bwd Pkts	0.2900	0.3205	0.4263	0.1471	0.1517	0.3595	0.3595	0.1517
05 TotLen Fwd Pkts	0.4000	0.3362	0.5031	0.2240	0.2791	0.4868	0.4869	0.2792
06 TotLen Bwd Pkts	0.4500	0.3433	0.5582	0.2791	0.3248	0.5325	0.5326	0.3248
07 Fwd Pkt Len Max	0.2500	0.0404	0.2170	0.2220	0.2640	0.2593	0.2640	0.2593
08 Fwd Pkt Len Min	0.2200	0.3181	0.3527	0.0816	0.0792	0.2789	0.2870	0.0711
09 Fwd Pkt Len Mean	0.2800	0.0487	0.2444	0.2540	0.2875	0.2784	0.2875	0.2785
10 Fwd Pkt Len Std	0.3800	0.0643	0.3833	0.3939	0.3713	0.3616	0.3713	0.3616
11 Bwd Pkt Len Max	0.5500	0.3100	0.3876	0.6207	0.4049	0.5904	0.4049	0.5905
12 Bwd Pkt Len Min	0.0700	0.0568	0.0648	0.0674	0.0601	0.0518	0.0768	0.0351
13 Bwd Pkt Len Mean	0.2900	0.0521	0.2610	0.2721	0.2989	0.2883	0.2990	0.2884
14 Bwd Pkt Len Std	0.5800	0.2626	0.5001	0.6812	0.4347	0.5962	0.4347	0.5962
15 Flow Byts/s	0.3800	0.3325	0.4569	0.1778	0.2633	0.4710	0.4711	0.2633
16 Flow Pkts/s	0.4300	0.3489	0.4746	0.2057	0.3348	0.5325	0.5421	0.3253
17 Flow IAT Mean	0.4200	0.3388	0.4725	0.1933	0.3212	0.5290	0.5290	0.3213
18 Flow IAT Std	0.4400	0.3361	0.5206	0.2459	0.3173	0.5217	0.5217	0.3173
19 Flow IAT Max	0.4200	0.3339	0.4732	0.1986	0.3178	0.5222	0.5223	0.3178
20 Flow IAT Min	0.2800	0.3193	0.3896	0.1105	0.1578	0.3656	0.3656	0.1578
21 Fwd IAT Tot	0.4700	0.3451	0.5176	0.2385	0.3653	0.5730	0.5731	0.3653
22 Fwd IAT Mean	0.4700	0.3457	0.5213	0.2421	0.3701	0.5779	0.5779	0.3702
23 Fwd IAT Std	0.4500	0.3374	0.5444	0.2697	0.3194	0.5238	0.5239	0.3195
24 Fwd IAT Max	0.4700	0.3405	0.5142	0.2395	0.3666	0.5710	0.5711	0.3666
25 Fwd IAT Min	0.4100	0.3362	0.4630	0.1838	0.2996	0.5073	0.5074	0.2996
26 Bwd IAT Tot	0.4000	0.2480	0.3952	0.3018	0.3497	0.4065	0.4561	0.3002
27 Bwd IAT Mean	0.4400	0.3437	0.5379	0.2614	0.3124	0.5175	0.5202	0.3098
28 Bwd IAT Std	0.4400	0.3400	0.5532	0.2779	0.3115	0.5161	0.5173	0.3104
29 Bwd IAT Max	0.4600	0.3593	0.5434	0.2923	0.3386	0.5198	0.5418	0.3168
30 Bwd IAT Min	0.4500	0.3434	0.5745	0.2955	0.3160	0.5236	0.5238	0.3158
31 Fwd PSH Flags	0.1100	0.1062	0.0820	0.1352	0.1206	0.0627	0.1342	0.0485
32 Fwd Header Len	0.3100	0.3230	0.4590	0.1798	0.1620	0.3698	0.3698	0.1620
33 Bwd Header Len	0.3000	0.3224	0.4411	0.1621	0.1621	0.3696	0.3699	0.1618
34 Fwd Pkts/s	0.4400	0.3447	0.4804	0.2068	0.3363	0.5387	0.5436	0.3315
35 Bwd Pkts/s	0.4200	0.3389	0.4867	0.2082	0.3073	0.5143	0.5151	0.3066
36 Pkt Len Min	0.2200	0.3172	0.3522	0.0811	0.0781	0.2778	0.2856	0.0703
37 Pkt Len Max	0.5100	0.2836	0.3622	0.5742	0.3717	0.5566	0.3717	0.5566
38 Pkt Len Mean	0.2800	0.0532	0.2399	0.2536	0.2945	0.2810	0.2946	0.2811
39 Pkt Len Std	0.3500	0.1155	0.2822	0.3490	0.3237	0.3490	0.3237	0.3490
40 Pkt Len Var	0.3300	0.0958	0.2800	0.3291	0.3048	0.3459	0.3049	0.3460
41 FIN Flag Cnt	0.1200	0.1715	0.1910	0.0569	0.0470	0.1442	0.1534	0.0358
42 SYN Flag Cnt	0.1000	0.0834	0.0797	0.1124	0.0978	0.0604	0.1114	0.0462
43 RST Flag Cnt	0.3700	0.0514	0.4560	0.4563	0.3086	0.3089	0.3077	0.3079
44 PSH Flag Cnt	0.0400	0.0057	0.0377	0.0377	0.0405	0.0405	0.0405	0.0405
45 ACK Flag Cnt	0.0200	0.0409	0.0421	0.0045	0.0048	0.0328	0.0328	0.0048
46 URG Flag Cnt	0.0800	0.1505	0.1505	0.0118	0.0150	0.1182	0.1182	0.0150
47 ECE Flag Cnt	0.3700	0.0514	0.4560	0.4563	0.3086	0.3089	0.3077	0.3079
48 Down/Up Ratio	0.1800	0.3053	0.3243	0.0452	0.0422	0.2498	0.2499	0.0420
49 Pkt Size Avg	0.2800	0.0477	0.2361	0.2452	0.2873	0.2789	0.2874	0.2790
50 Fwd Seg Size Avg	0.2800	0.0498	0.2446	0.2552	0.2886	0.2785	0.2887	0.2786
51 Bwd Seg Size Avg	0.3000	0.0558	0.2614	0.2759	0.3027	0.2887	0.3027	0.2888
52 Subflow Fwd Pkts	0.3000	0.3217	0.4553	0.1761	0.1505	0.3582	0.3583	0.1505
53 Subflow Fwd Byts	0.4000	0.3362	0.5031	0.2240	0.2791	0.4868	0.4869	0.2792
54 Subflow Bwd Pkts	0.2900	0.3205	0.4263	0.1471	0.1517	0.3595	0.3595	0.1517
55 Subflow Bwd Byts	0.4500	0.3432	0.5582	0.2790	0.3247	0.5325	0.5326	0.3248
56 Init Fwd Win Byts	0.4000	0.1406	0.3701	0.4519	0.4079	0.3259	0.4088	0.3250
57 Init Bwd Win Byts	0.5000	0.1861	0.4377	0.5540	0.5144	0.3983	0.5144	0.3983
58 Fwd Act Data Pkts	0.2300	0.3124	0.3639	0.0849	0.0992	0.3068	0.3069	0.0992
59 Fwd Seg Size Min	0.1400	0.0729	0.1284	0.1814	0.1446	0.0917	0.1445	0.0916
60 Active Mean	0.2000	0.3072	0.3202	0.0412	0.0642	0.2718	0.2720	0.0640
61 Active Std	0.1900	0.3068	0.3194	0.0405	0.0585	0.2660	0.2663	0.0582
62 Active Max	0.2000	0.3079	0.3203	0.0419	0.0649	0.2719	0.2727	0.0641
63 Active Min	0.2000	0.3072	0.3215	0.0423	0.0647	0.2725	0.2725	0.0647
64 Idle Mean	0.2100	0.3045	0.3186	0.0439	0.0869	0.2913	0.2913	0.0869
65 Idle Std	0.1900	0.3019	0.3153	0.0407	0.0613	0.2657	0.2657	0.0613
66 Idle Max	0.2100	0.3045	0.3189	0.0442	0.0871	0.2915	0.2915	0.0871
67 Idle Min	0.2100	0.3060	0.3202	0.0442	0.0872	0.2925	0.2926	0.0871

A.17. táblázat A Catboost algoritmus teljesítményének összehasonlítása

Dataset	Jellemezők száma		Train Dataset				Test Dataset			
		Models	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
FTP	5	Naive Bayes	*0,9970*	*0,9924*	*0,9944*	*0,9934*	*0,9969*	*0,9916*	*0,9948*	*0,9932*
		Logistic Regression	*0,9991*	*0,9962*	*1,0000*	*0,9981*	*0,9992*	*0,9964*	*1,0000*	*0,9982*
		Decision Tree	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*0,9999*	*0,9999*	*0,9999*
		SVM	*0,9999*	*0,9996*	*1,0000*	*0,9998*	*0,9999*	*0,9995*	*1,0000*	*0,9997*
		Random Forest	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*1,0000*
		CatBoost	***1,0000***	***1,0000***	***1,0000***	***1,0000***	***1,0000***	***0,9999***	***1,0000***	***1,0000***
SSH	6	Naive Bayes	*0,9999*	*0,9999*	*0,9997*	*0,9998*	*0,9999*	*0,9999*	*0,9998*	*0,9999*
		Logistic Regression	*0,9969*	*0,9861*	*1,0000*	*0,9930*	*0,9969*	*0,9862*	*1,0000*	*0,9930*
		Decision Tree	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*0,9998*	*1,0000*	*0,9999*
		SVM	*0,9999*	*0,9999*	*0,9998*	*0,9998*	*1,0000*	*1,0000*	*0,9998*	*0,9999*
		Random Forest	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*1,0000*	*0,9998*	*1,0000*	*0,9999*
		CatBoost	***1,0000***	***0,9999***	***1,0000***	***1,0000***	***1,0000***	***0,9998***	***1,0000***	***0,9999***
SQL	7	Naive Bayes	*0,9998*	*0,0000*	*0,0000*	*0,0000*	*0,9996*	*0,0000*	*0,0000*	*0,0000*
		Logistic Regression	*0,9998*	*1,0000*	*0,0345*	*0,0667*	*0,9996*	*1,0000*	*0,0345*	*0,0667*
		Decision Tree	*1,0000*	*1,0000*	*0,9540*	*0,9765*	*1,0000*	*1,0000*	*0,9540*	*0,9765*
		SVM	*0,9999*	*1,0000*	*0,3793*	*0,5500*	*0,9997*	*1,0000*	*0,3793*	*0,5500*
		Random Forest	*1,0000*	*1,0000*	*0,9770*	*0,9884*	*1,0000*	*1,0000*	*0,9770*	*0,9884*
		CatBoost	***1,0000***	***0,9759***	***0,9310***	***0,9529***	***1,0000***	***1,0000***	***0,9310***	***0,9643***
XSS	2	Naive Bayes	*0,9994*	*0,0000*	*0,0000*	*0,0000*	*0,9989*	*0,0000*	*0,0000*	*0,0000*
		Logistic Regression	*0,9994*	*0,0000*	*0,0000*	*0,0000*	*0,9989*	*0,0000*	*0,0000*	*0,0000*
		Decision Tree	*0,9999*	*0,9397*	*0,9478*	*0,9437*	*0,9999*	*0,9820*	*0,9478*	*0,9646*
		SVM	*0,7809*	*0,0012*	*0,4870*	*0,0024*	*0,7804*	*0,0024*	*0,4870*	*0,0049*
		Random Forest	*0,9999*	*0,9522*	*0,9522*	*0,9522*	*0,9999*	*0,9821*	*0,9522*	*0,9669*
		CatBoost	***0,9998***	***0,7926***	***0,9304***	***0,8560***	***0,9998***	***0,9264***	***0,9304***	***0,9224***
WEB	13	Naive Bayes	*0,9917*	*0,1064*	*0,6285*	*0,1820*	*0,9912*	*0,1915*	*0,6285*	*0,2936*
		Logistic Regression	*0,9984*	*0,0000*	*0,0000*	*0,0000*	*0,9969*	*0,0000*	*0,0000*	*0,0000*
		Decision Tree	*0,9998*	*0,9797*	*0,8674*	*0,9201*	*0,9995*	*0,9498*	*0,8674*	*0,9068*
		SVM	*0,4697*	*0,0010*	*0,3552*	*0,0020*	*0,4680*	*0,0020*	*0,3552*	*0,0039*
		Random Forest	*0,9996*	*1,0000*	*0,7447*	*0,8537*	*0,9993*	*1,0000*	*0,7447*	*0,8537*
		CatBoost	***0,9996***	***0,9978***	***0,7463***	***0,8539***	***0,9993***	***0,9978***	***0,7463***	***0,8539***