

37. OTDK, Biológia szekció

Gépi tanulási algoritmusok fejlesztése a gének kópiaszám-változásainak pontos azonosítására

Készítette:

Varga Valentin

Biológus MSc

Témavezető:

Prof. Dr. Haracska Lajos

Belső konzulens:

Dr. Laczi Krisztián

Szegedi Tudományegyetem

Kutatóhely:

Deltabio2000 Kft.

Szeged

2024

Tartalomjegyzék

1. Bevezetés	3
2. Célkitűzés:	4
3. Irodalmi áttekintés	5
3.1. Szerkezeti változások a humán genomban	5
3.2. A kópiaszám változások	5
3.2.1. Kópiaszám változások hatásai	6
3.2.2. CNV-k detektálása és azonosítása	7
3.2.3. Szekvenálási technikák	8
3.2.4. Panelszekvenálás által érintett fontos CNV-t tartalmazó gének	8
3.3. A bioinformatika szerepe a strukturális változások és CNV-k azonosításában	9
3.4. Gépi tanulási módszerek	10
4. Anyagok és módszerek	12
4.1. Felhasznált szkriptek	12
4.2. A minták származása és feldolgozása	13
4.3. <i>In silico</i> adatok generálása	13
4.4. A minták kezelése, CNV analízis	14
4.5. A CNV-s adatok feldolgozása és ellenőrzése	14
4.6. Felhasznált gépi tanulási modellek	15
4.6.1. Logisztikus regresszió modell	15
4.6.2. Random Forest modell	16
4.6.3. Gradient Boosting modell	16
4.6.4. XGBoost modell	17
4.6.5. Bagging Classifier modell	17
4.6.6. Support Vector Machine modell	17
4.7. Modellek tanítása és predikció az ismeretlen adatokon	18
4.8. A modellek kiértékeléséhez használt metrikák	18
4.8.1. Modell pontosság	18
4.8.2. Konfúziós mátrix	19
4.8.3. Keresztvalidációs pontosság	19
4.8.4. F1-score	19
4.8.5. Matthews korrelációs koefficiens	20
4.8.6. Receiver Operating Characteristic görbe alatti területe	20
4.8.7. A modellek által felhasznált változók fontossága	21

5. Eredmények és tárgyalásuk	22
5.1. A CNV analízis kiértékelése	22
5.2. Az adatok megfelelőségének ellenőrzése	23
5.3. A modellek kiértékelése.....	24
5.4. A logisztikus regressziós modell változóinak sorrendje.....	27
5.5. A Gradient Boosting modell klasszifikációját befolyásoló változók.....	27
5.6. A Random Forest modell döntési tényezői.....	28
5.7. Az XGBoost modell döntéseit meghatározó változók.....	28
5.8. A Bagging Classifier változóinak fontossági sorrendje.....	29
5.9. Az SVM modell döntéseit meghatározó tényezők sorrendje.....	30
5.10. Predikciók kiértékelése	30
Összefoglalás	32
Irodalomjegyzék	33
Köszönetnyilvánítás.....	Hiba! A könyvjelző nem létezik.
Függelék.....	38
Tételes szerzői hozzájárulás.....	39

1. Bevezetés

A genomban számos változás megy végbe a sejtek élete során. Ilyenek a mutációk is, amelyek általában a bázissorrend megváltozásával járnak, a rekombináció, az epigenetikus változások. Ezek a módosulások fontos szerepet játszanak a genetikai diverzitás fenntartásában. Azonban nem csak pozitív hatásaik lehetnek. Számos betegség, mendeli öröklődésű genomiális rendellenesség és tumoros elváltozás okozói is lehetnek ezek a genetikai modifikációk (Manolio et al., 2009; Shlien & Malkin, 2009). A kópiaszám változások (CNV), amik egyfajta strukturális variánsnak tekinthetők is ide tartoznak. A CNV, amely egy adott genomi régió kópia számának megváltozását jelenti, a genomi variáció legkevésbé ismert típusa, ugyanakkor a humán populációban széles körben elterjedt. A CNV-k érinthetnek egyetlen bázispárt, de akár több gént is. Befolyásolhatják a gének kifejeződésének mértékét és gének funkcióját is. Az ilyen CNV-knek két fő típusa van, nevezetesen a multiplikációk és a deléciók (Iafrate et al., 2004). Duplikációk vagy multiplikációk esetében az adott DNS szakaszból a normálisnál több másolat keletkezik, deléciók esetében pedig ezzel ellentétben kevesebb jön létre. A CNV-eket alapvetően nehéz kimutatni változó méretük, összetett töréspontjaik és a humán genomban található ismétlődő szekvenciák miatt, amelyek megzavarhatják a hagyományos kimutatási módszereket. Az új-generációs szekvenálási módszerek és a nagy áteresztőképességű technológiák térhódítása azonban áttörést jelentettek a CNV-k detektálásában is (Schuster, 2008). Jelenleg a detektálás különböző bioinformatikai szoftverek és pipeline-ok segítségével zajlik (Singh et al., 2021). Ezek az eszközök nem feltétlenül nyújtanak kellő megbízhatóságot és pontosságot (Zhao et al., 2013). A fentebb említett tulajdonságok többek között függenek a kiindulási fájljuktól és adatoktól, az elemző szoftvertől és magától a használt algoritmustól is. Minden algoritmusnak megvannak a maga paraméterei, melyek finomhangolásra szorulnak (W. Li & Olivier, 2013). Ezek mind fontos szerepet játszhatnak a tumordiagnosztikában és a személyre szabott terápiák kialakításában. A pontosabb CNV detektálás érdekében nélkülözhetetlenek a bioinformatikai szoftverek folyamatos fejlesztése, bővítése és a mesterséges intelligencia bevonása a döntéshozásba.

2. Célkitűzés:

A célom több olyan gépi tanulási algoritmust használó modell létrehozása volt, amely képes nagy pontossággal és megbízhatósággal prediktálni ismeretlen mintákban a kópiaszám változásokat.

- Különböző bináris osztályozó modellek létrehozása.
- A modellek összehasonlítása és a legmegfelelőbb kiválasztása.
- Validáció a modellek számára ismeretlen és osztályozatlan adatokon.

3. Irodalmi áttekintés

3.1. Szerkezeti változások a humán genomban

A humán genom folyamatosan változik, mely lehetővé teszi a folyamatos fejlődést és alkalmazkodást (Feuk et al., 2006). A strukturális változatokból kifolyólag számtalan változata lehetséges az emberi örökítőanyagnak. Leggyakoribbak az egy nukleotidot érintő változások (SNP – Single Nucleotide Polymorphism), a legalább 50 bázispár méretű változásokat már strukturális variánsoknak (SV) nevezzük (Sudmant et al., 2015). Emellett jelen vannak még a kópiaszám variációk (CNV), amik inszerciók és deléciók (azaz indelek) formájában találhatók meg. Ezeken kívül ide tartoznak még a mobilis genetikai elemek inszerciói is (Ho et al., 2020) és a nagy kromoszóma rendellenességek is (Shao et al., 2019). A strukturális variáns mérete és gyakorisága nagyban hozzájárul a mutációkhoz, a genom evolúciójához (Weischenfeldt et al., 2013). Emellett fontos résztvevői a csíravonal mutációknak és szomatikus betegségeknek (Beroukheim et al., 2010), mert befolyásolják a sejtben zajló folyamatokat, az örökítőanyag háromdimenziós szerkezetét és a génátírást is (Spielmann et al., 2018). Az egyes változatokat besorolhatjuk kiegyensúlyozott és kiegyensúlyozatlan változásként. Kiegyensúlyozatlan változás esetén (mint például a CNV-k) nyereség vagy veszteség éri az adott DNS szakaszt. Kiegyensúlyozott változás során nem történik mennyiségi változás az örökítőanyagban, ilyen változatok az inverziók és transzlokációk (Sudmant et al., 2015).

3.2. A kópiaszám változások

A genetikai variációk legkevésbé ismert típusai közé tartoznak a kópiaszám-variánsok (CNV-k). Genetikai szerkezeti variációk kiegyensúlyozatlan formái. Általánosságban úgy határozzák meg, mint a humán genom 1 kilobázisos vagy annál nagyobb szakaszának multiplikációja vagy számbeli csökkenése (deléciója) (Sebat et al., 2004). Ezek a szerkezeti eltérések nagyságrendekkel több polimorf bázispárt tartalmaznak, mint az SNP-k (Zarrei et al., 2015). A CNV-k jelentős hatással vannak a génexpresszióra és összefüggésbe hozhatók számos rákos megbetegedés kialakulásával (Budczies et al., 2016; Samulin Erdem et al., 2017). Több vizsgálatot végeztek a CNV-k eredetének feltárására emberben és más modellorganizmusokban egyaránt. Ehhez hibridizációs alapú vagy bioinformatikai módszereket alkalmaztak. Ezzel lehetővé téve a betegségek közötti kapcsolatok és genomi rendellenességek azonosítását. Kiderült az is, hogy ezek a strukturális variánsok nem csak a gén kópiaszámát befolyásolják, hanem a géntermék

menyiségét is. (Henrichsen et al., 2009). Evolúciós és klinikai relevanciájuk ellenére kevésbé tanulmányozzák őket, mint az SNP-ket. Ennek oka valószínűleg a CNV-k kimutatásának nehézsége, ugyanakkor kialakulásuk okait sem ismerjük még teljes mértékben (Lauer & Gresham, 2019).

3.2.1. Kópiaszám változások hatásai

Számos tanulmány bizonyítja, hogy a CNV-k széleskörben elterjedtek a humán genomban és nagymértékben hozzájárulnak genetikai betegségek és a diverzitás fenntartásához (Stankiewicz & Lupski, 2010). A kópiaszámok megváltozása különböző betegségeket okozhat a génexpresszió megváltoztatásával. Számos gyakori betegség összefüggését is kimutatták, ideértve az autoimmun és fertőző betegségeket, neuropszichiátriai és szív- és érrendszeri betegségeket. Emellett összefüggéseket találtak a csíravonalbeli és szomatikus CNV-k és a tumorok kialakulása és életciklusa között is (Fanciulli et al., 2010). A rákot a sejtek növekedését és differenciálódását szabályozó gének expressziós aktivitásának rendellenes szabályozása okozza, ami kóros sejtburjánzáshoz vezet. Ezeket a hatásokat gyakran az onkogének és tumorszuppresszor gének csíravonalbeli vagy szomatikus mutációi közvetítik. A teljes genom szekvenálási technológiák (WGS, WGES) alkalmazása révén a csíravonalbeli és szomatikus CNV-ket szélesebb körben azonosítják, és ma már ismert, hogy hozzájárulnak a rák kialakulását elősegítő mutációk sokféleségéhez (B. Weir et al., 2004). A szomatikus CNV-ket gyakori patogén eseményként azonosították. Egy tüdő adenokarcinómával foglalkozó tanulmányban, amely 528 esetet tartalmazott, összesen 57 ismétlődő kópiaszám-változást mutattak ki. Hasonló eredményeket találtak emlő-, vastagbél- és prosztaták kapcsán is (B. A. Weir et al., 2007). A CNV-k pusztán méretük miatt több nukleotidváltozást okoznak. Érdeemes megjegyezni, hogy a CNV-k által érintett gének 53%-ánál a megfelelő CNV a tényleges gén pozícióján kívül helyezkedik el, ami azt jelenti, hogy sok CNV befolyásolhat fontos szabályozó szekvenciákat is, amelyek távolabb helyezkednek el a célgéntől (Ionita-Laza et al., 2009). A CNV-k számos jellemzője alátámasztja szerepüket a betegség patogenezisében, bár kevésbé gyakoriak, mint az SNP-k. Emiatt van nagy jelentősége a tumordiagnosztikában a CNV-k megfelelő azonosításának és leírásának, mivel pontos azonosításuk elengedhetetlen lehet a megfelelő kezelés kiválasztásához és hatékonyságához. A tumorgenomikát már rutinszerűen alkalmazzák többféle daganatos betegség diagnosztizálásában és kezelésében. A genomikai profilalkotás sztemderd

eljárása a szöveti biopsziák szekvenálása. Ezt megnehezítheti, hogy egyes betegektől nem lehetséges szövetmintát venni, ennek oka lehet késői stádiumú tumor, rossz fizikai állapot, elérhetetlen helyen levő tumor stb. Ezen esetekben a folyékony (liquid) biopsziák szekvenálása egy jó alternatív módszer a genomikai információk megszerzésére rákbetegeknél. Az elmúlt években a keringő tumorsejtek DNS-ét (ctDNA) széles körben használják molekuláris diagnosztikában, kezelési válasz nyomon követésében, valamint a rák kiújulásának és gyógyszerrezisztencia kialakulásának kimutatásában (Peng et al., 2019). A különféle betegségekben betöltött szerepük miatt nélkülözhetetlen, hogy pontosan azonosítani tudjuk a kópiaszám megváltozását.

3.2.2. CNV-k detektálása és azonosítása

A kópiaszám változások szűrése már több, mint egy évtizede része a genetikai diagnosztikus vizsgálatoknak, azonban még mindig sok ismeretlen hatásuk van. Többféle molekuláris és képalkotó vizsgálatokkal is ki lehet mutatni őket. Az első ilyen módszerek a citogenetika és a gél elektroforézis voltak, de néhány év múlva már southern blot-ot és kvantitatív PCR-t is használtak genomi duplikációk vagy deléciók vizsgálatára. A genom léptékű hibridizációs technikák további CNV-s minták azonosítását tették lehetővé (Lauer & Gresham, 2019). Ma a legtöbb esetben új generációs szekvenálási (NGS) technikákkal azonosítják a kópiaszám változásokat. Ezeknek köszönhetően jelentősen megnőtt a vizsgálható minták száma. Azonban a jelenleg használt modern módszerekkel sem lehet olyan megbízhatósággal CNV-t kimutatni, mint SNP-ket (Rajagopalan et al., 2020). Eddig a CNV-k észlelésének alsó határa 50 bázispár volt. Új, nagyobb felbontású módszereknek köszönhetően ez a szám folyamatosan csökken, lehetővé téve a különböző hosszúságú variánsok kimutatását (Pös et al., 2021). A különböző metódusok alkalmazása során kiderült, hogy a CNV-k gyakoriak a humán genomban (Zarrei et al., 2015). A jelenlegi megközelítések főleg panelszekvenálásból vagy exom-szekvenálásból mutatnak ki CNV-ket (Gross et al., 2019). Számos megközelítést dolgoztak ki a CNV-k kimutatására a rövid leolvasásokból. A legtöbb módszer az egy régióra eső leolvasások alapján határozza meg, hogy a referenciagenomhoz képest az adott régióban található-e elváltozás vagy sem. A CNV-k keresésének alapja a leolvasási mélység, vagy "read depth". Ez az eljárás arra épül, hogy a strukturális változatok befolyásolják az leolvasási mélységet, azaz a genom adott részére eső olvasások számát. Egy másik módszer a szekvencialeolvasások speciális leolvasásait használja fel a CNV-k során létrejövő új

töréspontok meghatározásához, mely ezen struktúrális változások "ujjlenyomatai" (L. Chen et al., 2014). Ezek a töréspontok gyakran olyan régiókban fordulnak elő, ahol a DNS repetitív szakaszokat tartalmaz (Hu et al., 2020). Emiatt nehéz CNV-ket találni referenciára való illesztéssel. Az azonosító módszerek másik típusa a „long read sequencing”, ami hosszú leolvasásokat vesz alapul (Marx, 2023). Ezek a módszerek képesek kiküszöbölni a rövid leolvasás alapú metódusok hiányosságait és hibáit, de az előbbi technikához képest az áteresztőképességük alacsony és a költségük magas (Huddleston et al., 2017). Azonban még ezek sem tudják tökéletesen azonosítani a kópiaszám változásokat. Jelenleg egyetlen technológia, módszer és algoritmus sem képes a genomban előforduló struktúrális változatok egészét felismerni. Az NGS alapú módszerek tették lehetővé a kisebb szerkezeti variánsok felfedezését, a microarray-en alapuló metódusok, amiket SNP-k azonosítására használtak csak nagyobb indel-eket tudnak azonosítani (Duan et al., 2013). A CNV vizsgálat nagyban függ a genom összeszerelés minőségétől. A detektáló algoritmusok általában nagy számú hamis pozitív struktúrális variánst azonosítanak. Nincs egy egységes sztenderd, amihez lehet viszonyítani az azonosított változatokat, ezért külön-külön kell vizsgálni és validálni a különböző eseteket (Couldrey et al., 2017).

3.2.3. Szekvenálási technikák

Az újgenerációs szekvenálási (NGS) módszerek nagyobb áteresztőképességgel rendelkeznek az elődeikhez képest, és számos alkalmazásban hatékonyabbak. Az így keletkezett nagymennyiségű adat feldolgozásához új informatikai eszközöket, például illesztő algoritmusokat kellett fejleszteni, hogy minél gyorsabban és pontosabban lehessen feldolgozni a szekvenciákat és összeállítani a genomokat. Az NGS általános metodikájához tartozik a könyvtárkészítés, az ezt követő klonális amplifikáció, végül a bioinformatikai elemzés (Slatko et al., 2018).

3.2.4. Panelszekvenálás által érintett fontos CNV-t tartalmazó gének

Az NGS-szekvenálás megkönnyítette a rákos megbetegedésekkel potenciálisan összefüggésbe hozható gének célirányos vizsgálatát. Bár az emberi genom több mint 20 000 protein kódoló gént tartalmaz, ezekből körülbelül 500 gént azonosítottak, amelyek potenciálisan összefüggésbe hozhatók rákos megbetegedésekkel. Ezen gének között

megtalálhatóak a tumoros sejtek kialakulásában, fenntartásában és terjedésében kulcsfontosságú szerepű gének. Panelszekvenálással csak a tumorok kialakulását befolyásoló géneket szekvenálunk (Nagahashi et al., 2019).

Az általam vizsgált panelekben 54 és 55 gén található, melyek között találunk tumorszuppresszor és proto-onkogén funkciójúakat egyaránt. A tumorszuppresszor gének olyan kulcsfontosságú szabályozók, amelyek szerepet játszanak a sejtek növekedésének ellenőrzésében és a rákos átalakulások elkerülésében. (Lee & Muller, 2010a). Proto-onkogének olyan gének, amelyek normális körülmények között fontos szerepet játszanak a sejtek növekedésében és osztódásában. Ha ezek a gének túlzottan aktívvá válnak, például túltermelődnék, akkor onkogénekké válhatnak (Croce, 2008; Lee & Muller, 2010b). Egyes gének mindkét panelben megtalálhatóak. Annak érdekében, hogy maximalizáljam a mintaszámot és növeljem a gépi tanulási modellek pontosságát, egyidejűleg vizsgáltam a két panelt.

3.3. A bioinformatika szerepe a strukturális változások és CNV-k azonosításában

Az újgenerációs szekvenálás megjelenésével megsokszorozódott a hozzáférhető biológiai adatok mennyisége (Celesti et al., 2018; Kinjo et al., 2018; B. Schmidt & Hildebrandt, 2017). Ezzel lehetőség nyílt arra, hogy minden eddiginél többet tudhassunk meg a genetikáról és a biológiai folyamatokról, mint például a sejtekben zajló folyamatokról vagy gyógyszerek hatásmechanizmusairól. A bioinformatika a biológia és az informatika kombinációjával létrejött tudományág: a bioinformatika eszközeit használja fel a biológiai problémák megoldására. A kópiaszám változások kimutatására számos szoftver és eszköz készült már. Ezek a szekvenálás adataiból nyerik ki a szükséges adatokat, amik legtöbbször a leolvasás mélysége és a lefedettség. Az leolvasás mélysége csak azt mondja meg, hány leolvasás van egy adott pozíción, míg a lefedettség normalizált információt nyújt a genom méretéhez viszonyítva. Az egyik abszolút mennyiséget mér, míg a másik relatív mértéket ad. Ezeket az adatokat teljes genom szekvenálásból, exomszekvenálásból vagy célzott panelszekvenálásból nyerik (Roca et al., 2019). Az NGS szekvenálás megjelenésével szükségessé vált olyan eszközök fejlesztése, ami képes azonosítani a DNS strukturális elváltozásait az NGS adatokból kinyerhető jellemzők alapján. A leggyakrabban használt módszerek közé tartoznak a páros végű térképezés, a „split read” alapú módszerek, a *de novo* összeszerelés és a lefedettségi mélység alapú eszközök. A párosított végű módszerek használata során feltételezzük, hogy az inszertek mérete

meghatározott eloszlású. A CNV-ket a páros olvasások közötti eltérések alapján azonosítják. A „split read” is páros végű szekvenálásból származó leolvasásokat használ. Azonban ezeknek a leolvasásoknak csak az egyik fele térképeződik kellő megbízhatósággal, míg a másik fele csak részben vagy egyáltalán nem térképeződik. A de novo összeszerelés során DNS darabokat állítanak össze, majd ezeket illesztik a referenciagenomra, így azonosítva az eltérő kópiaszámú régiókat. A lefedettség alapú megközelítés a szekvenciára térképeződő leolvasások számán alapszik (Y. Chen et al., 2017; Wu et al., 2012). A legújabb technikák közé tartozik a célzott NGS szekvenálásból érkező adatok elemzése. Az SNP-ket és rövid indeleket nagy pontossággal ki lehet már mutatni, de a nagyobb kópiaszám változások felismerésére még így sincs tökéletes megoldás (Gutiérrez et al., 2023). Több kereskedelmi forgalomban lévő és nyilvános eszköz is elérhető a CNV-k kimutatására (Gabrielaite et al., 2021). Ilyen eszköz például a cn.MOPS, a CODEX2 és a ControlFREEC is (Boeva et al., 2012; Jiang et al., 2018; Klambauer et al., 2012). A legtöbbjük a szekvenálás mélységén alapul, de található olyan, amely páros leolvasásokon, összeszerelésen alapszik. Ezen eszközök WES (whole exome sequencing) vagy WGS (whole genome sequencing) adatokhoz készültek, azaz nem alkalmazhatóak panelszekvenálás eredményeinek elemzésére (Smolander et al., 2021). Ezen eszközök pontossága és robusztussága gyakran nem megfelelő a CNV-k megbízható azonosításához, különösen a genom nagy szekvencia-hasonlósággal vagy ismétlődő elemekkel rendelkező régióiban (Lepkes et al., 2021). A jelenlegi eszközök jól teljesítenek a nagy (megabázis méretű) CNV-k esetében, de nehézségekbe ütköznek az egy- és több-exonos variánsok esetében, amelyek szintén fontos szerepet játszanak számos genetikai betegség és tumor kialakulásában. A legtöbb eszköz specifikus, és a teljesítményük nagyban függ a használt adatokból (Moreno-Cabrera et al., 2020). Ezeknek a problémáknak a megoldására különböző gépi tanulási módszerek fejleszthetők.

3.4. Gépi tanulási módszerek

A gépi tanulás a mesterséges intelligencián (Artificial Intelligence - AI) és a számítástechnikán belüli tudományág, amely azzal foglalkozik, hogyan tanulnak a számítógépes rendszerek az adatokból, és hogyan fejleszthetik folyamatosan önmagukat. A gépi tanulási modellek nagymértékben a valószínűségszámításon és a statisztikán alapulnak. A modelleket nagy adathalmazokon tanítják be, hogy mintázatok és összefüggéseket találjanak a lehető legjobb döntések és előrejelzések meghozatala

érdekében. Az elmúlt években a keletkezett adatok jelentős növekedése miatt a gépi tanulási rendszerek egyre hatékonyabbá váltak (Strzelecki & Badura, 2022). Tekintettel a biológiai adatok hirtelen megugrására, amelyet az új generációs szekvenálás fejlődése és annak csökkenő költségei ösztönöznek, a gépi tanulás és a "mély tanulási" modellek sok esetben alkalmazhatók a biológiában és az orvostudományban is (Özgür & Orman, 2023). Az ilyen rendszerek beépítése az orvostudományba lehetőséget ad az orvosoknak és a klinikusoknak olyan problémák megoldására, amelyek egyébként fáradságosak, időigényesek és kézi módszerekkel nehezen megvalósíthatók. Jelenleg a gépi tanulási algoritmusok az orvosi képzőképzés (Liu et al., 2022), az eddig ismeretlen jelentőségű variánsok osztályozása (A. Schmidt et al., 2023), az amyotrófiás laterálszklerózis (ALS) kockázatának előrejelzése (Loh et al., 2020), a mikrobiom (P. Li et al., 2022) területén találnak alkalmazást. A biológiában széles körben alkalmazott gépi tanulási modellek három fő kategóriába sorolhatók: felügyelt tanulás, felügyelet nélküli tanulás és megerősítéses tanulás. A felügyelt tanulás során a modelleket egy tanuló adathalmazon tanítják be. A tanítási folyamat részeként a modell megtanulja a jellemzők hozzárendelését a címkékhez, és miután betanították, a jellemzőkből meg tudja jósolni az új adatok osztályát (Burkart & Huber, 2021; Jovel & Greiner, 2021). Például a felügyelt tanulási modellek képesek meghatározni, hogy egy genetikai variáció patogén vagy jóindulatú-e (Qorri et al., 2022). A dolgozatomban csak a felügyelt tanulási típusba tartozó modelleket készítettem. Ebbe a kategóriába tartoznak a logisztikus regressziós algoritmusok, a döntési fák, a Support Vector Machine és még számos más is (Nasteski, 2017). Az általam vizsgált problémára és adattípusra ezek az algoritmusok nyújthatják a legmegfelelőbb megoldást. A felügyelet nélküli tanulási modellek nem a címkékre, hanem a jellemzőkre támaszkodnak, és képesek a tudás kinyerésére a mintázatfelismerés folyamatán keresztül. A felügyelet nélküli tanulási modellek előnye azonban, hogy nem befolyásolják őket a bemeneti adatok pontatlan címkézéséből eredő torzítások, de nagy hátrányuk, hogy nem lehet tesztelni a teljesítményüket, mivel a "valódi osztály" ismeretlen (Jovel & Greiner, 2021; Usama et al., 2019). A megerősített tanulás olyan megközelítés, amely abban különbözik a felügyelt és a felügyelet nélküli tanulási modellektől, hogy a képzési folyamathoz nincs szükség ember által előállított adatokra. Az ilyen modellek próbálgatással és hiba útján képesek tanulni, ahol az algoritmus visszajelzést kap az elemzésből, és jutalmat kap a kitűzött cél felé vezető műveletekért, és büntetést azokért, amelyek nem járulnak hozzá a cél eléréséhez (Jovel & Greiner, 2021;

Shinde & Shah, 2018). Összességében a gépi tanuláson alapuló megközelítések hatékony megoldást jelentenek a különböző összetett osztályozási folyamatok megoldására.

4. Anyagok és módszerek

A CNV analízis egy kiváló osztályozási probléma lehet egyes gépi tanulási modellek számára. Egy bináris osztályozó modell segítségével meg lehet határozni a minták karakterisztikái alapján, hogy a vizsgált gén vagy annak exonja kópiaszám változáson esett-e át, vagy sem. A jelenleg használt módszer a CNV-k detektálására manuálisan történik. A pipeline-ból érkező kimeneti ábrákat megvizsgáljuk és ha a használt szkript alapján meghatározott küszöbérték felett vagy alatt van a vizsgált exon értéke, akkor kópiaszám változásnak tekintjük. A log2fold érték egy arányszám, amely a gének és exonok lefedettségi értékeinek relatív változását mutatja meg 2-es alapú logaritmikus skálán (Love et al., 2014). A vizsgálatainkban a 2 és -2 küszöbértéket átlépő értéket tekintjük valósnak.

$$1. \quad \log_2 \text{fold change} = \frac{\log_2(\text{vizsgált minta értéke})}{\log_2(\text{referencia minta értéke})}$$

A kiértékelés emberi erőforrásokkal történik, amit felválthatunk egy olyan algoritmus segítségével, ami a megfelelő tulajdonságokat vizsgálva el tudja dönteni a vizsgált génekről és exonokról, hogy megváltozott-e az örökítőanyag kópiaszáma vagy sem.

A szekvenálásból érkezett adatok feldolgozásához számos szoftverre van szükség. Ezek szabadon hozzáférhetőek és elérhetőek az interneten. Az adatok feldolgozása és osztályozása egy Linux operációs rendszerű számítógépen történt, ami az alábbi specifikációkkal rendelkezik: Intel® Core™ i7-8700 12 magos CPU, 64 GB RAM és integrált Intel® UHD Graphics 63. A táblázatok kezeléséhez Microsoft Excel-t (Microsoft Corporation, Redmond, WA, USA) és OpenOffice LibreCalc-ot (The Document Foundation, Berlin, Germany) használtam. Az analízishez és az osztályozáshoz készült szkriptek is ezen az operációs rendszeren készültek.

4.1. Felhasznált szkriptek

Az adatelemzést, a modellkészítést és a tesztelést Python3-ban írt egyéni függvények segítségével végeztem (Van Rossum & Drake, 2009). A szkript automatizálja a fájlok beolvasásának, az eszközök elérésének és az elemzés elvégzéséhez szükséges parancsok futtatásának folyamatát. A szkriptek a függelékben található hivatkozáson érhetőek el.

4.2. A minták származása és feldolgozása

A felhasznált minták többsége emberi eredetű. Annak érdekében, hogy minél több valódi pozitív mintát kapjunk, két azonos szekvenálási panel, nevezetesen az All-Cancer-54 és az All-Cancer-55 eredményeit használtuk fel. Az All-Cancer-54-es panel 54 gént, míg az All-Cancer-55-ös panel 55 gént tartalmaz, és az Illumina NextSeq 500-as platformon szekvenálták őket, ugyanazon protokollt alkalmazva. A panelekben előforduló gének listája a függelékben található. A könyvtárakat Shotgun technikával készítették. A többi mintát in-silico generáltuk az All-Cancer-54-es panel mintáinak lefedettségi fájljaiból.

4.3. In silico adatok generálása

A természetes minták analízise után kiderült, hogy nincs elég valós pozitív minta a modellek megfelelő működéséhez. Ezen kiegyensúlyozatlanság következtében az összes generált modell túlilleszkedett a negatív mintákhoz. Ennek a problémának a megoldására in-silico mintákat generáltam, amelyek ugyanazokat a géneket tartalmazzák, mint a természetes minták. A természetes minták exonjaihoz tartozó lefedettség értékek mediánját vettem alapul a generált minták készítésekor. Minden mintában a panelből csak egy gént generáltam kópiaszám változással (1. táblázat). Mindegyikből öt különböző mesterséges minta készült 10%, 20%, 30%, 40% és 50%-os tumorarányt modellezve. A tumorarány azt fejezi ki, hogy egy adott mintának mekkora része tumoros szövet vagy sejt. Jelen esetben a CNV nagyságára van kihatással. Összesen 85 darab *in silico* generált minta készült. Így már megközelítőleg 23% a mesterségesen elkészített minták aránya a természeteshez képest. A megfelelő arány megtalálása fontos, hogy a biológiai változatosság ne vesszen el a mesterségesen előállított adatok miatt és képesek legyenek a modellek a megfelelő előrejelzésre.

1. táblázat: Az egyes generált minták CNV-s génje és az elváltozás típusát és mértékét tartalmazó táblázat.

Gén	Amplifikáció/Deléción	Szorzó	Gén	Amplifikáció/Deléción	Szorzó
BRCA2	Deléción	5	MYC	Amplifikáción	2
CDK4	Amplifikáción	2	NRAS	Amplifikáción	6
CDKN2A	Deléción	6	PDGFRA	Amplifikáción	2
EGFR	Amplifikáción	2	PIK3CA	Amplifikáción	3
ERBB2	Amplifikáción	2	PTEN	Deléción	7
FGFR1	Amplifikáción	2	RB1	Deléción	9
JAK2	Amplifikáción	5	STK11	Deléción	7
KDR	Amplifikáción	8	TP53	Deléción	9
KRAS	Amplifikáción	2			

4.4. A minták kezelése, CNV analízis

A CNV analízist végző szkript a szekvenciaadatok feldolgozását végzi el, mely az egyes függelékben megtalálható. Moduláris felépítésű algoritmus, kezeli a szekvenciafájlokat, illeszti a leolvasásokat, eltávolítja a duplikátumokat és elvégzi a CNV azonosítást is. A szkript a CNV-elemzés kiértékeléséhez szükséges ábrákat is elkészíti.

A minták feldolgozása: A szkript létrehozza a szükséges mappákat, valamint konkatenálja az input fájlokat és levágja a nem megfelelő minőségű bázisokat a leolvasásokról. A trimmelt leolvasásokat egy referencia genomhoz illeszti, így létrejön mintánként egy-egy BAM fájl, majd utána eltávolítja ezekből a duplikátumokat.

BAM fájlok indexelése és összegyűjtése: A szkript ezen szakasza indexfájlokat (BAI) hoz létre a duplikátum mentes BAM-fájlokhoz, majd az összes BAM- és BAI-fájlt egyetlen mappába helyezi a további elemzéshez.

CNV-elemzés: A szkript a BAM-fájlok felhasználásával CNV-elemzést végez, és különböző kimeneti fájlokat generál. Továbbá elvégzi az adatok statisztikáinak kiszámítását és ábrák készítését is.

4.5. A CNV-s adatok feldolgozása és ellenőrzése

A klasszifikációt végző szkript bemeneti fájljai az analízisből érkező `region_coverage_summary.txt` és a `region_chromosome_summary.txt` nevű fájlok. Az algoritmus a bemeneti táblázatokból összegyűjti a modellek tanításához szükséges jellemzőket egy Pandas (McKinney, 2011; The pandas development team, 2020) adatkeretbe. Ebben az adatkeretben minden minta, minden génjének minden exonja szerepel és az ezekhez tartozó lefedettségi érték, $\log_2$ fold change érték, az exon méret, a csúszó ablak (sliding window) átlag és a kromoszómára eső összes leolvasás érték. A sliding window átlagot az egymás mellett lévő exonok $\log_2$ fold change értékeinek átlaga határozza meg. Az „ablak” a vizsgált exontól mindkét irányban 3 távolságra lévő exonokat veszi figyelembe. Emellett szerepel még az információ is, hogy az adott exonon találtunk-e CNV-s elváltozást, melyet a `TPlist.csv` táblázatból olvassa be az algoritmus, majd binárisan tárolja az adatkeretben. Az adatok alkalmasságát egyszempontos varianciaanalízissel (ANOVA) és az adatok hisztogramon történő ábrázolásával ellenőriztem. A varianciaanalízis feltétele, hogy az adatok normális eloszlást követnek. Az ANOVA (Analysis of Variance) egy statisztikai teszt, amelyet több minta közötti különbségek elemzésére használnak (Kaufmann & Schering, 2014). Az ANOVA segítségével megvizsgálható, hogy a csoportok között van-e szignifikáns

különbség. Az ANOVA teszthez a SciPy modul (Virtanen et al., 2020) `f_oneway` függvényét használtam. Kiválogattam az összes pozitív jellel ellátott minta $\log_2$ fold értékét és ugyanennyi negatív jellel rendelkező mintát is véletlenszerű mintavételezéssel.

4.6. Felhasznált gépi tanulási modellek

A legmegbízhatóbb osztályozó megtalálásának érdekében több modellt is készítettem. A modellek ugyanazokat a változókat használták fel a tanuláshoz. Ezek az X változóba kerültek: coverage érték, $\log_2$ fold érték, sliding window átlag, exon méret és a kromoszómára eső leolvasások száma. Y változóba pedig a változókhoz tartozó bináris jel került, ami megmutatja, hogy az adott exonban detektáltunk-e duplikációt vagy deléciót. Így a modellek tanításához használt negatív és pozitív minták aránya finomhangolás után 3:1 lett. A tanuló és teszt adatok felosztása 80:20 arányban történt a Scikit-learn (Pedregosa et al., 2011) `TrainTestSplit` függvényével. A modellek tanítása után az egyes változók befolyását is meghatároztam. Összesen így 1340 pozitívnak jelölt exont és 4020 negatívnak jelöltet használtam fel a modellépítésre.

4.6.1. Logisztikus regresszió modell

A logisztikus regressziós modell létrehozásához a Scikit-learn csomagot használtam. A logisztikus regresszió egy olyan statisztikai módszer, amelyet a bináris függő változó és egy vagy több független változó közötti kapcsolat elemzésére használnak. A módszer a lineáris regresszió kiterjesztése, és azért alkalmazzák, mert lehetővé teszi, hogy előrejelzéseket tegyünk vagy megmagyarázzuk a bináris kimeneti változó valószínűségét a független változók (magyarázó változók) értékei alapján (Cox, 1958). Az adatokat a Scikit-learn `StandardScaler` moduljával átalakítottam a pontosabb eredmény érdekében. Ez a módszer az adatokat úgy transzformálja, hogy azok átlaga 0, és szórása 1 legyen. Az adatokat az alábbi képlet alapján számolja át:

$$2. \text{ Scaled érték} = \frac{(\text{Adat értéke} - \text{átlag})}{\text{Szórás}}$$

Scaling után SMOTE-t (Synthetic Minority Over-sampling Technique) használtam, szintén a Scikit-learn modulból. A SMOTE egy olyan eljárás, ami az osztályok közötti egyensúlyt változtatja meg annak érdekében, hogy az osztályozás pontosabb legyen (Chawla et al., 2002). A modell paramétereinek beállításához a Scikit-learn `GridSearchCV` modulját használtam. Ez a függvény a megadott kombinációkat

végigiterálja, keresztvalidációval ellenőrzi és kiválasztja a legjobb metrikákkal rendelkező modellt, majd arra illeszti fel a tanulóhalmaz adatait (Hastie et al., 2009). A keresztvalidáció lényege, hogy csökkentjük az overfitting (túltanulás) jelenségét, ami rontja a modellünk teljesítményét (LaValle et al., 2004). A paraméterek az alábbiak voltak:

```
Best parameters: {'C': 0.01, 'dual': False, 'fit_intercept': True,
                  'max_iter': 100, 'penalty': 'l1', 'solver': 'saga', 'tol': 0.01,
                  'warm_start': True}
```

4.6.2. Random Forest modell

Ez a modell a Scikit-learn modul RandomForestClassifier algoritmusával készült. A Random Forest egy gyűjtemény vagy erdő döntési fákból, amelyeket a bemeneti adathalmazból véletlenszerűen választanak ki. Ezeket a fákat különböző részhalmazokban tanítják, ezáltal csökkentve a modell overfittingre való érzékenységet és növelve a zajos adatokkal szembeni ellenálló képességét. (Liaw & Wiener, 2002). A modell az alábbi paraméterekkel futott:

```
Best RF parameters: {'bootstrap': False, 'max_depth': None,
                     'min_samples_leaf': 1, 'min_samples_split': 5, 'n_estimators': 200}
```

4.6.3. Gradient Boosting modell

A modellt a Scikit-learn modul GradientBoostingClassifier függvényével készítettem. A gradient boosting egy döntési fa alapú modell. Több gyenge döntési fát kapcsol egybe így létrehozva egy jól teljesítő modellt. A módszer során a modell lépésről lépésre épül fel. Az első tanuló (alaptanuló) általában egy egyszerű modell, amelyet az adathalmazon tanítunk be. Ezután a hibák (azaz a különbség a predikció és a valós eredmény között) alapján tanítjuk meg a következő gyenge tanuló modellt. A következő algoritmus célja, hogy az előző gyenge tanuló algoritmus által elkövetett hibákat kijavítsa. Ezen a gyengébb modellek eredményeit összegzi és egy újabb modellt hoz létre. Ez addig ismétlődik, amíg a teljesítmény el nem éri a megfelelő szintet (Friedman, 2001). A modell illesztése előtt keresztvalidációt alkalmaztam. Az illesztési paraméterként az alapbeállításokat használtam, mert ez a modell nem támogatta a GridSearch algoritmust. Ezek a következők voltak:

```
Parameters for GB: {learning_rate=0.1, loss='deviance', max_depth=3,
                    max_leaf_nodes=None, min_samples_leaf=1, min_samples_split=2,
                    n_estimators=100, presort='auto', random_state=42, subsample=1.0,
                    validation_fraction=0.1}
```

4.6.4. XGBoost modell

Az XGBoost (eXtreme Gradient Boosting) egy hatékony nyílt forráskódú gépi tanulási algoritmus, amely a gradient boosting technikára épül. Az XGBoost kiváló prediktív képességeket nyújt számos gépi tanulási feladatban, beleértve a klasszifikációt, regressziót, rangsorolást és idősorozatok elemzését. Ez a modelltípus döntési fákra épül, több gyengébb fát készít el, majd azokat kombinálva hoz létre egy jobbat (T. Chen & Guestrin, 2016). Az algoritmusom az xgboost modult használja a modell létrehozásához. Az alábbi paramétereket használta fel a tanuláshoz:

```
Best XGB parameters: {'colsample_bytree': 0.8, 'learning_rate': 0.2,
                      'max_depth': 4, 'n_estimators': 300, 'subsample': 1.0}
```

4.6.5. Bagging Classifier modell

A Bagging Classifier a „zsákoló” (bagging) technikára épül. Lényege, hogy több, gyakran hasonló, de véletlenszerűen kiválasztott modellt kombinál egy „csomagba”, hogy egy erősebb és stabilabb modellt hozzon létre (Breiman, 1996). Az osztályozó algoritmus a Scikit-learn BaggingClassifier függvényében található. A modell a következő paramétereket alkalmazta:

```
Best parameters Bagging Classifier: {'bootstrap_features': True,
                                     'max_samples': 1.0, 'n_estimators': 100}
```

4.6.6. Support Vector Machine modell

Az algoritmus célja, egy határfelület megtalálása úgy, hogy megfelelően szétválassza a különböző osztályhoz tartozó adatokat. Az SVM működése során a tanító adathalmazból olyan támogató vektorokat (support vector) keres, amelyek a két osztály közötti határfelületre esnek, és legközelebb állnak egymáshoz. Ezek a támogató vektorok határozzák meg a határfelület pozícióját és orientációját. Az SVM arra törekszik, hogy az osztályok közötti távolságot, az úgynevezett "margin" távolságot, maximalizálja, amely azt jelenti, hogy a lehető legtöbb helyen biztosítja a két osztály közötti szeparációt (Cortes & Vapnik, 1995). Az SVM modell is a Scikit-learn modulból érhető el. A modell létrehozásához GridSearch-t is alkalmaztam. Az alábbi paraméterekkel készült el:

```
SVM parameters: {'kernel': 'linear', 'gamma': 'auto', 'C': 0.1,
                 'random_state': 42, 'verbose': 'False'}
```

4.7. Modellek tanítása és predikció az ismeretlen adatokon

A modellek illesztéséhez szükséges paraméterek beállításához (ahol az algoritmus lehetővé tette) GridSearch-öt alkalmaztam. Ahol ez nem volt lehetséges, ott az alapbeállításokon történt az tanítás. A modellek elkészülte után a Joblib modul segítségével mentettem el (Joblib Development Team, 2020). Ezután az elmentett „joblib” kiterjesztésű modelleket betölti az algoritmus, majd használja őket predikcióra olyan szekvenálásból érkező humán mintákon, amik nem voltak benne a tanító adathalmazban. Összesen 56 darab mintán futtattam előrejelzést. Ezek között volt 9 darab All-Cancer-54 es panelhez tartozó minta és 47 darab All-Cancer-55-ös.

4.8. A modellek kiértékeléséhez használt metrikák

A modellek teljesítményének meghatározására különböző metrikákat alkalmaztam. Mindegyik modell ugyanazokat a bemeneti adatokat kapta és ugyanazokat a teljesítménymérő algoritmusokat használtam. A metrikák értéke általában 0 és 1 közötti szám és minél nagyobbak, annál jobbnak tekinthető a modell. Ezeket az értékeket százalékos formában használtam a kiértékeléshez.

4.8.1. Modell pontosság

A pontossági (accuracy) érték a modellek egyik legfontosabb mérőszáma, de önmagában korántsem elég a teljesítmény teljeskörű leírására. Az osztályozó modell azon képességét méri, hogy mennyire pontosan képes prediktálni a különböző osztályokhoz tartozó mintákat. Ez az érték az osztályozási modellek által helyesen azonosított minták arányát mutatja az összes mintához képest. Minél magasabb ez az érték, annál jobb a modell teljesítménye, ha azonban az osztályok eloszlása jelentősen eltér, és az egyik osztály sokkal gyakoribb, mint a másik, akkor egy modell, amely mindig a gyakoribb osztályt jósolja meg, magas accuracy score-t kaphat, de valójában nem teljesít jól a feladatban (García et al., 2009; Sokolova et al., 2006). Ezért fontos más metrikák, alkalmazása is. A kalkulációt a Scikit-learn `accuracy_score` metódusa végzi. A pontosság kiszámításának képlete:

$$3. \text{ Pontosság \%} = \frac{\text{Helyes osztályzatok száma}}{\text{Összes minta száma}} \cdot 100$$

4.8.2. Konfúziós mátrix

A konfúziós mátrix (CM) egy kétdimenziós mátrix, ami tartalmazza a valós és hamis pozitív/negatív értékek számát. Tovább számolhatók belőle más metrikák (Kulkarni et al., 2020). Ilyenek a szenzitivitás, specificitás, F1-score és a Matthews korrelációs koefficiens is.

4.8.3. Keresztvalidációs pontosság

A keresztvalidáció (Cross validation) egy olyan technika, amely segít megbecsülni a modell által általánosított teljesítményt, és csökkenti a túltanulás (overfitting) kockázatát. Célja, hogy objektív és megbízható értékelést adjon a modellről. Az adataimon minden modell esetében KFold keresztvalidációt alkalmaztam, ami azt jelenti, hogy a tanítóadathalmazt öt egyenlő részre osztja fel az algoritmus. Ezek közül négy rész lesz a tanítóhalmaz, míg az ötödik a teszhalmaz. Ezután az algoritmus ötször lefut és minden alkalommal másik részhalmazon hajtja végre a tesztelést. A 5-k-szoros keresztvalidáció lehetővé teszi, hogy a modell megbízhatóbb és általánosíthatóbb teljesítményértékelést kapjon, mivel minden adatpont szerepel legalább egyszer a teszhalmazban. Az ötszöri kiértékelés átlagolása segít csökkenteni a modellezési folyamat változékonyságát, és csökkenti a tesztadathalmaz kiválasztásának érzékenységet. Ezáltal az eljárás lehetővé teszi, hogy jobb rálátást nyerjünk a modell viselkedésére, és csökkentsük az overfitting kockázatát (Marcot & Hanea, 2021). Minél magasabb ez az érték, annál jobb a modell. A keresztvalidációt a Scikit-learn `cross_val_score` függvénye hajtja végre.

4.8.4. F1-score

Az F1-score egy olyan metrika, amely az osztályozó modell teljesítményét értékeli pontosság (precision) és érzékenység (recall) metrikák kombinálásával. Különösen hasznos olyan esetekben, amikor az osztályok egyike jelentősen túlsúlyban van a másikkal szemben. A precision megmutatja, azoknak a pozitívrá osztályozott mintáknak az arányát, amelyek valóban a pozitív osztályba tartoznak az összes olyan minta között, amelyeket a modell pozitívnak ítélt. A Recall megadja azoknak a pozitívan osztályozott mintáknak az arányát, amelyeket a modell helyesen osztályozott a pozitív osztályba az összes valódi pozitív osztályú minta között. A precision növelése gyakran a recall csökkenéséhez vezet, és fordítva: míg a precision nagyobb értéke azt jelzi, hogy kevesebb hamis pozitív eredményt ad a modell, addig a recall nagyobb értéke azt jelzi, hogy

kevesebb hamis negatív eredmény keletkezik. Az F1-score, az egyensúlyt találja meg a precision és recall között. Az F1-score értéke 0 és 1 között változhat, ahol 1 a legjobb és azt jelenti, hogy a modell tökéletesen kezeli a precision és recall egyensúlyát (Yacouby & Axman, 2020). Az F1-score értékét a Scikit-learn `f1_score` algoritmus végzi el. Az alábbi képletek alapján történik a számítás:

$$4. \text{ Precision} = \frac{\text{Valós pozitívok száma}}{\text{Valós pozitívok száma} + \text{Hamis pozitívok száma}}$$

$$5. \text{ Recall} = \frac{\text{Valós pozitívok száma}}{\text{Valós pozitívok száma} + \text{Hamis negatívok száma}}$$

$$6. \text{ F1 score} = \frac{2 \cdot (\text{Precision} \cdot \text{Recall})}{\text{Precision} + \text{Recall}}$$

4.8.5. Matthews korrelációs koefficiens

A Matthews korrelációs együttható (Matthews Correlation Coefficient, MCC) egy olyan metrika a bináris klasszifikációs modellek értékelésére, amely az osztályozó modellek teljesítményét méri, különösen akkor, ha az osztályok egyensúlya és a hamis osztályozások jelentősége is számít. A MCC az előrejelzett osztályok és a valós osztályok alapján számítható ki. Ezek a valós pozitív (TP), valós negatív (TN), hamis pozitív (FP) és hamis negatív (FN) értékeket. A szkriptben a Scikit-learn csomagban található `matthews_corrcoeff` függvénnyel történik a meghatározásuk. Értéke -1 és 1 között változhat és minél nagyobb, annál jobb a modell teljesítménye (Chicco & Jurman, 2020). A jól teljesítő algoritmusok MCC-értékei 0,7 és 1 között mozognak, és 0,5 közelében az előrejelzéseik véletlenszerűek, a tökéletes osztályozó algoritmusok pedig 1-es MCC-értékkel rendelkeznek (Chicco et al., 2021). Ha a MCC negatív értéket vesz fel, az csak azt jelzi, hogy a változók között fordított arányú kapcsolat van, de nem feltétlenül van jelen ok-okozati összefüggés. Kiszámításának képlete a következő:

$$7. \text{ MCC} = \frac{(TP \cdot TN - FP \cdot FN)}{\sqrt{(TP + FP) \cdot (TP + FN) \cdot (TN + FP) \cdot (TN + FN)}}$$

4.8.6. Receiver Operating Characteristic görbe alatti területe

A ROC-AUC (Receiver Operating Characteristic - Area Under the Curve) egy olyan metrika a bináris klasszifikációs modellek értékelésére, amely az osztályozó modell teljesítményét méri a hamis pozitív arány (FPR - specificitás) és a valós pozitív arány (TPR - szenzitivitás) összefüggésében. Az ROC görbe egy olyan grafikon, amely az

osztályozó modell különböző döntési küszöbértékei mellett ábrázolja a FPR és TPR értékeket. A TPR azt mutatja meg, hogy az osztályozó modell mennyire képes helyesen azonosítani a pozitív osztályba tartozó mintákat a valódi pozitív osztályú minták közül. A FPR pedig azt mutatja meg, hogy mennyire tévesen osztályozza a modell a negatív osztályba tartozó mintákat a valódi negatív osztályú minták között (Hajian-Tilaki, 2013). Képletei:

$$8. \text{ Szenzitivitás} = \frac{TP}{(TP+FN)}$$

$$9. \text{ Specificitás} = \frac{TN}{(TN+FP)}$$

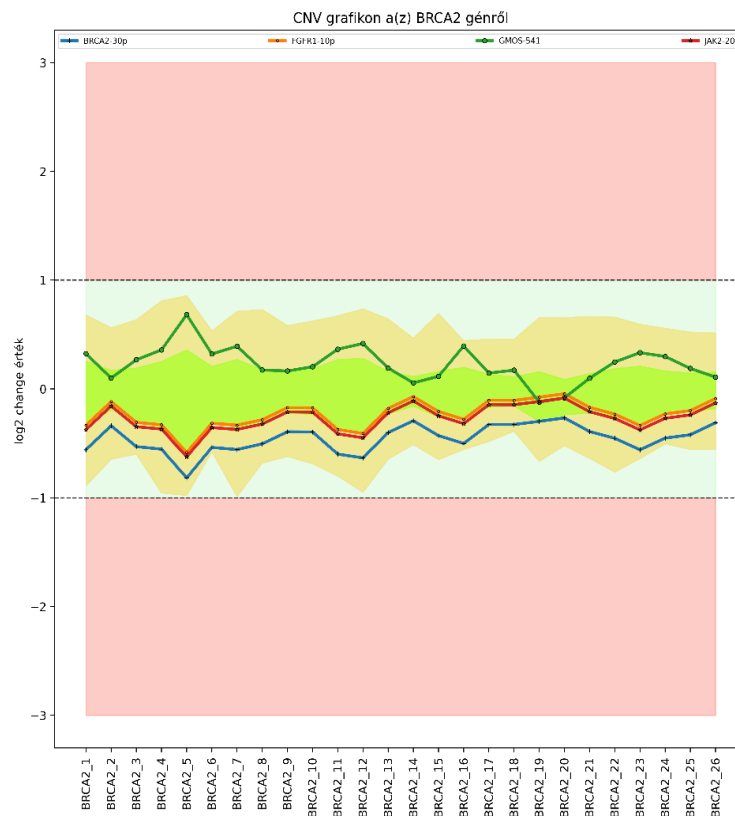
4.8.7. A modellek által felhasznált változók fontossága

A változók befolyásának vizsgálatára a SHAP (SHapley Additive exPlanations) python csomagot használtam (Lundberg & Lee, 2017). A legtöbb modellhez az alap modellmagyarázó funkcióját, míg a fa alapú modellekhez az erre a célra kifejlesztett magyarázó algoritmust használtam (Lundberg et al., 2020). A méhraj ábrákat is ezzel a csomaggal készítettem. Ezeken az ábrákon a színskála mutatja az adott változó értékét. Az x tengelyen az egyes minták SHAP értéke látható, míg az y tengelyen az egyes változók. Ha a SHAP érték pozitív, akkor a modell a pozitív osztályba soroláshoz használta fel az adott értéket, ha pedig negatív, akkor a negatívba. Az egyes változók befolyásának pontos értékei a függelékben található 6. táblázatban megtalálható. Ez alól a Bagging Classifier modell kivétel, mert azt jelenleg nem támogatja a SHAP csomag. Ennek az ábrázolásához a változók fontosságát oszlopdiagramon ábrázoltam. A SHAP-értékek részletesen leírják, hogy az egyes jellemzők milyen mértékben járulnak hozzá az egyes predikciókhoz. Ezen kívül kifejezik, hogy az egyes jellemzők mennyire fontosak az összes többihez képest, és rávilágítanak arra, hogy a modell mennyire támaszkodik a jellemzők közötti kölcsönhatásokra.

5. Eredmények és tárgyalásuk

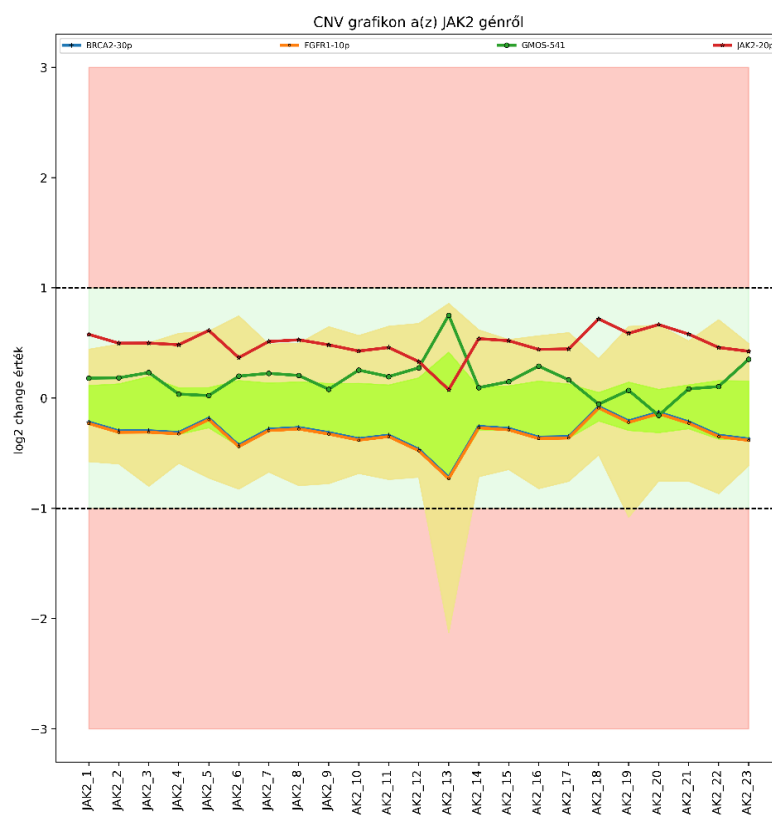
5.1. A CNV analízis kiértékelése

A sikeres CNV analízis után megvizsgáltam a készült ábrákat, hogy a generált CNV-k észlelhetőek-e a jelenleg használt módszerrel. A diagram a $\log_2$ fold érték változását (x tengely) ábrázolja egy génen belül minden exonra (y tengely). Az 1 és -1 közé eső $\log_2$ fold értékeket tekintjük normálisnak, 2 feletti és -2 alatti értékeket biztosan CNV-snek tekintjük. Az ábrázolást végző algoritmus megkeresi az 5., 25., 75. és 95. percentiliseket, majd ez alapján színezi sávosan a diagramot (5. és 95. között az értékek "khaki", 25. és 75. között pedig "zöldessárga" színnel vannak feltüntetve). A sávok kalkulációjához minden minta értékeit felhasználja. A láthatóság érdekében csak néhány mintát ábrázoltam. Az 1. és 2. ábrán is látható természetes minta (GMOS-541 – zöld vonal), ami validáltan nem tartalmaz CNV-s elváltozást és kontroll szerepét tölti be. Az 1. és 2. ábra is jól szemlélteti, hogy a kontrollhoz képest a generált CNV-s minta az 1. ábra esetén a kék vonallal jelölt deléció mint $\log_2$ fold change értékei alacsonyabbak a többi mintához képest deléció mértékének megfelelően. A 2. ábrán a JAK2-20p-val jelölt generált minta ugyan csak 20%-os tumoraránnal rendelkezik, így is jelentősen magasabb $\log_2$ fold change értékekkel rendelkezik. Azonban az is megfigyelhető, hogy ezek ugyan



1. ábra: A BRCA2 gén $\log_2$ fold change értékei exononként.

CNV-t tartalmazó gének, mégis a normális tartományban találhatók. Ezért is van szükség a gépi tanulási modellekre, hogy ezeket a rejtett kópiaszámváltozásokat is detektálni tudjuk.



2. ábra: A JAK2 gén $\log_2$ fold change értékei exononként.

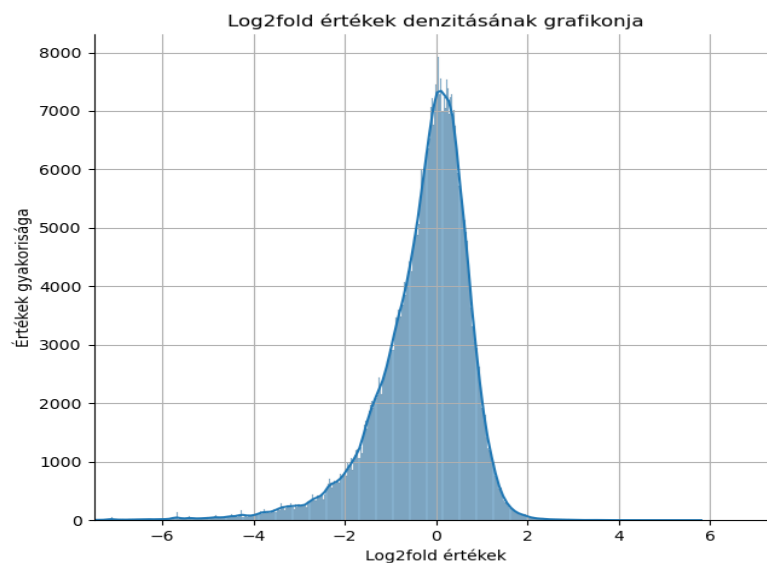
5.2. Az adatok megfelelőségének ellenőrzése

Mielőtt a gépi tanulási modelleket betanítottam, elvégeztem az egyszempontos varianciaanalízist annak érdekében, hogy a $\log_2$ fold értékek biztosan megfelelőek az osztályozó algoritmusoknak. Amint az a 2. táblázatban látható, hogy az ANOVA többszöri vizsgálatra is nagyon alacsony p-értéket (valószínűségi érték) kapott. Minden esetben kisebb volt, mint a 0,05-ös szignifikanciaszint. Az alacsony p-értékek azt jelzik, hogy a csoportok közötti különbségek informatívak az osztályozási algoritmusok számára.

2. táblázat: Az ANOVA vizsgálat eredményei.

Ciklus	p-érték
1	2.91724449046779E-66
2	3.76254154531302E-66
3	3.95854622657523E-68
4	4.90721299730774E-68
5	5.86657029340711E-70

Emellett egy hisztogrammot is készítettem, ami az ANOVA vizsgálathoz felhasznált adatok eloszlását mutatja (3. ábra). A hisztogram az adatok normál eloszlását mutatja, enyhe balra ferdeséggel. A legtöbb érték 2 és -2 között található, amelyek jól reprezentálják azokat az exonokat, amelyek nem estek át kópiaszám változáson. Az -2 alatti értékek a deléciós változásokat, míg a 2 feletti értékek a duplikációkat jelzik. A magasabb $\log_2\text{fold}$ értékek pedig a generált minták sokszoros multiplikációját és a természetes mintákban található nagy CNV-ket reprezentálják. Kijelenthető, hogy az adatok megfelelnek a modelleknek és alkalmasak az osztályozásra.

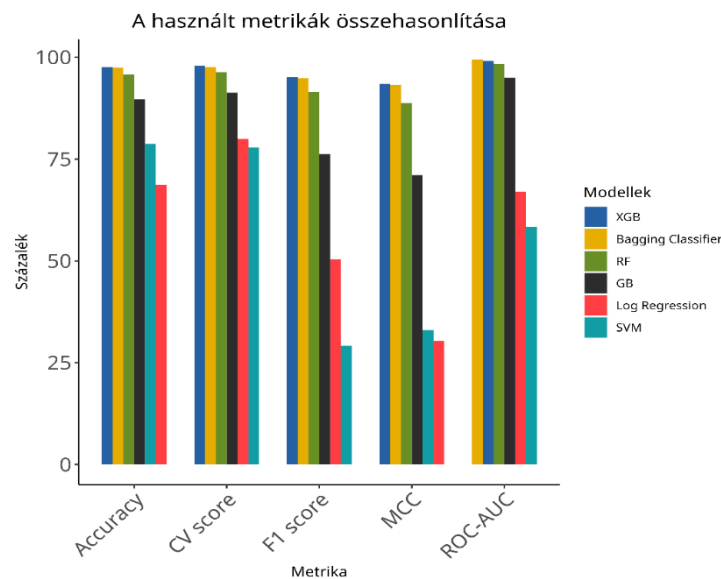


3. ábra: ANOVA teszthez használt adatok eloszlása

5.3. A modellek kiértékelése

A modellek értékeléséhez a 4.8.-as fejezetben leírt metrikákat használtam fel. A pontos értékeket a függelékben található 5. táblázat tartalmazza. A 4. ábrán található diagram alapján elmondható, hogy a döntési fa alapú modellek lényegesen jobb teljesítményt nyújtanak a vizsgált értékelőfüggvények tekintetében. A Bagging Classifier és az XGBoost modellek nagyon hasonló értékeket produkáltak. A Random Forest és a Gradient Boosting modellek teljesítménye is elfogadható, ha csak a jelzőszámok által mutatott képet vesszük figyelembe. A logisztikus modell és az SVM modelleknek ugyan

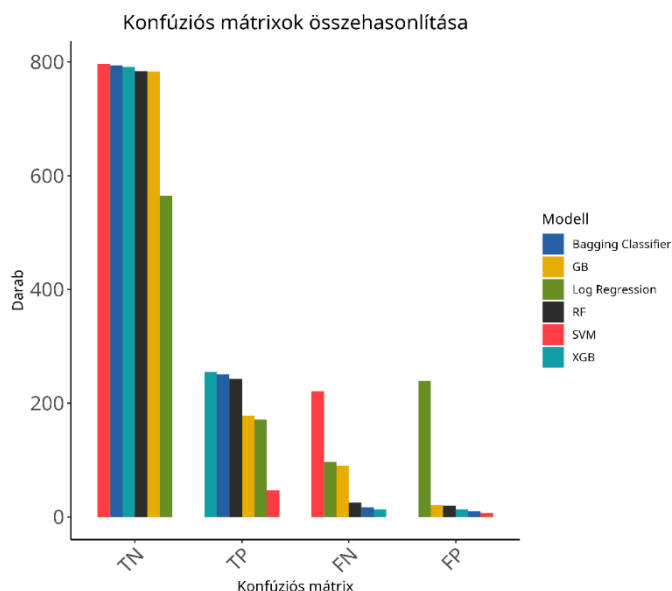
hiába magas a pontossági értékük, de a többi mérőszámban szignifikánsan alulmaradnak a többi modellhez képest.



4. ábra: Az egyes modellek metrikáit összehasonlító grafikon.

Fontos mérőszámok továbbá a konfúziós mátrixban található hamis és valós predikciók arányai. Ezek a függelék 6. táblázatában megtalálhatóak. Ezekből az adatokból készült az 5. ábrán látható diagram. Egy modell akkor teljesít jól, ha a különböző metrikái minél magasabbak és a hamis pozitív minták száma minél kevesebb. Egy jó modell képes minimalizálni mind a hamis pozitív (FP), mind a hamis negatív (FN) eredmények számát. Ez különösen fontos az olyan kontextusokban, ahol a hamis negatív eredmények következményei súlyosak lehetnek, például egészségügyi alkalmazásokban. Ha egy modell nem képes hatékonyan azonosítani a hamis negatív eredményeket, az azt jelenti, hogy a modell elmulasztja azonosítani a pozitív eseteket, ami potenciálisan súlyos következményekkel járhat. Ezért az ilyen esetekben egy hatékony gépi tanulási modellnek mindkét típusú hibát minimalizálnia kell, hogy maximalizálja az előrejelző teljesítményét és minimalizálja a negatív következményeket. Az F1-score és az MCC értékek a legfontosabb mérőszámok. Ezen értékek segítségével dönthető el, hogy egy modell jól teljesít vagy sem. Különösen kiegyensúlyozatlan adathalmazok esetén, ahol az MCC többet árul el a modell teljesítményéről, mint az F1-score. A mérőszámok és a hamis pozitívok száma alapján is a logisztikus regresszió és az SVM modellek nem teljesítettek elfogadhatóan. A fa alapú modellekhez képest nagyjából 10-szer több hamis negatív értéket prediktáltak. A függelék 6. táblázatban megtalálható egy valószínűségi érték, amely megmutatja, hogy a hamis negatívok valójában átlagosan mekkora valószínűséggel

pozitívok. A logisztikus regresszió esetében ez az érték ~44%, míg az SVM esetén ~28%. A többi modell 24% és 26% között található. Ebben a tekintetben is az XGBoost és a Bagging Classifier teljesítettek a legjobban. A modellek sorrendjét a 3. táblázat tartalmazza.



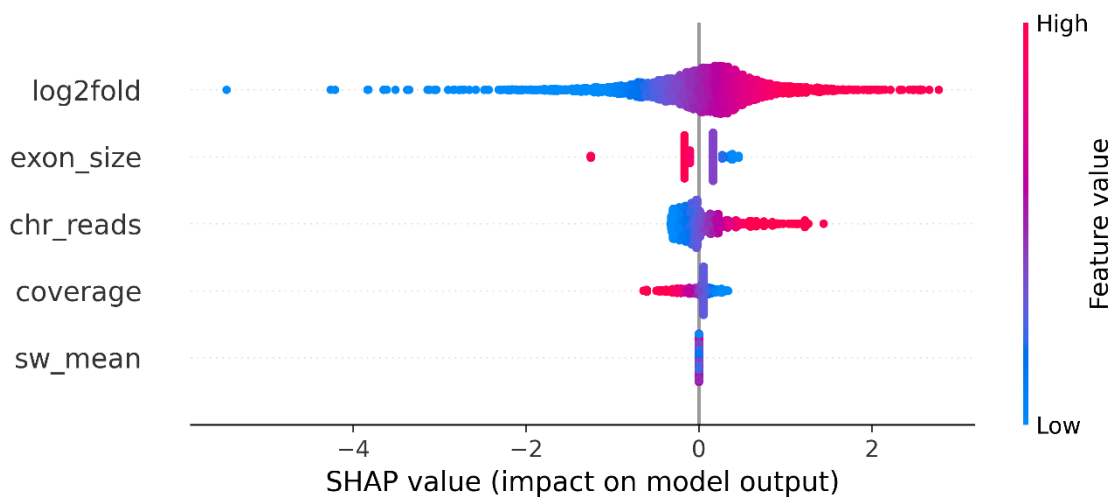
5. Ábra: A konfúziós mátrixok összegző grafikonja.

3. táblázat: A modellek sorrendjét tartalmazó táblázat

Sorszám	Modell
1.	XGBoost
2.	Bagging Classifier
3.	Random Forest
4.	Gradient Boosting
5.	Logisztikus regresszió
6.	Support Vector Machine

5.4. A logisztikus regressziós modell változóinak sorrendje

A logisztikus regresszió modell az egyes változókat különbözőképpen használta fel a predikcióhoz. Ahogy a 6. ábrán is látszik a besorolásra a legnagyobb befolyással a $\log_2\text{fold}$ változó volt. Megfigyelhető, hogy az értékek növekedésével egyértelműen pozitívnak jelöli az exont és minél alacsonyabb volt, annál inkább negatív osztályba sorolt. Hasonló tendencia látható a kromoszómára eső leolvasások (chr_reads) változó esetén is. A többi változó hozzájárulása eltörpül az előző kettő mellett. Az exon méret esetén egyértelműen meghatározható, hogy minél inkább nagyobb, annál kisebb az esély, hogy CNV-s legyen az adott gén. Az exonok mérete (exon_size) az előbbiekkel ellentétesen befolyásolta a döntéshozatalt, minél nagyobb volt egy exon, annál nagyobb valószínűséggel került egy minta a negatív osztályba. A lefedettség (coverage) is az exonmérethez hasonlóan változtatja a döntés kimenetelét. A csúszóablak átlag (sw_mean) pedig nem járul hozzá a döntéshez.

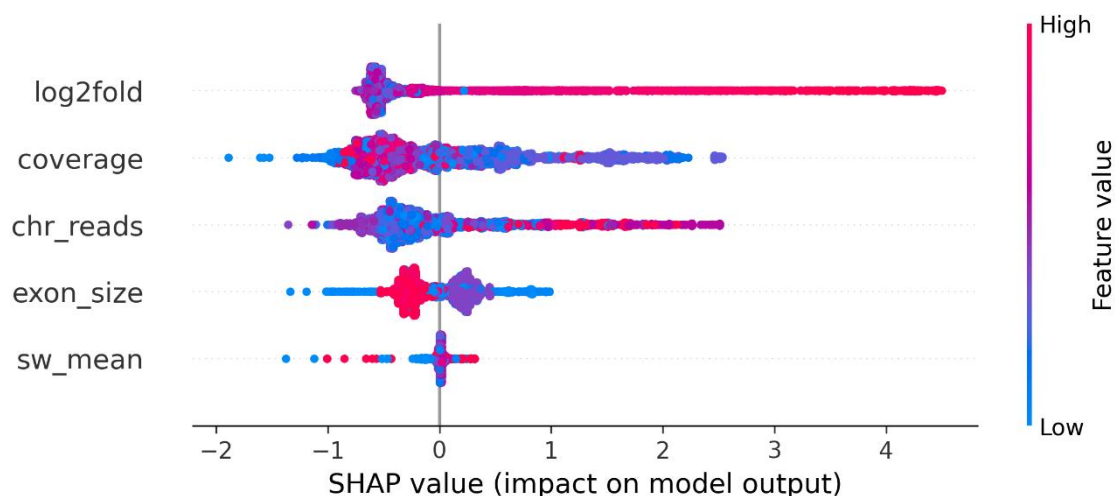


6. ábra: A logisztikus regressziós modell változóinak fontossága csökkenő sorrendben.

5.5. A Gradient Boosting modell klasszifikációját befolyásoló változók

A Gradient Boosting modell esetén már sokkal jobban megfigyelhető (7. ábra) az egyes értékek és változók hatása a modellre. A logisztikus regresszió modellhez hasonlóan a $\log_2\text{fold change}$ volt a hajtóerő. Minél nagyobb volt az érték, annál biztosabban került a pozitív osztályba a minta. A lefedettség esetén már kevésbé egyértelmű az elválasztás. Úgy tűnik, hogy minél nagyobb a lefedettség, annál biztosabban kerül a negatív csoportba egy minta. A kromoszómára eső leolvasás esetén is hasonló a tendencia, mint a $\log_2\text{fold change}$ esetében. A csúszóablak átlagnak itt már volt egy kis hatása, viszont itt csak

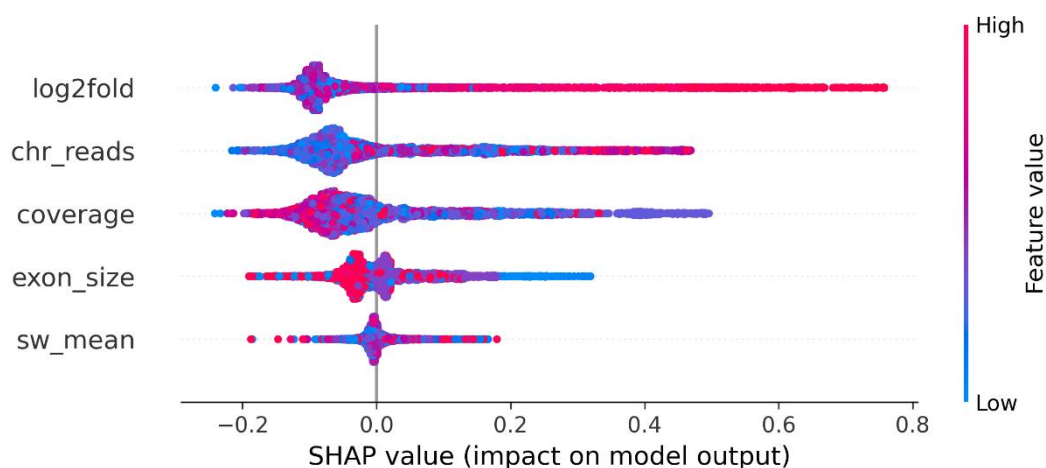
néhány esetben befolyásolta a döntést. Az érték nagyságával nőtt az esély, hogy a pozitív csoportba kerül a minta.



7. ábra: A Gradient Boosting modell változóinak fontossági sorrendje.

5.6. A Random Forest modell döntési tényezői

A Random Forest modell (8. ábra) is hasonló képet mutat a Gradient Boosting-hoz. Viszont az utóbbival ellentétben a kromoszómára eső leolvasások fontosabbak, mint a lefedettség. A log₂fold change az előző két modellhez hasonlóan dominálja a többi döntési tényezőt. Az eddigiekhez képest tisztábbak a határok az értékek nagyságát tekintve.

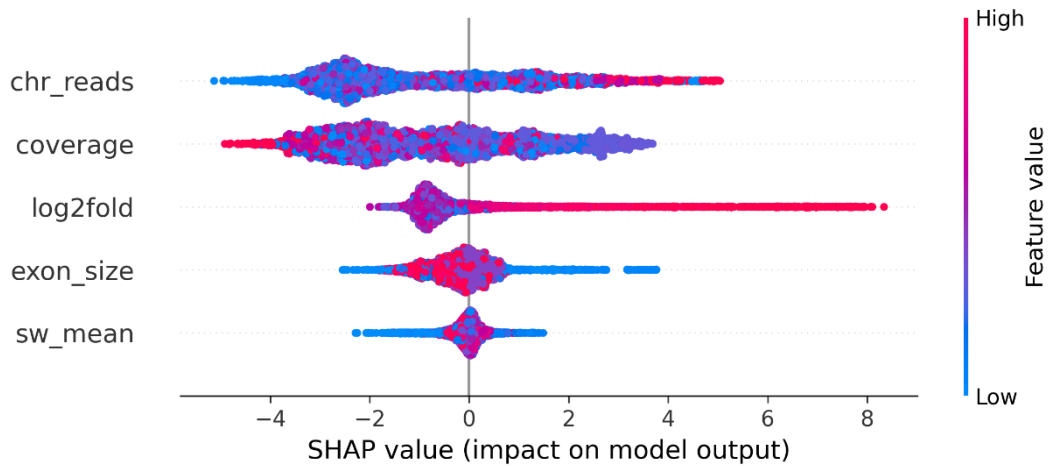


8. ábra: A Random Forest modell-t befolyásoló tényezők.

5.7. Az XGBoost modell döntéseit meghatározó változók

Az XGBoost változóinak fontossági sorrendje (9. ábra) eltér az eddigiektől. Ennek a modellnek a kromoszómákra eső leolvasások száma határozta meg legnagyobb részben a döntés kimenetelét. A második helyen a lefedettség áll, majd csak ez után következik a log₂fold change értékek. Az előbbi modellekhez hasonlóan befolyásolták ezek a

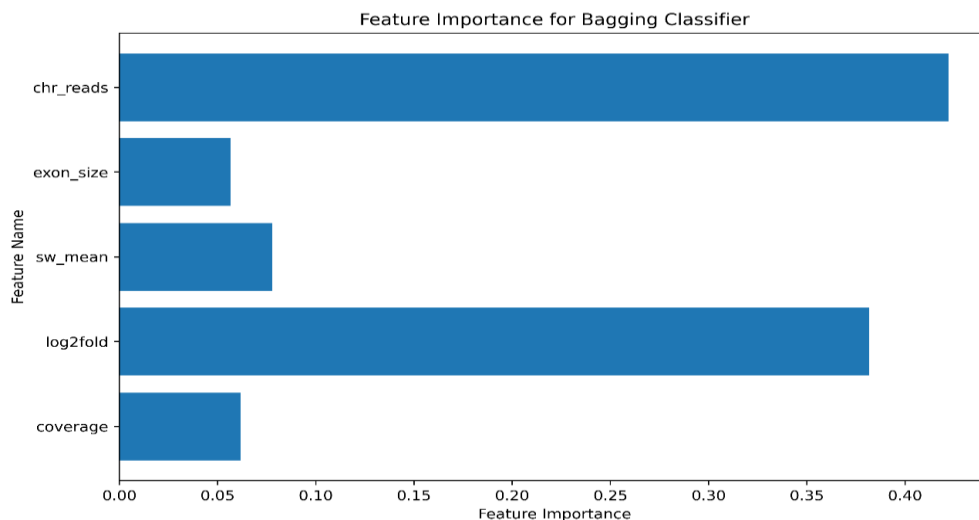
besorolást. Az exon méret és a csúszóablak átlag képe nem hasonlít az eddigiekre. A magas értékkel bíró minták mind a 0 érték köré tömörülnek. Az utóbbi változó csak akkor játszott szerepet, ha alacsony érték volt a mintában. Valószínűleg, ezek csak azokban az esetekben játszottak döntő szerepet, ha az adott exont az első három tényező alapján nem tudta a modell sehová sem besorolni.



9. ábra: Az XGBoost modell döntéseit meghatározó tényezők

5.8. A Bagging Classifier változóinak fontossági sorrendje

A Bagging Classifier algoritmust a SHAP csomag nem támogatja, ezért a Scikit-learn modul beépített algoritmusát használtam a fontossági sorrend meghatározására és ezeket az értékeket ábrázoltam a 13. ábrán. A legfontosabb tényező itt is a kromoszómára eső leolvasások száma volt, majd a log₂fold change értékek következnek. Az eddigi modellekkel ellentétben a Bagging Classifier a csúszóablak átlag értéke került a harmadik

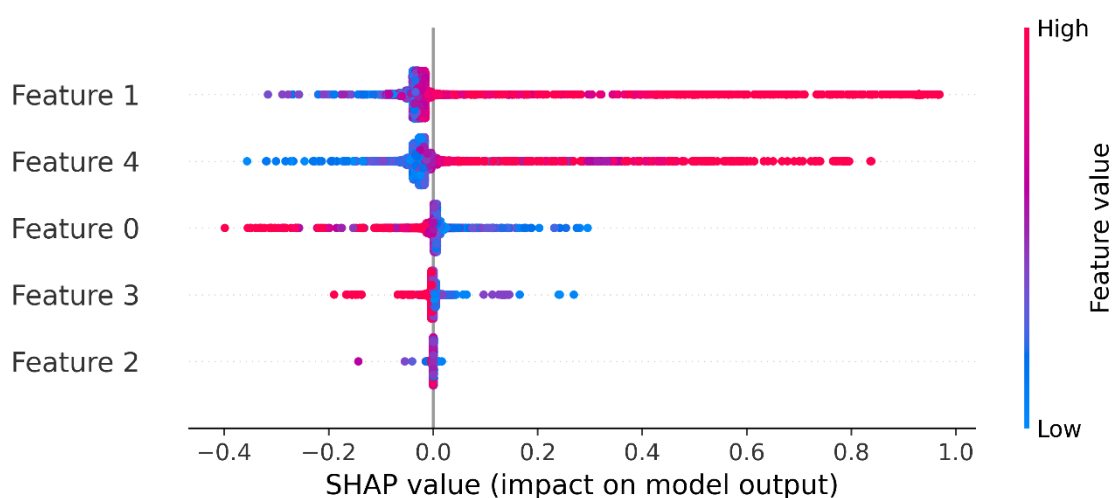


10. ábra: A Bagging Classifier változóinak fontosság értékei

helyre, bár igen alacsony százalékkal. Arról, hogy az exonokhoz tartozó egyes értékek nagysága hogyan befolyásolja a döntéshozatalt, sajnos nincs információ.

5.9. Az SVM modell döntéseit meghatározó tényezők sorrendje

A Support Vector Machine döntéseit (11. ábra) is javában a $\log_2$ fold change (Feature 1) érték határozta meg. Ezt követte a kromoszómára eső leolvasások (Feature 4) száma, a lefedettség (Feature 0), majd végül kis hozzájárulással az exonméret (Feature 3) és a csúszóablak átlag (Feature 2). Az első két változó esetében minél magasabbak voltak az értékek, annál biztosabban pozitív osztályba került a minta, míg a többi tényező fordítva befolyásolta a döntést.



11. ábra: Az SVM modell változóinak befolyása a döntéshozásra

5.10. Predikciók kiértékelése

A modellek számára ismeretlen mintákban ellenőriztem, hogy mely gének tartalmaznak validált kópiaszám változást. Ezzel a validált adatokkal vettem össze az egyes modellek predikcióit. A validációs adathalmazban mindösszesen 10 mintában találtunk kópiaszám változást, ami 14 érintett gént tartalmaz (4. táblázat). A 4. táblázat alapján rögtön szembetűnő, hogy a logisztikus regresszió (LR) és az SVM modellek minden valójában is pozitív exont felismertek. Azonban ezek a modellek minden egyes ismeretlen osztályú adatot pozitív osztályba soroltak. Emiatt az aránylag elfogadható metrikák alapján hiába tűnnek jónak, azonban a modellek mégsem működtek megfelelően. Ez lehet tanítási probléma következménye, vagy csak egyszerűen az, hogy nem megfelelőek ezek a modell erre a célra. Azt gondolom, hogy kicsi az esély arra, hogy tanítási hiba történhet, amikor a tesztadatokon a predikciók jól működtek, vagyis nem minden esetben csak pozitív osztályt kaptam vissza. Ez az állítás mindkét modell esetében igaz. A többi modell

egyenként 43%-os felfedezési aránnyal jelölte meg a kópiaszám változást tartalmazó exonokat. Az is megfigyelhető, hogy kivétel nélkül ugyanazokat a géneket osztályozták pozitívnak. Az egyedüli különbség az észlelt exonok számában található meg, azonban ez nem probléma, hiszen, ha egy exont is pozitívnak jelöl a modell, azt emberi ellenőrzés fogja követni. Ezért is fontos, hogy minél kevesebb hamis negatív értéket prediktáljanak a modellek, hiszen az egyik cél az, hogy minél kevesebb emberi erőforrással lehessen detektálni a kópiaszám változásokat. Az is megtörténhet, hogy valójában, amiket emberileg pozitívnak jelöltünk és a modellek pedig negatív csoportba sorolták ténylegesen negatívak. Ez további vizsgálatot igényel.

4. táblázat: A modellek által prediktált CNV-k összehasonlítása a már validált kópiaszám változást tartalmazó mintákkal

Minta	Validált CNV-s gén és típusa (exonok száma)	LR	RF	BC	SVM	GB	XGB
GMOS-659	CDKN2A deléció (4 exon)						
GMOS-659	EGFR amplifikáció (4 exon)						
GMOS-673	RAD54L exon 11-19 deléció (9 exon)						
GMOS-684	CDKN2A deléció (4 exon)						
GMOS-689	FGFR1 amplifikáció (17 exon)						
GMOS-690	FGFR1 amplifikáció (17 exon)						
GMOS-697	PDGFRA amplifikáció (22 exon)						
GMOS-698	MSH2 amplifikáció (16 exon)						
GMOS-698	MSH6 amplifikáció (10 exon)						
GMOS-711	PIK3CA deléció (5 exon)						
GMOS-733	MSH2 deléció (16 exon)						
GMOS-733	MSH6 deléció (10 exon)						

Jelmagyarázat:

	Egy exon-t sem jelölt pozitívnak.
	A CNV-s exonok kevesebb mint felét jelölte pozitívnak.
	A CNV-s exonok felét vagy több, mint felét jelölte pozitívnak.
	Minden CNV-s exont pozitívnak jelölt.

Összefoglalás

Számos olyan modellt dolgoztam ki, amelyek képesek az DNS-ben bekövetkező szerkezeti változások kimutatására és azonosítására. A korlátozott adatmennyiség ellenére a jelenlegi adatkészleten való tanítás során az XGBoost és a Bagging osztályozók ismeretlen minták előrejelzésére használhatók, így kiváló alapot biztosítanak a jövőbeni jobb algoritmusok fejlesztéséhez. Mindezt úgy, hogy csak limitált mennyiségű adat állt rendelkezésemre. Az eddig vizsgált 6 modell közül az XGBoost-tal és a Bagging Classifier-rel lehet a jelenlegi adatsoron tanítva ismeretlen mintákat a legjobban prediktálni. Ezek kiváló alapot adnak egy olyan új detektáló algoritmushoz, amely a jövőben szinte hiba nélkül fog tudni azonosítani kópiaszám változásokat és könnyíti meg a mindennapi elemzéseket és segíti a validációt.

A tanulmány korlátja a valós pozitív minták kis száma, ami akadályozza a különböző modellek teljesítményét. Idővel a szekvenálási költségek csökkenésével ez a probléma is megoldható.

A mesterséges intelligencia rohamos fejlődése ráadásul olyan jobb modelleket fog eredményezni, amelyeket alkalmazni lehet az ilyen problémák megoldására. A neurális hálózatok elterjedése az orvosi biológiai adatokban is gyorsabb és pontosabb előrejelzéseket tesz lehetővé. Fontos kiemelni, hogy a jelenlegi pozitív mintaméret mellett nem lehetséges neurális hálózati modell betanítása.

Összefoglalva a dolgozatomban egy olyan új gépi tanulási modell fejlesztését ismertetem, amely képes pontosan megjósolni a kópiák számában bekövetkező változásokat, ezáltal csökkentve a kutatók által ezen adatok manuális elemzésére fordított időt és energiát.

Irodalomjegyzék

- Beroukhir, R., Mermel, C. H., Porter, D., Wei, G., Raychaudhuri, S., Donovan, J., Barretina, J., Boehm, J. S., Dobson, J., Urashima, M., Mc Henry, K. T., Pinchback, R. M., Ligon, A. H., Cho, Y.-J., Haery, L., Greulich, H., Reich, M., Winckler, W., Lawrence, M. S., ... Meyerson, M. (2010). The landscape of somatic copy-number alteration across human cancers. *Nature*, 463(7283), 899–905. <https://doi.org/10.1038/nature08822>
- Boeva, V., Popova, T., Bleakley, K., Chiche, P., Cappel, J., Schleiermacher, G., Janoueix-Lerosey, I., Delattre, O., & Barillot, E. (2012). Control-FREEC: a tool for assessing copy number and allelic content using next-generation sequencing data. *Bioinformatics*, 28(3), 423–425. <https://doi.org/10.1093/bioinformatics/btr670>
- Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2), 123–140. <https://doi.org/10.1007/BF00058655>
- Budczies, J., Bockmayr, M., Denkert, C., Klauschen, F., Gröschel, S., Darb-Esfahani, S., Pfarr, N., Leichsenring, J., Onozato, M. L., Lennerz, J. K., Dietel, M., Fröhling, S., Schirmacher, P., Iafrate, A. J., Weichert, W., & Stenzinger, A. (2016). Pan-cancer analysis of copy number changes in programmed death-ligand 1 (PD-L1, CD274) - associations with gene expression, mutational load, and survival. *Genes, Chromosomes and Cancer*, 55(8), 626–639. <https://doi.org/10.1002/gcc.22365>
- Burkart, N., & Huber, M. F. (2021). A Survey on the Explainability of Supervised Machine Learning. *Journal of Artificial Intelligence Research*, 70, 245–317. <https://doi.org/10.1613/jair.1.12228>
- Celesti, F., Celesti, A., Wan, J., & Villari, M. (2018). Why Deep Learning Is Changing the Way to Approach NGS Data Processing: A Review. *IEEE Reviews in Biomedical Engineering*, 11, 68–76. <https://doi.org/10.1109/RBME.2018.2825987>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321–357. <https://doi.org/10.1613/jair.953>
- Chen, L., Zhou, W., Zhang, L., & Zhang, F. (2014). Genome Architecture and Its Roles in Human Copy Number Variation. *Genomics & Informatics*, 12(4), 136. <https://doi.org/10.5808/GI.2014.12.4.136>
- Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. <https://doi.org/10.1145/2939672.2939785>
- Chen, Y., Zhao, L., Wang, Y., Cao, M., Gelowani, V., Xu, M., Agrawal, S. A., Li, Y., Daiger, S. P., Gibbs, R., Wang, F., & Chen, R. (2017). SeqCNV: a novel method for identification of copy number variations in targeted next-generation sequencing data. *BMC Bioinformatics*, 18(1), 147. <https://doi.org/10.1186/s12859-017-1566-3>
- Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21(1), 6. <https://doi.org/10.1186/s12864-019-6413-7>
- Chicco, D., Tötsch, N., & Jurman, G. (2021). The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData Mining*, 14(1), 13. <https://doi.org/10.1186/s13040-021-00244-z>
- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297.
- Couldrey, C., Keehan, M., Johnson, T., Tiplady, K., Winkelman, A., Littlejohn, M. D., Scott, A., Kemper, K. E., Hayes, B., Davis, S. R., & Spelman, R. J. (2017). Detection and assessment of copy number variation using PacBio long-read and Illumina sequencing in New Zealand dairy cattle. *Journal of Dairy Science*, 100(7), 5472–5478. <https://doi.org/10.3168/jds.2016-12199>
- Cox, D. R. (1958). The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, 20(2), 215–242. <http://www.jstor.org/stable/2983890>
- Croce, C. M. (2008). Oncogenes and Cancer. *New England Journal of Medicine*, 358(5), 502–511. <https://doi.org/10.1056/NEJMr072367>
- Duan, J., Zhang, J.-G., Deng, H.-W., & Wang, Y.-P. (2013). Comparative Studies of Copy Number Variation Detection Methods for Next-Generation Sequencing Technologies. *PLoS ONE*, 8(3), e59128. <https://doi.org/10.1371/journal.pone.0059128>
- Fanciulli, M., Petretto, E., & Aitman, T. (2010). Gene copy number variation and common human disease. *Clinical Genetics*, 77(3), 201–213. <https://doi.org/10.1111/j.1399-0004.2009.01342.x>
- Feuk, L., Carson, A. R., & Scherer, S. W. (2006). Structural variation in the human genome. *Nature Reviews Genetics*, 7(2), 85–97. <https://doi.org/10.1038/nrg1767>
- Foundation, T. D. (n.d.). *LibreOffice Calc*. <https://www.libreoffice.org/discover/calc/>
- Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, 1189–1232.

- Gabrielaite, M., Torp, M. H., Rasmussen, M. S., Andreu-Sánchez, S., Vieira, F. G., Pedersen, C. B., Kinalis, S., Madsen, M. B., Kodama, M., Demircan, G. S., Simonyan, A., Yde, C. W., Olsen, L. R., Marvig, R. L., Østrup, O., Rossing, M., Nielsen, F. C., Winther, O., & Bagger, F. O. (2021). A Comparison of Tools for Copy-Number Variation Detection in Germline Whole Exome and Whole Genome Sequencing Data. *Cancers*, 13(24), 6283. <https://doi.org/10.3390/cancers13246283>
- García, S., Fernández, A., Luengo, J., & Herrera, F. (2009). A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability. *Soft Computing*, 13(10), 959–977. <https://doi.org/10.1007/s00500-008-0392-y>
- Gross, A. M., Ajay, S. S., Rajan, V., Brown, C., Bluske, K., Burns, N. J., Chawla, A., Coffey, A. J., Malhotra, A., Scocchia, A., Thorpe, E., Dzidic, N., Hovanes, K., Sahoo, T., Dolzhenko, E., Lajoie, B., Khouzam, A., Chowdhury, S., Belmont, J., ... Taft, R. J. (2019). Copy-number variants in clinical genome sequencing: deployment and interpretation for rare and undiagnosed disease. *Genetics in Medicine*, 21(5), 1121–1130. <https://doi.org/10.1038/s41436-018-0295-y>
- Gutiérrez, L., Parada-Fennen, L., & Solano, A. R. (2023). In Silico Copy Number Variation (CNVs) Bioinformatics Estimation: Dream or Nightmare? *EJIFCC*, 34(1), 72–75.
- Hajian-Tilaki, K. (2013). Receiver Operating Characteristic (ROC) Curve Analysis for Medical Diagnostic Test Evaluation. *Caspian Journal of Internal Medicine*, 4(2), 627–635.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (Second edition). Springer.
- Henrichsen, C. N., Chaignat, E., & Reymond, A. (2009). Copy number variants, diseases and gene expression. *Human Molecular Genetics*, 18(R1), R1–R8. <https://doi.org/10.1093/hmg/ddp011>
- Ho, S. S., Urban, A. E., & Mills, R. E. (2020). Structural variation in the sequencing era. *Nature Reviews Genetics*, 21(3), 171–189. <https://doi.org/10.1038/s41576-019-0180-9>
- Hu, L., Liang, F., Cheng, D., Zhang, Z., Yu, G., Zha, J., Wang, Y., Xia, Q., Yuan, D., Tan, Y., Wang, D., Liang, Y., & Lin, G. (2020). Location of Balanced Chromosome-Translocation Breakpoints by Long-Read Sequencing on the Oxford Nanopore Platform. *Frontiers in Genetics*, 10. <https://doi.org/10.3389/fgene.2019.01313>
- Huddleston, J., Chaisson, M. J. P., Steinberg, K. M., Warren, W., Hoekzema, K., Gordon, D., Graves-Lindsay, T. A., Munson, K. M., Kronenberg, Z. N., Vives, L., Peluso, P., Boitano, M., Chin, C.-S., Korlach, J., Wilson, R. K., & Eichler, E. E. (2017). Discovery and genotyping of structural variation from long-read haploid genome sequence data. *Genome Research*, 27(5), 677–685. <https://doi.org/10.1101/gr.214007.116>
- Iafrate, A. J., Feuk, L., Rivera, M. N., Listewnik, M. L., Donahoe, P. K., Qi, Y., Scherer, S. W., & Lee, C. (2004). Detection of large-scale variation in the human genome. *Nature Genetics*, 36(9), 949–951. <https://doi.org/10.1038/ng1416>
- Ionita-Laza, I., Rogers, A. J., Lange, C., Raby, B. A., & Lee, C. (2009). Genetic association analysis of copy-number variation (CNV) in human disease pathogenesis. *Genomics*, 93(1), 22–26. <https://doi.org/10.1016/j.ygeno.2008.08.012>
- Jiang, Y., Wang, R., Urrutia, E., Anastopoulos, I. N., Nathanson, K. L., & Zhang, N. R. (2018). CODEX2: full-spectrum copy number variation detection by high-throughput DNA sequencing. *Genome Biology*, 19(1), 202. <https://doi.org/10.1186/s13059-018-1578-y>
- Joblib Development Team. (2020). *Joblib: running Python functions as pipeline jobs*. <https://joblib.readthedocs.io/>
- Jovel, J., & Greiner, R. (2021). An Introduction to Machine Learning Approaches for Biomedical Research. *Frontiers in Medicine*, 8. <https://doi.org/10.3389/fmed.2021.771607>
- Kaufmann, J., & Schering, A. (2014). Analysis of Variance ANOVA. In *Wiley StatsRef: Statistics Reference Online*. Wiley. <https://doi.org/10.1002/9781118445112.stat06938>
- Kinjo, S., Monma, N., Misu, S., Kitamura, N., Imoto, J., Yoshitake, K., Gojobori, T., & Ikeo, K. (2018). Maser: one-stop platform for NGS big data from analysis to visualization. *Database*, 2018. <https://doi.org/10.1093/database/bay027>
- Klambauer, G., Schwarzbauer, K., Mayr, A., Clevert, D.-A., Mitterecker, A., Bodenhofer, U., & Hochreiter, S. (2012). cn.MOPS: mixture of Poissons for discovering copy number variations in next-generation sequencing data with a low false discovery rate. *Nucleic Acids Research*, 40(9), e69–e69. <https://doi.org/10.1093/nar/gks003>
- Kulkarni, A., Chong, D., & Batarseh, F. A. (2020). Foundations of data imbalance and solutions for a data democracy. In *Data Democracy* (pp. 83–106). Elsevier. <https://doi.org/10.1016/B978-0-12-818366-3.00005-8>
- Lauer, S., & Gresham, D. (2019). An evolving view of copy number variants. *Current Genetics*, 65(6), 1287–1295. <https://doi.org/10.1007/s00294-019-00980-0>
- LaValle, S. M., Branicky, M. S., & Lindemann, S. R. (2004). On the relationship between classical grid search and probabilistic roadmaps. *The International Journal of Robotics Research*, 23(7–8), 673–692.

- Lee, E. Y. H. P., & Muller, W. J. (2010a). Oncogenes and Tumor Suppressor Genes. *Cold Spring Harbor Perspectives in Biology*, 2(10), a003236–a003236. <https://doi.org/10.1101/cshperspect.a003236>
- Lee, E. Y. H. P., & Muller, W. J. (2010b). Oncogenes and Tumor Suppressor Genes. *Cold Spring Harbor Perspectives in Biology*, 2(10), a003236–a003236. <https://doi.org/10.1101/cshperspect.a003236>
- Lepkes, L., Kayali, M., Blümcke, B., Weber, J., Suszynska, M., Schmidt, S., Borde, J., Klonowska, K., Wappenschmidt, B., Hauke, J., Kozłowski, P., Schmutzler, R. K., Hahnen, E., & Ernst, C. (2021). Performance of In Silico Prediction Tools for the Detection of Germline Copy Number Variations in Cancer Predisposition Genes in 4208 Female Index Patients with Familial Breast and Ovarian Cancer. *Cancers*, 13(1). <https://doi.org/10.3390/cancers13010118>
- Li, P., Luo, H., Ji, B., & Nielsen, J. (2022). Machine learning for data integration in human gut microbiome. *Microbial Cell Factories*, 21(1), 241. <https://doi.org/10.1186/s12934-022-01973-4>
- Li, W., & Olivier, M. (2013). Current analysis platforms and methods for detecting copy number variation. *Physiological Genomics*, 45(1), 1–16. <https://doi.org/10.1152/physiolgenomics.00082.2012>
- Liaw, A., & Wiener, M. (2002). Classification and Regression by randomForest. *R News*, 2(3), 18–22. <https://CRAN.R-project.org/doc/Rnews/>
- Liu, R., Dai, W., Wu, T., Wang, M., Wan, S., & Liu, J. (2022). AIMIC: Deep Learning for Microscopic Image Classification. *Computer Methods and Programs in Biomedicine*, 226, 107162. <https://doi.org/10.1016/j.cmpb.2022.107162>
- Loh, H. W., Ooi, C. P., Vicnesh, J., Oh, S. L., Faust, O., Gertych, A., & Acharya, U. R. (2020). Automated Detection of Sleep Stages Using Deep Learning Techniques: A Systematic Review of the Last Decade (2010–2020). *Applied Sciences*, 10(24), 8963. <https://doi.org/10.3390/app10248963>
- Love, M. I., Huber, W., & Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15(12), 550. <https://doi.org/10.1186/s13059-014-0550-8>
- Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1), 2522–5839.
- Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. In I. Guyon, U. V Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 30* (pp. 4765–4774). Curran Associates, Inc. <http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf>
- Manolio, T. A., Collins, F. S., Cox, N. J., Goldstein, D. B., Hindorff, L. A., Hunter, D. J., McCarthy, M. I., Ramos, E. M., Cardon, L. R., Chakravarti, A., Cho, J. H., Guttacher, A. E., Kong, A., Kruglyak, L., Mardis, E., Rotimi, C. N., Slatkin, M., Valle, D., Whittemore, A. S., ... Visscher, P. M. (2009). Finding the missing heritability of complex diseases. *Nature*, 461(7265), 747–753. <https://doi.org/10.1038/nature08494>
- Marcot, B. G., & Hanea, A. M. (2021). What is an optimal value of k in k-fold cross-validation in discrete Bayesian network analysis? *Computational Statistics*, 36(3), 2009–2031. <https://doi.org/10.1007/s00180-020-00999-9>
- Marx, V. (2023). Method of the year: long-read sequencing. *Nature Methods*, 20(1), 6–11. <https://doi.org/10.1038/s41592-022-01730-w>
- Mckinney, W. (2011). *pandas: a Foundational Python Library for Data Analysis and Statistics*. <http://pandas.sf.net>
- Microsoft Corporation. (n.d.). *Microsoft Excel*. <https://office.microsoft.com/excel>
- Moreno-Cabrera, J. M., Del Valle, J., Castellanos, E., Feliubadaló, L., Pineda, M., Brunet, J., Serra, E., Capellà, G., Lázaro, C., & Gel, B. (2020). Evaluation of CNV detection tools for NGS panel data in genetic diagnostics. *European Journal of Human Genetics : EJHG*, 28(12), 1645–1655. <https://doi.org/10.1038/s41431-020-0675-z>
- Nagahashi, M., Shimada, Y., Ichikawa, H., Kameyama, H., Takabe, K., Okuda, S., & Wakai, T. (2019). Next generation sequencing-based gene panel tests for the management of solid tumors. *Cancer Science*, 110(1), 6–15. <https://doi.org/10.1111/cas.13837>
- Nasteski, V. (2017). An overview of the supervised machine learning methods. *HORIZONS.B*, 4, 51–62. <https://doi.org/10.20544/HORIZONS.B.04.1.17.P05>
- Özgür, S., & Orman, M. (2023). Application of deep learning technique in next generation sequence experiments. *Journal of Big Data*, 10(1), 160. <https://doi.org/10.1186/s40537-023-00838-w>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85), 2825–2830. <http://jmlr.org/papers/v12/pedregosa11a.html>

- Peng, H., Lu, L., Zhou, Z., Liu, J., Zhang, D., Nan, K., Zhao, X., Li, F., Tian, L., Dong, H., & Yao, Y. (2019). CNV Detection from Circulating Tumor DNA in Late Stage Non-Small Cell Lung Cancer Patients. *Genes*, 10(11), 926. <https://doi.org/10.3390/genes10110926>
- Pös, O., Radvanszky, J., Styk, J., Pös, Z., Buglyó, G., Kajsik, M., Budis, J., Nagy, B., & Szemes, T. (2021). Copy Number Variation: Methods and Clinical Applications. *Applied Sciences*, 11(2), 819. <https://doi.org/10.3390/app11020819>
- Qorri, E., Takács, B., Gráf, A., Enyedi, M. Z., Pintér, L., Kiss, E., & Haracska, L. (2022). A Comprehensive Evaluation of the Performance of Prediction Algorithms on Clinically Relevant Missense Variants. *International Journal of Molecular Sciences*, 23(14), 7946. <https://doi.org/10.3390/ijms23147946>
- Rajagopalan, R., Murrell, J. R., Luo, M., & Conlin, L. K. (2020). A highly sensitive and specific workflow for detecting rare copy-number variants from exome sequencing data. *Genome Medicine*, 12(1), 14. <https://doi.org/10.1186/s13073-020-0712-0>
- Roca, I., González-Castro, L., Fernández, H., Couce, M. L., & Fernández-Marmiesse, A. (2019). Free-access copy-number variant detection tools for targeted next-generation sequencing data. *Mutation Research/Reviews in Mutation Research*, 779, 114–125. <https://doi.org/10.1016/j.mrrev.2019.02.005>
- Samulin Erdem, J., Arnoldussen, Y. J., Skaug, V., Haugen, A., & Zienolddiny, S. (2017). Copy number variation, increased gene expression, and molecular mechanisms of neurofascin in lung cancer. *Molecular Carcinogenesis*, 56(9), 2076–2085. <https://doi.org/10.1002/mc.22664>
- Schmidt, A., Röner, S., Mai, K., Klinkhammer, H., Kircher, M., & Ludwig, K. U. (2023). Predicting the pathogenicity of missense variants using features derived from AlphaFold2. *Bioinformatics*, 39(5). <https://doi.org/10.1093/bioinformatics/btad280>
- Schmidt, B., & Hildebrandt, A. (2017). Next-generation sequencing: big data meets high performance computing. *Drug Discovery Today*, 22(4), 712–717. <https://doi.org/10.1016/j.drudis.2017.01.014>
- Schuster, S. C. (2008). Next-generation sequencing transforms today's biology. *Nature Methods*, 5(1), 16–18. <https://doi.org/10.1038/nmeth1156>
- Sebat, J., Lakshmi, B., Troge, J., Alexander, J., Young, J., Lundin, P., Månér, S., Massa, H., Walker, M., Chi, M., Navin, N., Lucito, R., Healy, J., Hicks, J., Ye, K., Reiner, A., Gilliam, T. C., Trask, B., Patterson, N., ... Wigler, M. (2004). Large-Scale Copy Number Polymorphism in the Human Genome. *Science*, 305(5683), 525–528. <https://doi.org/10.1126/science.1098918>
- Shao, X., Lv, N., Liao, J., Long, J., Xue, R., Ai, N., Xu, D., & Fan, X. (2019). Copy number variation is highly correlated with differential gene expression: a pan-cancer study. *BMC Medical Genetics*, 20(1), 175. <https://doi.org/10.1186/s12881-019-0909-5>
- Shinde, P. P., & Shah, S. (2018). A Review of Machine Learning and Deep Learning Applications. *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 1–6. <https://doi.org/10.1109/ICCUBEA.2018.8697857>
- Shlien, A., & Malkin, D. (2009). Copy number variations and cancer. *Genome Medicine*, 1(6), 62. <https://doi.org/10.1186/gm62>
- Singh, A. K., Olsen, M. F., Lavik, L. A. S., Vold, T., Drabløs, F., & Sjørnsen, W. (2021). Detecting copy number variation in next generation sequencing data from diagnostic gene panels. *BMC Medical Genomics*, 14(1), 214. <https://doi.org/10.1186/s12920-021-01059-x>
- Slatko, B. E., Gardner, A. F., & Ausubel, F. M. (2018). Overview of Next-Generation Sequencing Technologies. *Current Protocols in Molecular Biology*, 122(1). <https://doi.org/10.1002/cpmb.59>
- Smolander, J., Khan, S., Singaravelu, K., Kauko, L., Lund, R. J., Laiho, A., & Elo, L. L. (2021). Evaluation of tools for identifying large copy number variations from ultra-low-coverage whole-genome sequencing data. *BMC Genomics*, 22(1), 357. <https://doi.org/10.1186/s12864-021-07686-z>
- Sokolova, M., Japkowicz, N., & Szpakowicz, S. (2006). *Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation* (pp. 1015–1021). https://doi.org/10.1007/11941439_114
- Spielmann, M., Lupiáñez, D. G., & Mundlos, S. (2018). Structural variation in the 3D genome. *Nature Reviews Genetics*, 19(7), 453–467. <https://doi.org/10.1038/s41576-018-0007-0>
- Stankiewicz, P., & Lupski, J. R. (2010). Structural Variation in the Human Genome and its Role in Disease. *Annual Review of Medicine*, 61(1), 437–455. <https://doi.org/10.1146/annurev-med-100708-204735>
- Strzelecki, M., & Badura, P. (2022). Machine Learning for Biomedical Application. *Applied Sciences*, 12(4), 2022. <https://doi.org/10.3390/app12042022>
- Sudmant, P. H., Rausch, T., Gardner, E. J., Handsaker, R. E., Abyzov, A., Huddleston, J., Zhang, Y., Ye, K., Jun, G., Hsi-Yang Fritz, M., Konkel, M. K., Malhotra, A., Stütz, A. M., Shi, X., Paolo Casale, F., Chen, J., Hormozdiari, F., Dayama, G., Chen, K., ... Korbel, J. O. (2015). An integrated map of structural variation in 2,504 human genomes. *Nature*, 526(7571), 75–81. <https://doi.org/10.1038/nature15394>

- The pandas development team. (2020). *Pandas* (latest).
- Usama, M., Qadir, J., Raza, A., Arif, H., Yau, K. A., Elkhathib, Y., Hussain, A., & Al-Fuqaha, A. (2019). Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges. *IEEE Access*, 7, 65579–65615. <https://doi.org/10.1109/ACCESS.2019.2916648>
- Van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace.
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., ... Vázquez-Baeza, Y. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3), 261–272. <https://doi.org/10.1038/s41592-019-0686-2>
- Weir, B. A., Woo, M. S., Getz, G., Perner, S., Ding, L., Beroukhi, R., Lin, W. M., Province, M. A., Kraja, A., Johnson, L. A., Shah, K., Sato, M., Thomas, R. K., Barletta, J. A., Borecki, I. B., Broderick, S., Chang, A. C., Chiang, D. Y., Chirieac, L. R., ... Meyerson, M. (2007). Characterizing the cancer genome in lung adenocarcinoma. *Nature*, 450(7171), 893–898. <https://doi.org/10.1038/nature06358>
- Weir, B., Zhao, X., & Meyerson, M. (2004). Somatic alterations in the human cancer genome. *Cancer Cell*, 6(5), 433–438. <https://doi.org/10.1016/j.ccr.2004.11.004>
- Weischenfeldt, J., Symmons, O., Spitz, F., & Korbel, J. O. (2013). Phenotypic impact of genomic structural variation: insights from and for human disease. *Nature Reviews Genetics*, 14(2), 125–138. <https://doi.org/10.1038/nrg3373>
- Wu, J., Grzeda, K. R., Stewart, C., Grubert, F., Urban, A. E., Snyder, M. P., & Marth, G. T. (2012). Copy Number Variation detection from 1000 Genomes project exon capture sequencing data. *BMC Bioinformatics*, 13(1), 305. <https://doi.org/10.1186/1471-2105-13-305>
- Yacouby, R., & Axman, D. (2020). Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models. *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, 79–91. <https://doi.org/10.18653/v1/2020.eval4nlp-1.9>
- Zarrei, M., MacDonald, J. R., Merico, D., & Scherer, S. W. (2015). A copy number variation map of the human genome. *Nature Reviews Genetics*, 16(3), 172–183. <https://doi.org/10.1038/nrg3871>
- Zhao, M., Wang, Q., Wang, Q., Jia, P., & Zhao, Z. (2013). Computational tools for copy number variation (CNV) detection using next-generation sequencing data: features and perspectives. *BMC Bioinformatics*, 14(S11), S1. <https://doi.org/10.1186/1471-2105-14-S11-S1>

Függelék

A felhasznált szkriptek és algoritmusok az alábbi OneDrive mappában elérhetőek:

<https://deltabio2000->

my.sharepoint.com/:f/g/personal/valentin_varga_deltabio_eu/Eg_klDndQqNNiFVeDELWHNcByt5yhYCKmKo0qxktUo9JuQ?e=4LeKhK

- **5. táblázat:** A modellek kiértékelését segítő metrikákat összesítő táblázat.

	Samples used for training:		POSITIVES: 1340		NEGATIVES: 4020	
	LR	GB	RF	XGB	BC	SVM
Accuracy	68,66%	89,65%	95,80%	97,57%	97,48%	78,73%
CV score	80,00%	91,29%	96,32%	97,85%	97,59%	77,89%
F1 score	50,44%	76,23%	91,53%	95,15%	94,90%	29,19%
MCC	30,36%	71,06%	88,74%	93,53%	93,24%	33,00%
ROC-AUC	67,04%	94,95%	98,38%	99,10%	99,46%	58,33%

- **6. táblázat:** A modellek konfúziós mátrixait összesítő táblázat

	LR	GB	RF	XGB	BC	SVM
TP	171	178	243	255	251	47
TN	565	783	784	791	794	797
FP	239	21	20	13	10	7
FN	97	90	25	13	17	221
Mean p of FN being +	45,84%	24,36%	24,98%	24,98%	26,27%	26,45%

- **7. táblázat:** Az egyes modellek változóinak fontossági értékei/koefficiensei

	LR	GB	RF	XGB	BC	SVM
Coverage	-0,15855765	23,38%	23,57%	17,92%	6,18%	-0,00025187
log2fold	0,67060474	45,51%	33,20%	41,09%	38,17%	0,04524254
sw_mean	0	0,86%	8,39%	4,69%	7,79%	1,87E-05
exon_size	-0,25087274	5,03%	7,65%	16,57%	5,65%	-0,00121269
chr_reads	0,265125693	25,22%	27,19%	19,73%	42,21%	0,01815299

- All-Cancer-54 -es panelben található gének:

AKT1, ALK, APC, APLNR, ATM, ATR, BARD1, BRAF, BRCA1, BRCA2, BRIP1, CDK12, CDK4, CDKN2A, CHEK1, CHEK2, EGFR, ERBB2, ESR1, FANCL, FGFR1, FGFR2, FGFR3, HDAC2, HRAS, IDH1, IDH2, KIT, KRAS, MAP2K1, MET, MLH1, MSH2, MSH6, NF1, NOTCH1, NRAS, NTRK1, NTRK2, NTRK3, PALB2, PDGFRA, PIK3CA, POLE, PTEN, RAD51B, RAD51C, RAD51D, RAD54L, RET, ROS1, SETD2, SMARCB1, STK11, TP53

- All-Cancer-55-ös panelben található gének:

ABL1, AKT1, ALK, APC, ATM, BRAF, BRCA1, BRCA2, CDH1, CDK4, CDK6, CDKN2A, CSF1R, CTNNB1, DDR2, EGFR, ERBB2, ERBB4, ESR1, FGFR1, FGFR2, FGFR3, GNA11, GNAQ, GNAS, HRAS, IDH1, IDH2, JAK2, KDR, KIT, KRAS, MAP2K1, MET, MLH1, MTOR, MYCN, MYC, NOTCH1, NRAS, NTRK1, PDGFRA, PIK3CA, PTCH1, PTEN, PTPN11, RB1, RET, ROS1, SMAD4, SMO, SRC, STK11, TP53

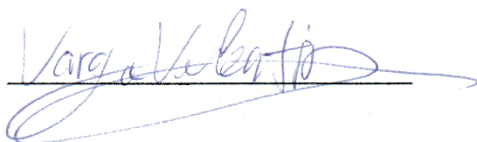
Tételes szerzői hozzájárulás

Alulírott **Varga Valentin** nyilatkozom, hogy a Biológia TDK, tavaszi fordulóra benyújtott **(Gépi tanulási algoritmusok fejlesztése a gének kópiaszám-változásainak pontos azonosítására)** című pályamunkámban leírtak esetében az alábbi munkafolyamatok elvégzésében vettem részt:

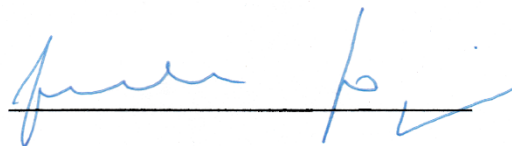
Részfeladat	A részvétel mértéke (ötfokozatú skálán)
Minták begyűjtése és szekvenálása	0
A szekvenált minták fastq fájljainak feldolgozása	5
Pályamunkához szükséges szkriptek megírása	5
In silico minták generálása	5
Modellek megírása, tanítása és kiértékelése	5
Predikció elvégzése ismeretlen adatsoron	5

Kelt: 2024.04.15.

A nyilatkozatban foglaltak helyességét megerősítem:



Pályamunkát benyújtó hallgató aláírása



Témavezető (név, munkahely) aláírása