

OTDK-dolgozat

2023

SELYE JÁNOS EGYETEM
GAZDASÁGTUDOMÁNYI ÉS INFORMATIKAI KAR



KRIPTOVALUTA ELŐREJELZÉS SVM GÉPI TANULÓ
ALGORITMUSSAL

CRYPTOCURRENCY PREDICTION WITH SVM MACHINE
LEARNING ALGORITHM

Mgr. Izsák Tilla

Konzulens: Marák László, PhD.

Kézirat lezárásának dátuma: 2023. 01. 10.

TARTALOMJEGYZÉK

1.	BEVEZETÉS	1
2.	A TÉMAKÖR HAZAI ÉS NEMZETKÖZI SZAKIRODALMA.....	2
2.1.	GÉPI TANULÁS	2
2.1.1.	<i>Support Vector Machine (SVM)</i>	2
2.1.2.	<i>Lineáris SVM</i>	2
2.1.3.	<i>Kernel trükk</i>	3
2.1.4.	<i>Kernel függvények</i>	4
2.1.5.	<i>Távolságok koncentrációja, vagyis a dimenzionalitás átka</i>	5
2.1.6.	<i>MRMR – Minimális redundancia leírók kiválasztása</i>	5
2.1.7.	<i>Osztályozás (Classification)</i>	6
2.2.	KRIPTOTÖZSDE	6
2.2.1.	<i>Kriptovaluták</i>	7
3.	A KUTATÁS MÓDSZERTANA	9
3.1.	A MEGOLDANDÓ PROBLÉMA MEGFOGALMAZÁSA	9
3.2.	A PROGRAM FELÉPÍTÉSE	9
3.2.1.	<i>Binance API és adatgyűjtés</i>	9
3.2.2.	<i>Tanuló és teszt adatok</i>	10
3.2.3.	<i>Kereskedési stratégia</i>	11
3.2.4.	<i>A minták feldolgozása</i>	12
3.2.5.	<i>Az SVM Modell felépítése</i>	13
3.2.6.	<i>SVM Tanulás</i>	15
3.3.	TESZTELÉS.....	16
4.	MEGVALÓSÍTÁS	18
4.1.	A KERESKEDÉS AUTOMATIZÁLÁSA	18
	ÖSSZEFOGLALÁS.....	19
	IRODALOMJEGYZÉK.....	20

1. BEVEZETÉS

A kriptovaluták egyre növekvő népszerűsége tesznek szert, emellett kereskedésük a kriptotőzsdéknek köszönhetően viszonylag egyszerűvé vált az átlagember számára is. Nyereséges stratégia-e a napi és rövid távú kereskedés a kriptovaluta piacon? Tudunk-e jobb teljesítményt elérni a hosszú vagy középtávú vétel-tartás stratégiához képest? Megvizsgáljuk, hogy a nagy volatilitás és kockázatok ellenére, a gépi tanulás használata alkalmas-e arra, hogy sikeresen találjon az árfolyam-adatokban olyan mintákat, amelyek alapján a rövid távú növekedés előre jelezhető. A kutatás célja egy olyan prediktív modell tervezése és implementációja, amely megfelelő pontossággal és jövedelmezőséggel képes előre jelezni a kriptovaluta árfolyam változásokat.

A kriptovaluták népszerűsége 2017-ben az egekbe szökött, a piaci kapitalizációjuk több egymást követő hónapon át tartó exponenciális növekedése miatt, amely 2018 januárjában érte el a 800 milliárd dollár feletti csúcst. Manapság a kriptovaluta kereskedés egy népszerű kereskedési forma. A közelmúltbeli koncepciójuk ellenére széles körű piaci elfogadottságot és gyors fejlődést tapasztalhatunk velük kapcsolatban. Jelenleg is nagy választékkal szolgálnak különböző online tőzsdék, mint a Binance, Coinbase, Revolut, stb. Ezeken a tőzsdéken a kriptovaluták hagyományos fiat valutával is megvásárolhatóak. Sok fedezeti alap és vagyonkezelő elkezdte a kriptovalutákkal kapcsolatos eszközöket beépíteni portfóliójába és kereskedési stratégiájába. Az akadémiai közösség hasonlóképpen jelentős erőfeszítéseket tett a kriptovaluta kereskedési stratégiák vizsgálatára (Kok, 2020),(Izzat, 2020),(Alessandretti 2018) . A gépi tanulás fejlődésével párhuzamosan nőtt az érdeklődés és a kereslet az olyan modellek iránt, amik képesek a kriptovaluta árfolyamok előrejelzésére. Azonban népszerűsége ellenére még mindig nagy kihívást jelent a kutatók számára egy olyan prediktív modell elkészítése, ami képes lenne pontosan megjósolni az árfolyamok jövőbeli mozgását (Lakshmi, 2021). Előrejelzésük egy izgalmas kihívás a pénzügyi világban, tehát egy sikeres algoritmus vagy modell alkalmazása nagy pénzügyi előnyökkel járhatna és csökkenthetné a befektetési kockázatokat. Az árfolyamok előrejelzéséhez többféle technika létezik, amire a későbbiekben még bővebben kitérünk, mi azonban a Support Vector Machine (SVM) algoritmust alkalmaztuk a kutatásaink során.

2. A TÉMAKÖR HAZAI ÉS NEMZETKÖZI SZAKIRODALMA

2.1. GÉPI TANULÁS

A gépi tanulás, mint fogalom az 1950-es években született, és az 1990-es évektől már önálló területként tartják számon (Samuel, 1959). Ezek a rendszerek a bevitt adatokból önállóan vagy emberi segítséggel képesek szabályszerűségeket vagy mintákat felismerni és megjegyezni, majd pedig az ismert adatok alapján megtanult szabályokat ismeretlen adatokra is alkalmazni (Kohavi, 1998). A gépi tanulási metódusokat/algorithmusokat főként a megadott adatokból előrejelzésekre (predictions) vagy osztályozások (classifications) készítésére használják. Az algoritmus egy adott bemenetre ad előrejelzést. A tanulás során a tanulóalgoritmus megpróbálja azonosítani a változókon belüli kapcsolatokat és kölcsönhatásokat, melyeket később az előrejelzéshez használ a szimulációs fázisban.

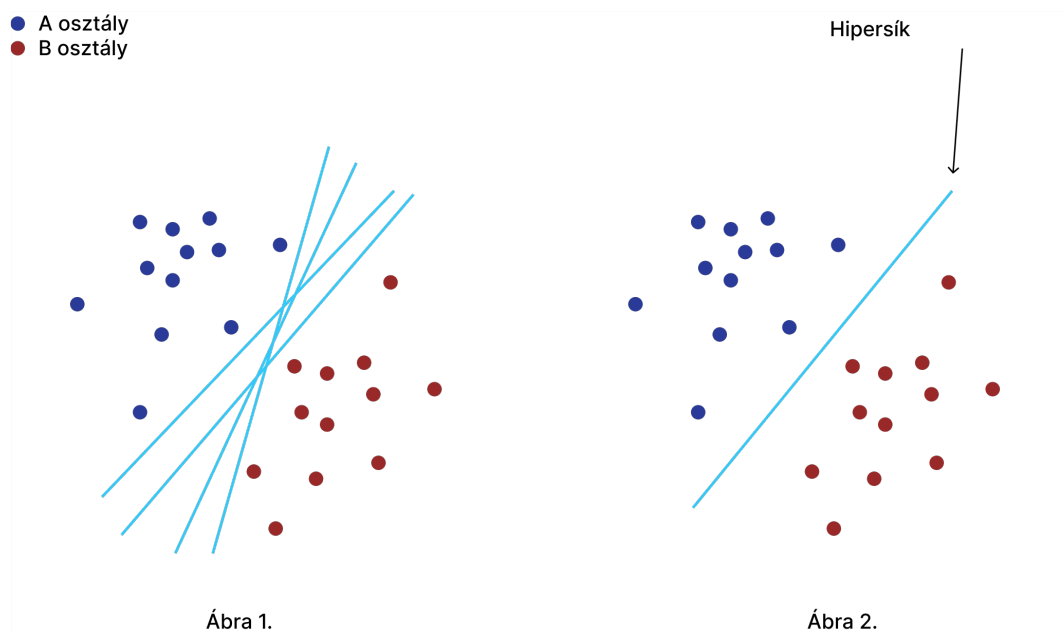
2.1.1. Support Vector Machine (SVM)

Az SVM egy osztályozási algoritmus, mely ígéretes eredményeket mutat sok gyakorlati alkalmazásban, a kézzel írt számjegyek felismerésétől kezdve szövegfelismerésig. Az SVM nagyon jól működik sokdimenziós adatokkal. A módszer egy másik egyedi jellege, hogy a döntési határt a tanuló esetek egy részhalmazának segítségével reprezentálja, amelyeket tartóvektoroknak (support vector) nevezünk (Vapnik, 1995). Ezen tartóvektorok száma indikációul szolgál a túltanulás vizsgálatához.

2.1.2. Lineáris SVM

Az SVM alapváltozata lineáris szeparálásra képes, amely azonban kiterjeszthető nemlineáris szeparálásra és nemlineáris regressziós feladatokra is. Amíg a (Ábra 1.) bal oldalán látható illusztráción az összes egyenes vonal szétválasztja a tanuló-pontokat, addig a jobb oldalon látható (Ábra 2.) kép esetén a választott egyenes a legjobb a lehetséges megoldások között, mert az van a lehető legtávolabb mindkét halmaz elemeitől. A mintapontokból történő tanulásnál ugyanis nemcsak a tanuló-pontok megtanulására törekszünk, hanem egy minél hatékonyabb általánosító képesség elérésére is. A feladatunk a véges számú ismert válaszu tanítópont helyes osztályozása és az egész osztályozási feladat minél jobb “megtanulása” is. Célunk megkeresni azt a hipersíkot, amely a legjobb

általánosító-képességgel rendelkezik (legjobban alkalmazható majd az ismeretlen új adatokra), amely a két osztályba tartozó tanítópontok között, a tanuló-pontoktól a lehető legnagyobb távolságra helyezkedik el. A lineáris kétosztályos osztályozási feladat megoldását adó SVM ezt a legoptimálisabb elválasztó felületet keresi meg. A magasabb dimenziós leíróterben a hipersík magasabb dimenzióssá válik, amely lehet lineáris vagy nem lineáris. A döntési határ továbbra is egy $n-1$ dimenziójú hipersík, ahol n a leíróter dimenziója. Az SVM-ben olyan hipersíkot keresünk, amely az adatokat két régióra választja szét, és az adatoktól a lehető legnagyobb távolságra helyezkedik el (Beutelspacher, 1998).



1. ábra és 2. ábra: Egy lineárisan szeparálható feladat különböző megoldásai

2.1.3. Kernel trükk

A kernel-trükk egy SVM-ben alkalmazott módszer, amely a bemeneti adatokat egy magasabb dimenziós térbe ágyazza be, ahol a lineáris döntési határ könnyebben megtalálható. A kernel-trükk lényege egy kernel-függvény használata, amely egy olyan matematikai függvény, amely képes a bemeneti adatokat egy magasabb dimenziós térbe leképezni anélkül, hogy ténylegesen kiszámítaná az adatok koordinátáit az adott térben. (Sergios, 2008).

2.1.4. Kernel függvények

Az SVM-ben használt gyakori kernelfüggvények a következők:

- Lineáris kernel,
- Polinomiális kernel,
- Radial basis function (RBF) kernel, más néven Gauss-kernel.

A kernel függvény kiválasztása az adatok jellegétől és az adott problémától függ. Egyes kernel függvények bizonyos típusú adatokhoz jobban illeszkednek, mint másokhoz. Az RBF kernelt például gyakran használják kép- és beszéd felismerési problémákban, míg a polinomiális kernelt gyakran használják szövegosztályozási problémákban. Például egy lineáris kernel használatakor a döntési határ egy egyenes vonal, de egy polinomiális kernel használatakor a határ lehet egy parabola, az RBF kernel használatával pedig a döntési határ hipersík lesz. Az 1. táblázatban láthatjuk, hogy az egyes kernel függvények használatával olyan kernel térbeli háló struktúrát kapunk, amely egy bizonyos klasszikus neuronháló struktúrához hasonló. A Gauss kernel függvény az RBF hálóval rokon struktúrájú, a tangens hiperbolikus kernel függvény hasonló mint a MLP hálózat és a kimenete lineáris (Rasmussen, 2006).

1. táblázat: Kernel függvények

Lineáris	$K(\mathbf{x}, \mathbf{x}_i) = \mathbf{x}_i^T \mathbf{x}$
Polinomiális (d foksámú)	$K(\mathbf{x}, \mathbf{x}_i) = (\mathbf{x}_i^T \mathbf{x} + 1)^d$
Gauss (RBF)	$K(\mathbf{x}, \mathbf{x}_i) = \exp\{-\ \mathbf{x} - \mathbf{x}_i\ ^2 / \sigma^2\}$, ahol σ konstans.
Tangens hiperbolikus (MLP)	$K(\mathbf{x}, \mathbf{x}_i) = \tanh(k \mathbf{x}_i^T \mathbf{x} + \theta)$, ahol k és θ megfelelően választott konstansok, mert nem minden kombináció eredményez magfüggvényt.

2.1.5. Távolságok koncentrációja, vagyis a dimenzionalitás átka

A dimenzionalitás átka egy olyan jelenség, amely akkor jelentkezik, amikor nagy dimenziós adatokkal dolgozunk. Arra utal, hogy a nagydimenziós térben lévő adatok kezelése nehézségekbe ütközik a dimenziók nagy száma és az egyes dimenziókban található megfigyelések viszonylag kis száma miatt. Ez a jelenség nagy dimenziós terekben fordul elő, alacsony dimenziós környezetben nem találkozunk vele. A kifejezést Richard E. Bellman alkotta meg, amikor a dinamikus programozás problémáit vizsgálta (Bellman, 2003). A nagydimenziós adatok egyik fő problémája, hogy a dimenziók számával exponenciálisan nő a leíróterben az értékek lehetséges kombinációinak száma. Ez azt jelenti, hogy a dimenziók számának növekedésével a leíróter térfogata gyorsan nő, és az adatok ritkábbá válnak. Ez azért problematikus, mert a hagyományos statisztikai módszerek, például a valószínűségi sűrűségbecslés és a legközelebbi szomszédos módszerek azon a feltételezésen alapulnak, hogy az adatok nem túl ritkák. Ha az adatok ritkák, ezek a módszerek megbízhatatlan vagy félrevezető eredményeket adhatnak. A nagydimenziós adatokkal kapcsolatos másik probléma, hogy a leíróter pontjai közötti távolság kevésbé informatív válik. A nagydimenziós térben a legtöbb pont messze van egymástól, így a távolsági metrika elveszíti megkülönböztető jellegét. Ez azt jelenti, hogy nehéz lesz különbséget tenni az adatok különböző osztályai között, vagy mintákat azonosítani az adatokban.

2.1.6. MRMR – Minimális redundancia leírók kiválasztása

A dimenziócsökkentéshez használható egyik módszer, az MRMR, egy olyan algoritmus, amelyet releváns leírók kiválasztásához alkalmazhatnak. A leírók kiválasztása, a mintafelismerés és a gépi tanulás egyik alapvető problémája. A nagy dimenziójú adathalmazok gyakran tartalmaznak releváns, de redundáns információkat, az mRMR pedig megpróbálja megoldani ezt a problémát a redundáns részhalmazok eltávolításával. A leírók többféle képpen is kiválaszthatóak. Az egyik séma szerint olyan leírókat választunk, amelyeknek a lehető legtöbb kölcsönös információjuk van a megtanulandó osztállyal. Ezt nevezik maximális relevanciájú kiválasztásnak. A másik módszer szerint a leírókat pedig úgy választjuk ki hogy azoknak a lehető legkevesebb kölcsönös információja legyen a többi kiválasztott leíróval. Ezt nevezik minimális redundanciának. A minimumális redundancia

és a maximális relevancia módszerrel eltávolíthatunk sok irreleváns leíró, ami által csökkentjük a mintaadatok dimenzióit megőrizve az osztályozás szempontjából fontos információkat (Auffarth, 2020).

2.1.7. Osztályozás (Classification)

Osztályozó gépi tanuló modelleket számos tudományterületen alkalmaznak. Az osztályozó modellekben a tanuló adatokat a modell felépítésére használják. A modellekkel megpróbálják megjósolni a címkéket az új mintákhoz. Az osztályozási modellek kimenetei lehetnek diszkrét, mint a döntési fa, vagy pedig folytonosak, mint az SVM (Duda, 2001). Az osztályozási teljesítményt skaláris értékek képviselik, pontosságban, érzékenységekben és specificitásban. A különböző osztályozók összehasonlítása ezekkel a mérőszámokkal egyszerű, de számos problémát vet fel, mint például a kiegyensúlyozatlan adatokra való érzékenység és egyes osztályok teljesítményének figyelmen kívül hagyása. A grafikus értékelési módszerek, mint például a ROC (Receiver Operating Characteristics) egy árnyaltabb értelmezést adnak az osztályozási teljesítményről, de konfúziós mátrixot is használhatunk a diagnosztikai teszt kiértékeléséhez (Nello, 2000).

2.2. KRIPTOTÓZSDE

A kriptovaluta tőzsde vagy a digitális valutaváltók lehetővé teszik az ügyfelek számára, hogy digitális valutákat vásároljanak, tároljanak és kereskedjenek velük. A tőzsdei előrejelzések gépi tanulási technikáinak sikere arra utal, hogy ezek a módszerek a kriptovaluták árának előrejelzésében is hatékonyak lehetnek (Enke, 2005).

Egyes bróker cégek (pl. eToro, Revolut) lehetővé teszik a felhasználók számára, hogy kriptovalutákat vásároljanak, de azok nem kivehetőek vagy továbbíthatóak a platform kriptovaluta-pénztárcájából. Más kriptovaluta tőzsdék (pl. Binance, Coinbase) azonban lehetővé teszik a kriptovaluta kivonást vagy utalást másik kripto-pénztárcába. A hagyományos tőzsdéken is megjelentek a határidős kriptovaluta ügyletek, jellemzően future formájában. Az általunk választott és használt kripto-tőzsde a Binance. A platform mára piacvezetővé nőtte ki magát a kriptovaluta kereskedés terén, mivel a jutalékok versenyképesek és a sok kripto-tőzsdéhez képest alacsony díjakkal dolgoznak (Davda, 2021).

2.2.1. Kriptovaluták

A kriptovaluták egyik legfontosabb tulajdonsága, hogy megszünteti a megbízható harmadik fél szükségességét. A blokklánc a bitcoin mögötti kulcsfontosságú technológia, amely nyilvános digitális főkönyvként működik, ahol a felhasználók közötti tranzakciókat rögzítik. Mivel nem létezik központi hatóság, ez a főkönyv szétsztható a hálózat résztvevői között, akik dedikált szoftver segítségével közösen tartják karban (Yaga et al. 2019). A blokklánc rögzíti az olyan tranzakciókkal kapcsolatos információkat, mint hogy honnan küldték és hova ment, a tranzakció idejét, az összeget, a kifizetett díjat és hasonló adatokat. Az összes ilyen adat blokkban tárolódik, amik láncként kapcsolódnak egymáshoz.

A magas volatilitás, a szélsőséges rövid távú hozam és a buborékszerű ár viselkedése miatt a korai kutatások vitatták, hogy ténylegesen egy új típusú valutáról vagy pusztán spekulatív eszközről van-e szó (Yermack, 2015).

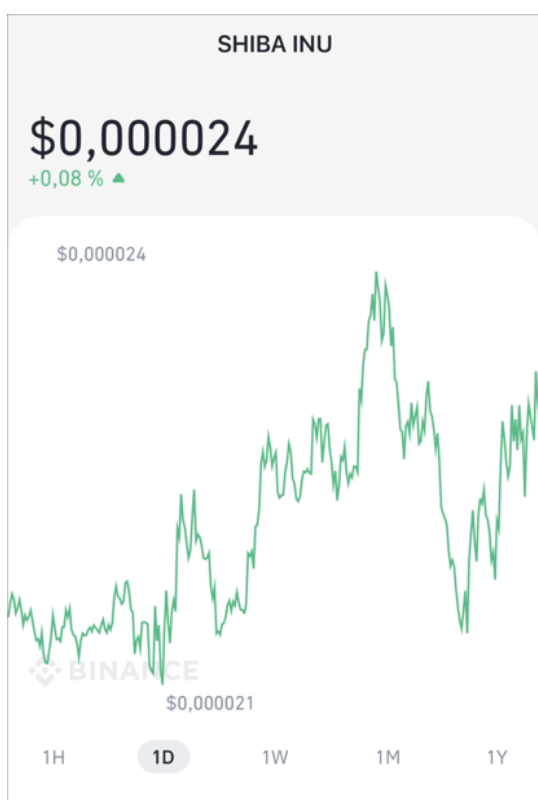
A kriptovaluták néhány fontos tulajdonsága:

- papírmintes - elektronikus rendszer, amely kriptográfiai entitásokon alapul
- decentralizált - nem kell megbízható harmadik fél közvetítő
- hozzáférhető és konszenzusos - nyílt forráskódú, a hálózat kezeli a valuták egyenlegét
- globális - használatának nincs földrajzi vagy gazdasági akadálya
- olcsó - az átutalási költségek viszonylag alacsonyak
- visszafordíthatatlan és megváltoztathatatlan - a bitcoin tranzakciókat nem lehet visszafordítani, és a blokkláncba való rögzítés után a kereskedés nem módosítható
- biztonságos - a hálózat bizonyítottan ellenáll a támadásoknak
- diszkrét - a rendszer nem fedi fel a felhasználók kilétét, de nyilvánosságra hozza a pénztárcáik címét és tartalmát

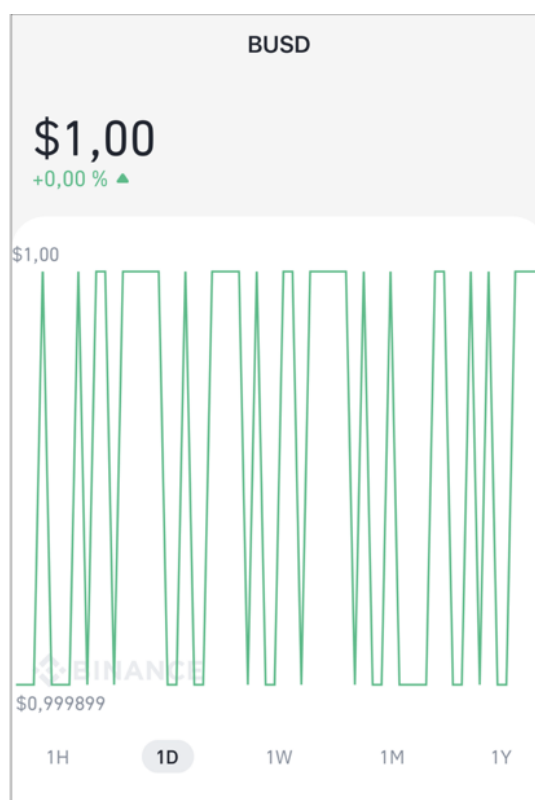
A Binance applikációjában való kriptovaluta kereskedéshez két féle kriptovalutára van szükségünk (mivel fiat valutát nem nyereséges erre használni), amiket BASE Asset (BA) és QUOTE Asset-nek (QA) neveznek. A BA esetünkben a Shiba Coin lesz (3.ábra), az QA pedig a BUSD (4.ábra). A BUSD egy úgynevezett *stablecoin*, ami olyan kriptovaluta, ahol az ár egy fiat pénzhez (esetünkben USD) vagy tőzsdén kereskedett árukhoz (például nemesfémekhez vagy ipari fémekhez) köthető (Ossinger, 2021).

Az általunk vizsgált kriptovaluták a következők:

- XRP -Ripple coin
- SHIB -Shiba Inu coin
- ATP - Atlas Protocol coin
- CHZ - Chiliz coin
- PHB - Phoenix coin



3. ábra: SHIB napi árfolyama



4. ábra: BUSD napi árfolyama

3. A KUTATÁS MÓDSZERTANA

A kutatás célja egy olyan SVM osztályozó modell tervezése és implementációja, amely megfelelő pontossággal és jövedelmezőséggel képes előre jelezni a kriptovaluta árfolyam trendeket. A modellnek képesnek kell lennie osztályozni az adatmintákat, vagyis el kell döntenie hogy az adott pillanatban a részvény érdemes-e a vásárlása vagy sem.

3.1. A MEGOLDANDÓ PROBLÉMA MEGFOGALMAZÁSA

Mivel a kriptotőzsde még a hagyományos tőzsdével szemben is sokkal volatilisabb és rendezetlenebb, ezért arra keresünk választ a kutatás során, hogy vajon tudunk tanítani egy modellt ami nyereséges a napi és rövid távú kereskedésben. A két fő megoldandó probléma a megfelelő kereskedési stratégia kiválasztása, és a predikcióra legjobban alkalmas leírók kiválasztása.

3.2. A PROGRAM FELÉPÍTÉSE

A munkánk négy főbb lépésből áll:

1. Hozzáférés az adatokhoz (Binance)
2. Leírók kiválasztása, kinyerése, adatfeldolgozás
3. A modell betanítása
4. Kereskedés automatizálása

3.2.1. Binance API és adatgyűjtés

Az adatok kinyerésére a Binance alkalmazását használtuk, mivel a többi kripto-tőzsdéhez képest alacsony díjakkal jár, emellett nagy mennyiségű dokumentációt találhatunk róla az interneten. A Binance rendelkezik egy RESTful API-val, amely HTTP lekéréseket használ az adatok küldésére és fogadására. A Binance API egy olyan interfész, amely alkalmas kereskedési stratégiák automatizálására, valós idejű piaci adatok lekérdezésére, és a felhasználó számlaegyenlegének és portfóliójának megfigyelésére. Az API-kliens alkalmazás tcp-socket kapcsolatot használ. A felhasználónak manuálisan kell megadnia

felhasználónevét és jelszavát a bejelentkezési oldalon, és kérvényeznie kell egy API kulcsot. Az API-kulcs egy egyedi azonosító, amely a felhasználó, fejlesztő vagy program API-hoz való csatlakozás hitelesítésére szolgál. Az API lehetővé teszi, hogy többféle programozási nyelv segítségével csatlakozzunk a Binance szerverekhez.

Az adataink feldolgozását, a modell tanítását és az automatizált vasalást a Matlab segítségével építjük fel. A Matlab modul az egyedi API azonosítóval csatlakozik a szerverhez, amelyről így lekérdezhetjük az aktuális és múltbeli árfolyamadatokat. Az API lehetőséget nyújt valós és historikus idejű piaci adatok lekérdezésére és kriptovaluta kereskedésre is. Egy működő kereskedési stratégia megvalósításához a modellnek képesnek kell lennie arra, hogy nagyobb árváltozások alapján előrejelzéseket adjon a magasabb jövedelmezőség eléréséhez, ezért a megfelelő tanulóadatok kiválasztása az első lépés.

3.2.2. Tanuló és teszt adatok

A tanuló adatok az XRP_BUSD valuta-pár esetén 2021 január 01. és 2022 december 11. között keletkeztek, és 1 perces mérési részletességgel kérdeztük le őket, ez 627 229 egyedi árfolyamadatot jelentett. Az 1 perces intervallum a legalacsonyabb hozzáférhető intervallum a Binance letölthető múltbeli adatai közül. Az XRP adatait fogjuk használni a modell tanítására, míg a többi valután csak tesztelésre. A teszt adatok mind az öt valuta esetén 2022 december 12. és 2023 január 10. közötti időszakot fedték le, szintén 1 perces intervallumokban, ami 44 771 mérést jelentett. Minden tanítás során a konkrét tanuló adatbázist úgy állítottuk össze, hogy abban csak múltbeli (egy adott dátumot megelőző) adatok legyenek, a tesztadatbázist pedig úgy hogy abban csak jövőbeli (az adott dátumot követő) adatok. Így a tanuló adatbázis és a teszt adatbázis az egyik tanítás során sem tartalmazott közös adatokat. A tanuló adatok 50-50%-ban tartalmaznak pozitív és negatív mintákat, amíg a tesztelésre szánt adatokban a pozitív és negatív minták aránya megközelítőleg valós elosztású (30% pozitív és 70% negatív).

1. táblázat: Adatfolyam leírók percenként (részlet)

60x7 timetable								
	Time	1 open	2 high	3 low	4 close	5 volume	6 quoteVolume	7 numTrades
1	05-Apr-2022 00:48:00	2.6560e-05	2.6580e-05	2.6560e-05	2.6580e-05	258572728	6.8691e+03	24
2	05-Apr-2022 00:49:00	2.6590e-05	2.6600e-05	2.6580e-05	2.6600e-05	953468354	2.5360e+04	63
3	05-Apr-2022 00:50:00	2.6590e-05	2.6770e-05	2.6590e-05	2.6770e-05	6.4109e+09	1.7102e+05	603
4	05-Apr-2022 00:51:00	2.6760e-05	2.6760e-05	2.6690e-05	2.6690e-05	2.9139e+09	7.7870e+04	163
5	05-Apr-2022 00:52:00	2.6690e-05	2.6730e-05	2.6680e-05	2.6710e-05	2.1543e+09	5.7541e+04	49
6	05-Apr-2022 00:53:00	2.6710e-05	2.6770e-05	2.6710e-05	2.6760e-05	939473311	2.5132e+04	52
7	05-Apr-2022 00:54:00	2.6760e-05	2.6770e-05	2.6740e-05	2.6750e-05	739162631	1.9775e+04	40
8	05-Apr-2022 00:55:00	2.6760e-05	2.6780e-05	2.6740e-05	2.6740e-05	1.4615e+09	3.9110e+04	62
9	05-Apr-2022 00:56:00	2.6740e-05	2.6740e-05	2.6690e-05	2.6700e-05	2.3442e+09	6.2618e+04	89
10	05-Apr-2022 00:57:00	2.6710e-05	2.6730e-05	2.6710e-05	2.6720e-05	222016824	5.9317e+03	24
11	05-Apr-2022 00:58:00	2.6720e-05	2.6730e-05	2.6720e-05	2.6720e-05	332633561	8.8899e+03	18
12	05-Apr-2022 00:59:00	2.6730e-05	2.6740e-05	2.6690e-05	2.6730e-05	293489357	7.8413e+03	20
13	05-Apr-2022 01:00:00	2.6730e-05	2.6760e-05	2.6730e-05	2.6750e-05	1.0607e+09	2.8372e+04	43
14	05-Apr-2022 01:01:00	2.6750e-05	2.6750e-05	2.6700e-05	2.6710e-05	2.1090e+09	5.6398e+04	50
15	05-Apr-2022 01:02:00	2.6700e-05	2.6710e-05	2.6690e-05	2.6690e-05	513188593	1.3704e+04	24

Forrás: Matlab

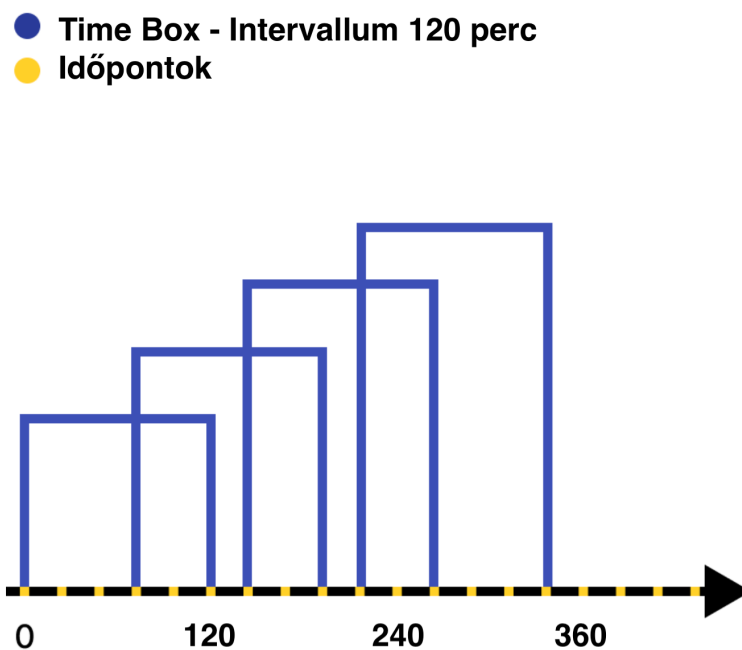
3.2.3. Kereskedési stratégia

Mivel napi kereskedést szeretnénk folytatni, az általunk választott stratégia egy időablakos stratégia, aminek célja, hogy a vásárolt valutát a vásárlás pillanatától számítva, egy általunk választott időkereten belül (120 perc) értékesítjük (Ábra 5.). A napi kereskedés során, az adott eszköz viszonylag rövid élettartamú árváltozásból próbálunk profitot realizálni. A vásárolt eszköz birtoklásának időtartama néhány másodperctől néhány óráig terjedhet, de ritkán haladja meg a kereskedési nap időtartamát.

Azért választottuk ezt a stratégiát mert így az adatelemzés során csak egy szűkebb intervallumot kell vizsgálnunk és ezáltal reményeink szerint növekedhet a predikciós modell pontossága, továbbá mivel így a valuta tartása (hold) rövidebb időre szűkül, csökkenhet a kockázat egy nagy veszteséggel járó kimenetelre.

Egy bináris osztályozó modell számára gyűjtjük az adatokat így szükségünk van pozitív és negatív mintákra, mivel a modellnek képesnek kell lennie osztályozni az adatmintákat úgy, hogy az adott pillanatban a részvény vásárlása érdemes-e vagy sem. A célunk olyan momentumok keresése, ahol az árfolyam egy általunk választott százalékot növekedett (5%) , egy általunk meghatározott időkorláton belül (120 perc). Például, ha az árfolyam 6%-

ot növekedett 45 percen belül - az egy pozitív momentum lesz, negatív momentumot pedig akkor kapunk, ha az árfolyam a megadott 120 percen belül egyszer sem érte el az 5%-os határt. A beállított időkeret lejártá után a pozíciót mindenképpen zárjuk. Amikor a program talál egy pozitív vagy negatív mintát, akkor elmenti a vásárlás pillanatának árfolyam előzményét, egy szintén általunk választott hosszúságú árfolyam adatsort (pl. 200 perc hosszúságú), amihez párosít egy negatív vagy pozitív osztályt. Ezeket az adatmintákat használjuk a gépi tanulásra és tesztelésre.



5. ábra: Az időablak illusztrációja

3.2.4. A minták feldolgozása

Az adat-transzformáció és adatfeldolgozás az adatoknak a gépi tanulási algoritmus által használható formátumba való átalakításának folyamatára utal, amiket a Matlab segítségével végeztünk el. Az első lépésünk az adattisztítás volt, ami az adatokban lévő hibák, ellentmondások és hiányzó értékek azonosításának és javításának folyamata. Az adatok feldolgozása során a kiválasztott tanuló- és tesztadatokat normalizáljuk úgy, hogy azok azonos skálán legyenek. A releváns leírók kiválasztásához MRMR algoritmust használtuk. A mentett minták mellett a könnyebb ellenőrzés végett még szerepel az elért profit %-ban, és a momentum osztálya (pozitív vagy negatív).

3.2.5. Az SVM Modell felépítése

Az tanuló és teszt adatok elődeldolgozása után a Matlab segítségével elkezdjük a modell tanítását és tesztelését. Mivel a gépi tanuló modell több állítható paraméterrel is rendelkezik, így az első lépés az optimális paraméterek keresése. Egy másik változtatható paraméter a tanulóadatok mennyisége is, tehát itt szintén meg kell keresnünk az optimális mennyiséget.

Általunk választott állítható SVM paraméterek:

- Kernel függvény: lineáris, RBF
- Kernel skála: a kernel skála egy olyan paraméter, amelyet bizonyos kernelfüggvényekben, például a radiális bázisú függvény (RBF) kernelben használnak. Az RBF kernel a következőképpen van definiálva: $K(x, x') = \exp(-\gamma * \|x - x'\|^2)$, ahol x és x' bemeneti vektorok, γ a kernel skála paramétere, és $\|x - x'\|$ az x és x' közötti euklideszi távolság. A kernel skála paramétere, a γ , a radiális bázisfüggvény szélességét szabályozza. A kisebb γ érték szélesebb radiális bázisfüggvényt eredményez, ami olyan döntési határt eredményez, amely kevésbé érzékeny az egyes adatpontokra, és ezért jobban általánosítható az új adatokra. A nagyobb γ érték szűkebb radiális függvényt eredményez, ami a képzési adatokhoz való túlillesztéshez vezethet.
- Box constraint paraméter: ez a paraméter arra szolgál, hogy szabályozza a kompromisszumot a margó (a döntési határ és a legközelebbi adatpontok közötti távolság) maximalizálása és a képzési adatok osztályozási hibájának minimalizálása között. μ -vel jelöljük, ami egy nem negatív valós szám, amely a hibásan osztályozott mintákért kapott büntetés. Ha μ kicsi, az SVM nagyobb hangsúlyt fektet a gyakorló példák helyes osztályozására, ami a gyakorló adatokhoz szorosan illeszkedő döntési határhoz vezethet, és túlillesztést eredményezhet. Ezzel szemben, ha μ nagy, akkor az SVM nagyobb hangsúlyt fektet a margó maximalizálására, ami általánosíthatóbb döntési határokat eredményezhet, és jobb általánosítást eredményezhet az új adatokra, viszont megengedőbb a hibásan osztályozott mintákkal szemben.
- Cost: a költségparamétert, az osztályok kiegyensúlyozatlanságának problémájának kezelésére használjuk. Segít eldönteni, hogy mekkora legyen a büntetés súlya egy adott osztály téves osztályozásáért. Pozitív számra állítható, minél nagyobb a költség értéke, annál inkább elkerüli az osztályozó egy pont téves osztályozását.

- Legjobb leírók: az MRMR algoritmus segítségével választott leírók száma.
- Tanuló minták: a tanuló adatminták mennyisége és hossza. Például 1 év adataiból tanul vagy csupán 3 hónap adatait használja.

A felsorolt állítható paraméterek rengeteg kombinációja létezik, ezért az optimális paraméter beállítások automatikus megtalálására Bayes-optimalizálót használtunk. A Bayes-optimalizálás (BO) egy modellfüggvény paramétereinek optimalizálására szolgáló módszer. A BO alkalmas olyan függvények optimalizálására is, melyek kiértékelése sokáig tart. Ez az algoritmus a Bayes-tételt használja fel a célfüggvény valószínűségi eloszlásának modellezésére, és a modell paramétereinek frissítésére. Az optimális paraméterek keresése úgy működik, hogy az algoritmus iteratív módon átjárja a paraméterteret, és az eddigi megfigyelések alapján valószínűségi modellt épít a célfüggvényre (pl. az SVM osztályozó pontossága egy validációs halmazon). Az algoritmus ezt a modellt használja arra, hogy előrejelzéseket készítsen arra vonatkozóan, hogy a paraméterter mely régiói tartalmazznak nagy valószínűséggel jó megoldásokat, és ezen információk alapján választja ki a következő kiértékelendő pontot. A tréning paraméterkeresési szakaszában 5-szörös keresztvalidációt alkalmaztunk $10^{-5} \leq \gamma, \mu \leq 5$ értékkel.

Optimalizációs célfüggvény

A modellünk elkészítése során nagy hangsúlyt fektettünk egy olyan optimalizációra, amikor a modell minimalizálja a rossz (False Positive) osztályt, tehát nagyon kevés vagy nulla veszteséges kereskedést végez. Egy ilyen fajta modell bár kevesebb kereskedést hajt végre, de ellenben minimalizálja a rizikót.

Egy ilyen modell felépítésére szolgáló paraméterek keresése esetén szükségünk lesz egy saját célfüggvényre, mivel az osztályozó pontosságára vagy precizitására alkalmas mérőszámok ebben az esetben használhatatlanok.

A célfüggvény a következőképp épül fel:

$$f = 1 + (p/100.0) - \text{precision}$$

Ahol:

- p az a Fals Pozitív aránya %-ban az összes igaz negatívhoz képest

$$\text{pontosság (precision)} = (TP)/(TP + FP)$$

3.2.6. SVM Tanulás

A paraméter optimalizáló által legjobb eredményeket mutató paraméterek alapján betanítottuk az SVM osztályozó modellt. A tesztelések során többféle mennyiségű tanuló mintával próbálkoztunk a legjobb eredmény elérése érdekében. A végleges verzióban 12000x240 nagyságú mintahalmazt használtunk, mely 50-50%-ban tartalmazott pozitív és negatív mintákat. A bemutatott modell paraméterei a következők:

Kernel függvény: lineáris

Kernel skála: 19.6345

Box constraint: 7.5117

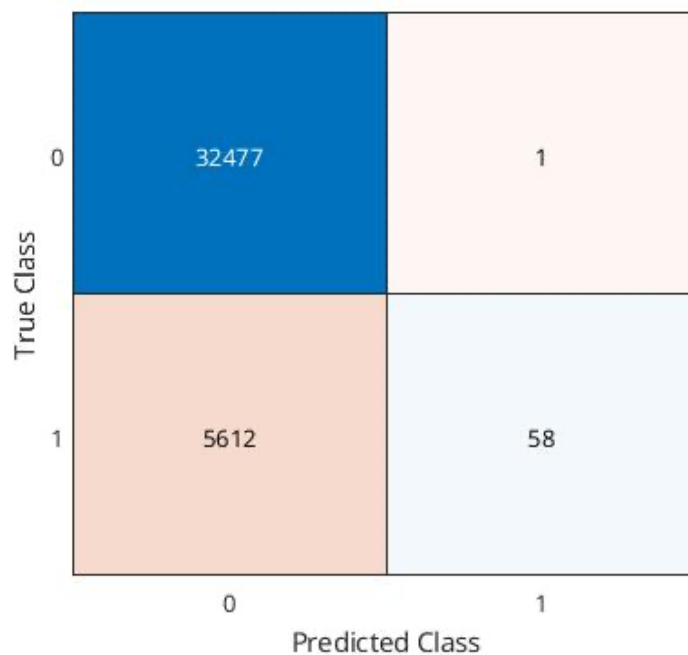
Costs: [0 3;1 0]

Legjobb leírók: legjobb 35 az MRMR rangsorában

Tanuló mintahalmaz: 10 922x35 minta

A XRP_BUSD árfolyamán betanított modellt három különböző árfolyam teszt adathalmazain teszteltük (XRP, APT, CHZ, PHD). Minden teszt adathalmaz esetén az adatok 2022 december 12. után keletkeztek, és a pozitív - negatív minták aránya megközelítőleg, valós elosztású volt (30% pozitív - 70% negatív minta).

Az eredményeket egy konfúziós mátrix segítségével ábrázoltuk, ahol láthatóak a valódi pozitív, negatív valamint a fals pozitív és negatív találatok (6.ábra). Ezekből számunkra a valódi pozitív (TP: 58) és a fals pozitív (FP: 1) aránya számít, mivel ha a modellt majd valuta vásárlásra akarjuk használni, a pozitív osztálycímke eredmény lesz az, amely elindítja a vásárlást az automatizált programban. Tehát, a tesztelések során arra törekedtünk, hogy minél nagyobb legyen a valódi pozitívok találatok száma, és minél kevesebb fals pozitívok száma, ezáltal próbálva növelni a profit mértékét a vásárlásnál.



6. ábra: Konfúziós mátrix (ATP teszt)

Forrás: Matlab

3.3. TESZTELÉS

A teszteléshez kiválasztottuk azt a modellt, amely az optimalizáció során a legjobb eredményeket mutatta, majd a modellt lefuttattuk a teszt adatainkon (ami egy 4 hetes tesztelési időintervallumot jelentett).

A következő eredményeket kaptuk:

XRP - 12 helyes predikció, 0 helytelen

SHIB - 2 helyes predikció, 0 helytelen

ATP - 58 helyes predikció, 1 helytelen (6.ábra)

CHZ - 12 helyes predikció, 2 helytelen

PHB - 1 helyes predikció, 0 helytelen

A szimuláció során, 100 dollár kezdő kerettel indult a program. Részlet a szimuláció folyamatáról, a kezdő összeg, a profit százalékban, és az értékesítés utáni keret:

Van most: 135,36986807883736	profit: 1,0230218558440608	Lett: 136,7544314155707
Van most: 136,7547444155707	profit: 1,0333286993196111	Lett: 138,1678573029652
Van most: 138,16785730496525	profit: 1,006200275810528	Lett: 139,5581026642291
Van most: 139,55810266422918	profit: 1,0047206703418492	Lett: 140,9630629348868
Van most: 140,963446293088688	profit: 1,0033676837920453	Lett: 142,3774407504144

A szimuláció során kedvező eredményt kaptunk, a modell 100 dollár induló tőke esetén a négy hét elteltével az egyenleg 142,37 dolláron állt. Ez több mint 42% hasznot jelent.

4. MEGVALÓSÍTÁS

4.1. A KERESKEDÉS AUTOMATIZÁLÁSA

A kiválasztott modellünket importáltuk a Binance API-ba a Matlab segítségével. Az automatizált kereskedés során a program 1 percenként lekérdezi a valuta adatait (ami az általunk választott leírókat tartalmazza) négy órára visszamenőleg ami egy 240x7 elemű táblázatot eredményez. Ezt a programunk mátrixxá alakítja, normalizálja, beállítja a megadott paramétereket, majd soronként szimulálja a betanított osztályozónk választát. Eztán a program osztályozza a kapott mintát és eredményül egy negatív vagy egy pozitív osztály címkét ad kimenetnek. Amennyiben a kimenet negatív a program vár egy percet és újra elkezd a folyamatot. Abban az esetben ha a modell kimenete pozitív (vagyis szerinte az árfolyam a következő 2 órában eléri az általunk kívánt emelkedést) a program megnézi van e mozgatható összeg (BUSD) a számlánkon, és ha igen akkor elküldi a utasítást a pozíció megnyitására. Miután a vásárlás megtörtén a programunk küld egy LIMIT rendelést az eladásra, a megvásárolt árfolyamhoz viszonyított 6%-al nagyobb árfolyamon. Ha az árfolyam eléri a megadott növekedési határt akkor a program zárja a pozíciót. Ha az árfolyam nem éri el a megadott határt akkor az időkorlát elérése után szintén zárja a pozíciót tekintet nélkül annak értékére. A program ezután folytatja az adatok vizsgálatát a modell segítségével, és minden alkalommal, ha az pozitív kimenetet ad az osztályozó és van mozgatható egyenleg a számlán akkor új pozíciót nyit.

ÖSSZEFOGLALÁS

A digitális blokklánc-platformokon a kriptovaluta kereskedésben tapasztalható jelentős fellendülés miatt a gépi tanulási rendszerek alkalmazása további népszerűsége tette szert. Az intelligens gépi tanulási rendszerek számos előnnyel járnak a nagy adathalmazok modellezésében és előrejelzésében és egyre elterjedtebb használatuk a kriptovaluta kereskedés körében is. A Support Vector Machine gépi tanulási algoritmus használata kiváló megoldást jelenthet a kereskedés során, mivel segítségével nyereségesen kereskedhetünk egy extrém volatilis piacon, mivel képes lehet megfelelő pontossággal előre jelezni a kriptovaluta trendeket. A kutatás arra is rávilágított, hogy a Matlab SVM perdikciós modellje kiegészítve a Binance applikációjával remekül használható valós idejű automatizált kriptovaluta kereskedéshez.

A program elkészítése során fontos stratégiai pont volt számunkra, hogy nem csak a helyes osztályok eltalálása volt a fő célunk (true positive), hanem a hibás (false positive) találatok minimálisra csökkentése. A szimulációhoz olyan modellt választottunk, amelyik a tesztelések során a legkevesebb false positive osztályt produkálta, hogy ezáltal minimálisra csökkentsük a kedvezőtlen vásárlások számát. A választott kriptovaluta párt többféle idő intervallumokban tanítottuk és teszteltük. A legjobb eredményeket egy 10 922x35 dimenziójú tanuló mintahalmazzal érték el. A gépi tanuláshoz egy bináris-osztályozó modellt, Support Vector Maschine-t használtunk, aminek paramétereit egy Bayes-optimalizáló algoritmussal állítottuk be. A betanított modellt több kriptovaluta árfolyamon is teszteltük, és többségében nyereséges tranzakciókat hajtott végre.

A kutatás ezen a területen sok további fejlesztésekre ad lehetőséget, például a leírók változtatásával, további tanuló algoritmusok és más kriptovaluta párok tesztelésével.

IRODALOMJEGYZÉK

Szakirodalom

ALESSANDRETTI, L.,(2018), Anticipating Cryptocurrency Prices Using Machine Learning", Complexity, vol.2 Article ID 8983590, 16 pages, <https://doi.org/10.1155/2018/8983590>

AUFFARTH, B., Lopez, M., Cerquides, J. (2010). *Comparison of redundancy and relevance measures for feature selection in tissue classification of CT images*. Advances in Data Mining. Applications and Theoretical Aspects. p. 248--262. Springer. <http://www.csc.kth.se/~auffarth/publications/redrel.pdf>

BELLMAN, R., E. (2003). *Dynamic Programming*. Courier Dover Publications. ISBN 978-0-486-42809-3.

BEUTELSPACHER, A.,; ROSENBAUM, U. (1998), *Projective Geometry: From Foundations to Applications*, Cambridge University Press, p. 10, ISBN 9780521483643

DUDA, O. R – HART, P. E. – STROK, G. D. (2001) *Pattern Classification*, 2nd Edition, New York : Wiley, 2001, ed.2, 18s. ISBN 9780471056690

IZZAT, M., A., RAHOUTI, M.,, (2020) "Combining Blockchain and Machine Learning to Forecast Cryptocurrency Prices", Blockchain Computing and Applications (BCCA) 2020 Second International Conference on, pp. 52-58, 2020.

LAKSHMI,S,(2021) N.G. SWADTHI, V., SHREE,"Investigation on Bitcoin Prognosis", Advanced Computing and Communication Systems (ICACCS) 2021 7th International Conference on, vol. 1, pp. 1085-1089, 2021.

KOHAVI R.,F. PROVOST, (1998) "Glossary of terms," Machine Learning, vol. 30, no. 2–3, pp. 271–274, 1998.

KOK, S.H., A. AZWEEN, N.Z., (2020) ,*Evaluation metric for crypto-ransomware detection using machine learning*, Journal of Information Security and Applications, Volume 55,2020,102646, ISSN 2214-2126, <https://doi.org/10.1016/j.jisa.2020.102646>.

NELLO, C., SHAVE, J.,T.(2000). An Introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press, 2000. ISBN 0-521-78019-5

RASMUSSEN, C. E.; WILLIAMS, C. K. I. (2006). "*Gaussian Processes for Machine Learning*".

SAMUEL, A., (1959). "*Some Studies in Machine Learning Using the Game of Checkers*". IBM Journal of Research and Development. **3** (3): 210– 229. CiteSeerX 10.1.1.368.2254. doi:10.1147/rd.33.0210

SERGIOS, Th. (2008). *Pattern Recognition*. Elsevier B.V. p. 203. ISBN 9780080949123.

YERMACK, D. (2015), *Is Bitcoin a Real Currency? An Economic Appraisal*, Editor(s): David Lee Kuo Chuen, Handbook of Digital Currency, Academic Press, Pages 31-43, ISBN 9780128021170, <https://doi.org/10.1016/B978-0-12-802117-0.00002-3>.

Internetes források

DAVDA, (2021) <https://algotrading101.com/learn/binance-python-api-guide/>

TIWARI, M., (2021). "*Shiba Inu is being hailed as Dogecoin killer and it has already given 1900 pct returns in one week*". India Today. New Delhi.