Magyar Pszichológiai Szemle 79 (2024) 2, 171–193
DOI: 10.1556/0016.2024.00052

AKADÉMIAI KIADÓ

Felügyelt gépi tanulási módszerek alkalmazása a pszichológiai kutatásokban

Hajdú Nándor[1,2*], Szászi Barnabás[2], Aczél Balázs[2] és Nagy Tamás[2]

[1] ELTE Eötvös Loránd Tudományegyetem Pszichológiai Doktori Iskola, Budapest, Magyarország

[2] ELTE Eötvös Loránd Tudományegyetem Pszichológiai Intézet, Budapest, Magyarország

EREDETI KÖZLEMÉNY

Beérkezett: 2023. április 17. – Elfogadva: 2024. március 2.
Megjelent az interneten: 2024. június 18.

A pszichológiai kutatások számos területén hasznosítható a felügyelt gépi tanulás, amelynek alkalmazásával összetettebb adatok elemzése válik lehetségessé. Célunk, hogy bemutassuk a felügyelt gépi tanulás fajtáit, működését és használatát a pszichológiai kutatásokban. Áttekintjük a gépi tanulás előnyeit, valamint a túlillesztés, torzítási hiba és adatvariabilitás fogalmait, amelyek segítenek a modellválasztásban, és az eredmények robusztusságának biztosításában. Röviden bemutatjuk továbbá a legfontosabb felügyelt gépi tanulási algoritmusokat, és leírjuk a változók és adatok előkészítésének legfontosabb lépéseit. Egy példaelemzés keretében bemutatjuk, hogyan modellezhető egyetemi hallgatók lépcső és lift közötti választása felügyelt gépi tanulással. A cikk végén kitérünk a gépi tanulás korlátaira és annak helyére a pszichológusok oktatásában. Reméljük, hogy a bemutatott ismeretek segítenek a pszichológusoknak a gépi tanulás hatékonyabb és kreatívabb használatában.

KULCSSZAVAK

gépi tanulás, statisztika, feltáró adatelemzés, kutatásmódszertan

BEVEZETÉS

A gépi tanulás (machine learning, ML) az adattudomány, matematika, statisztika és számítástechnika határain elhelyezkedő, gyorsan fejlődő terület. Ez a cikk bevezetést nyújt a felügyelt gépi

* Levelező szerző. E-mail: hajdu.nandor@ppk.elte.hu

tanulás alapjaiba, beleértve a gyakori algoritmusokat és alkalmazásokat, valamint a gépi tanulás pszichológiai kutatásokban való alkalmazásának lehetséges előnyeit és korlátait. A cikket a felügyelt gépi tanulás meghatározásával és a területen használt főbb fogalmak és terminológia ismertetésével kezdjük. Ezután tárgyaljuk a felügyelt gépi tanulási algoritmusok leggyakrabban használt típusait, beleértve a regressziós módszereket, döntési fákat és neurális hálókat. Leírjuk a gépi tanulás folyamatát és legfontosabb lépéseit, és egy példán keresztül kitérünk a felügyelt gépi tanulás pszichológiai alkalmazására. Tárgyaljuk továbbá a gépi tanulás néhány kihívását és korlátját, például a nagy és változatos adathalmazok szükségességét, a túlillesztés lehetőségét és az eredmények árnyalt értelmezését. Végül kitérünk arra, hogy miért és hogyan lehetne a pszichológushallgatók képzésébe bevezetni a gépi tanulás módszertanát. Összességében célunk az, hogy alapvető ismereteket nyújtsunk a felügyelt gépi tanulásról és annak lehetséges alkalmazásairól a pszichológiatudományban, anélkül, hogy az olvasótól mély matematikai vagy programozási ismereteket várnánk el.

A gépi tanulás előnyei a pszichológiai kutatásokban

A gépi tanulás azoknak az adatfeldolgozási folyamatoknak a gyűjtőneve, amelyek során egy algoritmus egy optimalizációs folyamatot végrehajtva olyan modellt hoz létre, amely alapján képes előrejelzéseket alkotni (Jordan és Mitchell, 2015). Előnyei a pszichológiában leggyakrabban használt statisztikai eljárásokhoz képest, hogy (1) a predikció pontosságára vonatkozó kutatási kérdések tesztelése során elkerülhető a modell túlillesztése, valamint pontosabb és robusztusabb becsléseket tudunk adni a segítségükkel (Yarkoni és Westfall, 2017), (2) eredményei közelebb állnak ahhoz az igényhez, amit a laikusok és piaci szereplők megfogalmaznak a tudománnyal szemben, (3) és a hipotézistesztelési és feltáró elemzésektől eltérő módon informálhatják az elméletalkotást.

1. A klasszikus statisztikai eljárások a predikció helyett inkább a prediktorok szignifikanciájára és – jobb esetben – hatásméretére helyezik a hangsúlyt. A gépi tanulás eszközeivel azonban olyan kérdésekre is választ kaphatunk, amelyek az egyes vizsgált tényezőnek az adott elméletben, modellben betöltött szerepére vonatkoznak. Például ahelyett a kérdés helyett, hogy összefügg-e a depresszióval a közösségi média használata, feltehetjük úgy is a kérdést, hogy az összes mért viselkedéses elem közül – amelyek között a táplálkozástól, a munkahelyi stresszen át a szabadidős tevékenységekig számos dologról állnak rendelkezésre adatok – melyek alapján lehet leginkább arra következtetni, hogy valaki depressziós? Azt a kérdést is feltehetjük, hogy ezen tényezők figyelembevételével mennyire lehet pontos becslést adni arra vonatkozóan, hogy valaki depressziós-e. Ez utóbbi kérdésre válaszolhatnánk nem gépi tanulási statisztikai módszerekkel, azonban a gépi tanulás eszköztára lehetőséget arra, hogy elkerüljük a modellek túlillesztését, és ezáltal az eredményeink nagyobb mértékben lesznek általánosíthatók. Új adatokra pontosabb és robusztusabb becslések adhatóak a gépi tanulási modellek segítségével. A korábbi példánál maradva tegyük fel, hogy gépi tanulási módszerekkel illesztett modellt használunk arra, hogy új páciensek adatait elemezzük. Ez a becslésünk pontosabb és általánosíthatóbb lesz, mint ha nem tettünk volna lépéseket a túlillesztés elkerülésére. Ezeket a módszereket, amelyek segítenek a túlillesztés elkerülésében, és biztosítják a modell robusztusságát, a közlemény későbbi részeiben tárgyaljuk.
2. A tudomány feladata egyrészt jelenségek leírása és megmagyarázása, de az is, hogy az elméletek alapján képes legyen predikciót adni arra, hogy mi fog történni. Az utóbbi a

pszichológia tudományában kevésbé hangsúlyos, pedig ez is fontos eleme a mentális jelenségek megértésének. Azok, akik nem tudományosan közelítenek a pszichológiához, de szeretnék felhasználni az eredményeit, jogosan kérhetik számon a kutatóktól, hogy a modelljeik miért ilyen nehezen átültethetők a gyakorlatba. Számukra a legfontosabb, hogy lehessen adatok alapján pontos predikciókat adni a viselkedésre, vagy mentális jelenségekre vonatkozóan: mit fog csinálni, érezni, gondolni az adott személy, ha bizonyos tulajdonságai, viselkedése, stb. ismertek? Véleményünk szerint a gépi tanulási módszerek ismeretének hiánya a pszichológus kutatókat versenyhátrányba kényszeríti más tudományterületekkel és ipari szereplőkkel szemben, amelyek számára nagy mennyiségben hozzáférhetővé váltak adatok az emberek viselkedéséről, érzelmeiről, kommunikációjáról, és más, a pszichológia hatáskörébe tartozó jelenségekről (Vélez, 2021).

3. A gépi tanulási modellek alkalmasak lehetnek arra, hogy az elméletalkotást informálják. Olyan módon teszik ezt, hogy megmutatják, mely tényezők prediktívek a kérdéses jelenséget tekintve, és melyek azok a tényezők, amelyek nem. A nem prediktív elemeket a későbbi kutatások során nem szükséges mérni, figyelembe venni, a többivel kapcsolatban pedig megfogalmazhatunk hipotéziseket. Ezeknek a modelleknek a definiálására alkalmas a gépi tanulás. A gépi tanulás keretrendszere megfelelő minta esetén a predikciók robusztusságát és mintán kívüli általánosíthatóságát is garantálja.

Ebben a közleményben amellett érvelünk, hogy a gépi tanulási módszerek kiválóan alkalmazhatóak pszichológiai témájú kutatásokban. Az eddig felvázolt okok miatt indokoltnak tartjuk a gépi tanulásos módszereknek, azon belül is a felügyelt tanulási módszerek madártávlatból való bemutatását a pszichológia területéről származó példákkal illusztrálva. Célunk segíteni az érdeklődőknek abban, hogy megtalálhassák, milyen irányban folytatják a témában való elmélyülést.

A GÉPI TANULÁS FAJTÁI

A gépi tanulási módszereknek három főbb típusa van: a *felügyelt tanulás* (supervised learning), a *felügyeletlen tanulás* (unsupervised learning) és a *megerősítéses tanulás* (reinforcement learning). A felügyelt tanulásról akkor beszélhetünk, ha rendelkezésre áll egy címkézett adatállomány, amelyre a modellünket illesztjük, majd pedig egy addig nem elemzett tesztállományon megvizsgáljuk, hogy milyen a modell predikciós pontossága. Ebben a keretrendszerben a modell teljesítményének egyik mutatója lehet a predikciós pontosság, azaz hogy a korábbi adatok alapján illesztett modellünk milyen pontossággal jelzi előre az új adatok vizsgált tulajdonságait. Egy pszichológiában is gyakran használt statisztikai módszer, a lineáris regresszió is gépi tanulási módszernek tekinthető, hiszen egy bizonyos paraméterre, a négyzetes hibák összegére optimalizálunk: azt a modellt keressük a lehetséges lineáris modellek közül, amely esetén ez az érték a legkisebb. A felügyeletlen tanulás esetén nem tudjuk megmutatni az algoritmusnak, hogy mi a „jó” megoldás. Pszichológiában is gyakran használt nem felügyelt gépi tanulási módszer például a főkomponens-elemzés vagy a klaszteranalízis (lásd pl. Vargha, 2022). A harmadik gépi tanulási módszer a megerősítéses tanulás, ami a kívánt viselkedések jutalmazásán és a nem kívánatosak büntetésén alapul. Ennek során egy tanuló ágens képes cselekvéseket végrehajtani és visszajelzések alapján tanulni. Jelen közlemény a felügyelt tanulással foglalkozik, a másik két módszert nem érinti. A mindennapi nyelvben gyakran összekeverednek a gépi tanuláshoz és adattudományhoz

kapcsoló egyéb fogalmak, ezért az *1. táblázat*ban bemutatjuk az ebben a kontextusban leggyakrabban előforduló kulcsszavakat.

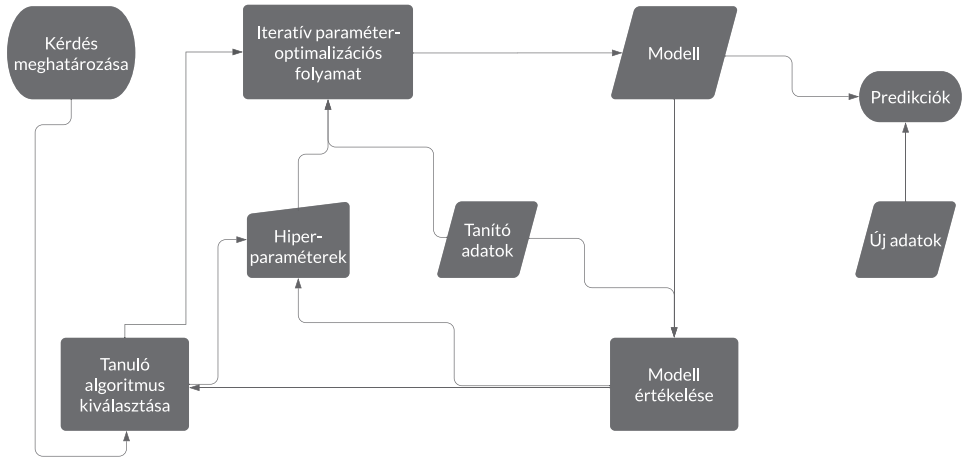
HOGYAN MŰKÖDIK A FELÜGYELT GÉPI TANULÁS?

A felügyelt gépi tanulás folyamatát az *1. ábrá*n szemléltetjük. Először választunk egy tanuló algoritmust, amelyet alkalmazni fogunk, majd ha ennek az illesztéséhez szükség van hiperparaméterekre, ezeket is megadjuk. Ezután az algoritmus a tanítóadatokkal egy iteratív folyamaton halad végig, amelynek a végére megkapjuk a modell optimális paramétereit. Ezt a modellt értékeljük, majd ha szükséges, megváltoztatott hiperparaméterekkel megismételjük az előző lépéseket, amíg a legoptimálisabb modellt nem kapjuk. Ezután megvizsgáljuk, hogy ez a modell hogyan teljesít új adatok esetén, milyen predikciókat ad. A továbbiakban ennek a folyamatnak az egyes lépéseit tárgyaljuk részletesebben, hangsúlyt fektetve azokra az elérni kívánt tulajdonságokra, amelyek indokolják az egyes lépéseket.

1. táblázat. Gépi tanuláshoz kapcsolódó gyakori fogalmak gyűjteménye

Fogalom	Meghatározás	Példa
Adattudomány (data science)	Az a tudományterület, amely egyesíti a programozási készségeket, a statisztikai ismereteket, valamint szakterületi tudást, hogy adatok elemzése által segítse a vizsgált jelenségek megértését.	Egy ország online médiájának tanulmányozásához algoritmikusan összegyűjteni a cikkeket, ezeket automatikusan témák szerint csoportosítani, és megállapítani, hogy mely témák aránya növekedett az utóbbi 5 évben.
Gépi tanulás (machine learning)	Számítógépes algoritmusok alkalmazása adatokon alapuló modellek létrehozására, előrejelzések készítésének céljából.	Egy írott szöveg alapján számszerű becslést adni arra, hogy az író mennyire van kitéve szuicid fenyegetettségnek.
Mesterséges intelligencia (artificial intelligence)	Egy digitális számítógép vagy számítógép-vezérelt robot azon képessége, hogy az intelligens lényekhez általában társított cselekvést végezzen.	Szöveg alapján kép generálása. Útvonal tervezése a kiindulási és végpont alapján.
Big data	Olyan adat, amely a mérete (pl. terabyte terjedelem), beérkezési sebessége (pl. valós idejű streaming), és/vagy struktúrája (nyers, strukturálatlan) miatt speciális módon dolgozható csak fel.	Egy város kültéri kameráinak folyamatosan beáramló felvételei, amelyek valós idejű érzelemfelismerést tesznek lehetővé.
Számítástudomány (computational science)	Egy tudományterület komplex kérdéseinek matematikai és számítástechnikai eszközökkel való vizsgálata.	Tudományos modell vizsgálata szimuláció használatával.

Források: Copeland (2022), Robinson (2017), Sagiroglu és Sinanc (2013).

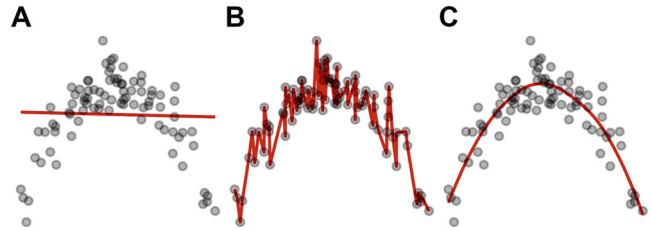


1. ábra. A felügyelt gépi tanulás folyamatának ábrázolása
Megjegyzés: A hengerek a folyamat kezdetét és végét, a téglalapok valamilyen folyamatot, a rombuszok bemeneti adatot vagy eredményt, a trapézok pedig manuálisan beállítandó bemenetet jelentenek, a folyamatábrákon megszokott jelölésnek megfelelően. A felügyelt gépi tanulás során a tanuló algoritmus a tanító adatok segítségével egy iteratív folyamaton keresztül megtalálja az optimális modell paramétereket. A modell predikciós hatékonysága alapján még további hiperparaméter-állítás történhet, egészen addig, amíg a modell hatékonysága javítható. Ezután a modell használható predikciók alkotására új adatok alapján.

Modellválasztás: túlillesztés, modelltorzítási hiba, adatvariabilitás

Ahhoz, hogy predikciókat tudjunk adni a kimeneti változónkkal kapcsolatban, először a vizsgált jelenség matematikai modelljét kell létrehozni. A gépi tanulás során a tanuló algoritmus egy iteratív folyamat által találja meg a modell optimális paramétereit. Például a lineáris regresszióban az OLS algoritmus megtalálja az optimális regressziós egyenes tengelymetszetét és meredekségét. Ezek a paraméterek írják le azt az összefüggést, ahogy a bemeneti változók prediktálják a kimeneti változó értékét. A gépi tanulást kétféle feladatra használhatjuk: regresszióra és osztályozásra (klasszifikáció). A regresszió olyan felügyelt tanulási feladatokat jelent, ahol a cél egy folytonos számszerű érték előrejelzése (pl. egy ember depresszió pontszámának becslése néhány viselkedéses jellemző alapján). Ezzel szemben az osztályozás során az algoritmus megtanulja a megfigyeléseket kategóriákba sorolni (pl. az előző példánál maradva azt megbecsülni, hogy az illető klinikai értelemben depressziós-e, vagy nem).

Minden modell egyszerűsítésen alapul, így tökéletlen leképezése a valóságnak. Egy modell lehet túlságosan érzéketlen az adatokban lévő összefüggések leképezésére (bias vagy modelltorzítási hiba), míg az is probléma, ha a modell túl érzékeny az adatokban lévő „zajra" (variance, azaz modellvariabilitási hiba). Ezen hibák közötti különbségeket a *2. ábrá*n szemléltetjük. Modelltorzítási hiba alatt azt értjük, hogy a modellezéshez használt algoritmus működésmódja korlátozza a lehetséges megoldások terét, és ezáltal azok pontosságát. A torzítási hiba a modellünk várható előrejelzése és a helyes, prediktálni szándékozott érték közötti különbség (Hastie, Friedman és Tibshirani, 2009). Képzeljük el, hogy az egész modellépítési folyamatot többször is megismételjük:



2. ábra. A modell torzításából fakadó és az adatok varianciájából fakadó hibák, valamint az optimális modell ábrázolása

Megjegyzések: Az alulillesztett modell (A) a saját egyszerűsége miatt nem képes jól megragadni a változók közötti összefüggést. A túlillesztett modell (B) a bonyolultsága miatt pontosan prediktálja a kimeneti változót, azonban a mintán kívül rosszul általánosítható. Az optimális modell (C) csak annyira bonyolult, hogy még viszonylag kis hibával képes legyen leírni a változók közötti összefüggést, ami a mintán kívül is pontos predikciókat tud adni.

ugyanabból a populációból új mintát gyűjtünk, új elemzést végzünk, új modellt hozva létre. Az alapul szolgáló adathalmazok véletlenszerűsége miatt az így kapott modellek különböző predikciókat fognak adni. A torzítási hiba azt mutatja, hogy ezen modellek előrejelzései általánosságban mennyire térnek el a helyes értéktől. A variancia miatti hiba a modell predikciójának egy adott adatpontra vonatkozó változékonyságát jelenti. Ismét képzeljük el, hogy a teljes modellépítési folyamatot többször is megismételjük. A variancia az, hogy egy adott pontra vonatkozó predikciók mennyire térnek el az ugyanabból a populációból különböző mintákra illesztett modellek között.

Egy lineáris modell például csak nagy pontatlansággal, vagy egyáltalán nem képes nemlineáris összefüggések leírására. Tehát ebben az esetben arról van szó, hogy a modell túlságosan leegyszerűsítve próbálja leképezni a változók közötti kapcsolatokat, és emiatt a rugalmatlanság miatt nem képes az összefüggések valódi természetének leírására – ezt a jelenséget alulillesztésnek (underfitting) hívják.

Ezzel szemben az adatok variabilitása esetében ennek az ellenkezője a probléma. Az adatok variabilitása miatti hiba egy modellnek a mintában megfigyelt változatosságára való túlzott érzékenységét jelenti. Ebben az esetben a mintában lévő zaj (pl. mintavételezési hiba) is beépül a modellbe, ami így túl bonyolulttá válik. Következésképpen a mintán belül magas lesz a megmagyarázott variancia aránya, azonban a mintán kívülre nem lesznek általánosíthatók a predikciók. Ezt hívják túlillesztésnek (overfitting).

A modellek építése során e két hibatípus között kell egyensúlyoznunk: tisztában kell lennünk a választott tanuló algoritmus korlátaival, amelyek hajlamosíthatják a torzításból vagy varianciából fakadó nem optimális modell létrehozására. Míg a modelltorzítási hiba kiküszöböléséhez gyakran más tanuló algoritmust kell választani, a túlillesztés érzékelésére és megakadályozására számos technika létezik, amelyek gyakran rutinszerű részét képezik a gépi tanulási munkamenetnek.

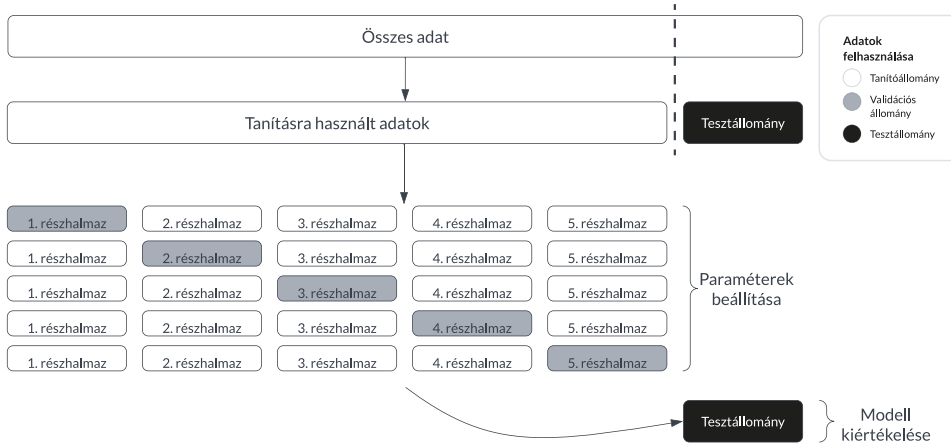
A hiperparaméter a modell azon jellemzője, amelynek értéke a tanulási folyamat szabályozására szolgál. Ezeket az algoritmusnak kívülről kell kapnia, azaz mi határozzuk meg őket. A pszichológiában is ismertebb példa a hiperparaméterre a klaszterek száma a klaszterelemzésben. Ezt nekünk kell előre meghatároznunk, és az algoritmus ebből kiindulva kategorizálja az eseteket. Egymás után több klaszteres megoldást kipróbálva lemérhetjük, hogy javult-e a modellünk, így lehetőség van a hiperparaméter fokozatos állítására és optimalizálására (hyperparameter tuning).

Mintán kívüli valid predikciók biztosítása

A folyamat több lépését is azért végezzük el, hogy biztosítsuk azt, hogy a mintánkon kívüli adatokra is a leginkább valid predikciókat tudjunk adni. Ilyen lépés a kiinduló adatállomány részekre osztása, ennek a részekre osztásnak egy speciális formája, a keresztvalidáció, valamint az újra- vagy alulmintavételezés.

A mintán kívüli érvényesség biztosítása. Annak érdekében, hogy elkerüljük a túlillesztést, és ne becsüljük felül a modellünk pontosságát, az adatainkat szétválasztjuk tanító- és tesztállományokra, vagy tanító-, teszt- és validációs állományokra. A tanítóállomány az adataink nagy többsége, amelyen tanítjuk a modellünket, azaz megtaláljuk azokat a paramétereket, amelyekkel a legjobb illeszkedést mutatja az adatokra. Azt, hogy mennyire pontos ez a modell a tanulás során nem látott adatok esetén, onnan tudjuk meg, hogy a tesztállomány alapján adott képzett predikciók mennyire pontosak. Például egy többszörös lineáris regressziós modellt a tanítóállományon tanítunk, azaz kiszámítjuk a regressziós együtthatókat (paramétereket), ezáltal meg tudjuk mondani, hogy a független változók adott értékénél a modell alapján a függő változó milyen értékét várjuk. A következő lépés az, hogy a modellünkbe behelyettesítjük a tesztállományban szereplő független változók értékeit, és kiszámítjuk, milyen értékeket prediktál a modellünk a függő változóra. Ezeket a predikciókat aztán összehasonlítjuk a tesztállományban meglévő függő változónk értékeivel, valamilyen veszteségfüggvény (loss function) segítségével, pl. átlagos abszolút eltéréssel (MAE, Mean Absolute Error), vagy négyzetes középértékkel (RMSE, Root Mean Squared Error). A prediktált és a mért értékek különbsége adja meg a modellünk pontosságát: *mennyire működik eddig még nem ismert adatokon a modellünk,* azaz *mennyire jól általánosítható?* Jó gyakorlat továbbá a folyamatot tovább bontani úgy, hogy a pontosságot egy validációs állományon vizsgáljuk meg, majd azt a modellt, amelyik ez alapján a legjobbnak tűnik, a tesztállománnyal is megvizsgáljuk (Hastie és mtsai, 2009). Szerzőnként és publikációnként eltér, hogy e két állomány közül melyiket nevezzük validációs állománynak, és melyiket tesztállománynak; sok esetben ezek a mi példánkhoz képest felcserélve szerepelnek. A közleményben mi az előbb leírt módon hivatkozunk az egyes adatállományokra. Konkrét példán keresztül szemléltetve a folyamat az alábbi módon zajlik: lineáris regressziós modellt illesztünk 10 független változóval, 300 fős mintán, aztán megvizsgáljuk a predikciós pontosságot egy 50 fős tesztállományon. Ugyanilyen módon illesztünk egy modellt 11 független változóval, amelyek közül 10 az előző folyamattal megegyezik. Ennek a modellnek is megvizsgáljuk a pontosságát a validációs állományon, és amelyik pontosabb becslést tud adni a függő változó értékéről, egyedül annak a pontosságát vizsgáljuk meg az 50 fős tesztállományon, amelyet eddig nem használtunk. Végül az ezen a tesztállományon kiszámított jellemzőket jelentjük elemzésünk eredményeként.

Keresztvalidáció. A tanító- és tesztállományra való bontás még kifinomultabb variációja az ún. keresztvalidáció, amely egy modell teljesítményének értékelésére szolgál az adatok különböző részhalmazain való modellillesztéssel és a fennmaradó részhalmazokon való teszteléssel. Ezt a módszert arra használjuk, hogy megbecsüljük a modell teljesítményét a még nem látott adatokon, és hogy megelőzzük a túlillesztést. A keresztvalidációnak több típusa létezik, ezek közül a leggyakrabban használt a k-szoros keresztvalidáció. Ennek során az adatokat k részhalmazra osztjuk, majd a modellt k alkalommal illesztjük, minden alkalommal egy másik részhalmazt használva validációs állományként, a többi részhalmazt pedig tanítóállományként. A modell teljesítményét ezután átlagoljuk a k ismétlés során. A k-szoros keresztvalidáció folyamatát a *3. ábra* szemlélteti. Egy másik elterjedt módszer az egy kihagyásos keresztvalidáció,



3. ábra. k-szoros keresztvalidáció használata a mintán kívüli predikciós érvényesség biztosításához
Megjegyzések: Az adatok szétbontása tanító- és teszthalmazokra. A tanítóadatok azután k-szoros keresztvalidációval használhatók a modell (pl. hiperparaméterek) finomhangolására. A példában látható ötszörös keresztvalidáció öt különböző módon bontja szét az adatokat tanító és tesztelő részekre. Az öt tanító részhalmaz alapján létrejövő öt modellt a teszthalmazokon tesztelve elkerülhető a túlillesztés. A modell végső értékelése a validációs halmazon történik.

amely hasonló a k-szoros keresztvalidációhoz, de ebben az esetben egy adatpont kivételével az összes adatpontot tanítóállományként, a fennmaradó adatpontot pedig validációs állományként használják fel. Ezt a számítást megismétlik úgy, hogy minden egyes adatpont pontosan egyszer legyen a tesztállomány, majd ezen modellek hibáinak átlagolásából kapjuk meg a keresztvalidált hibamutatót.

Mintavételezési eljárások. Abban az esetben, ha a mintánkban kategorikus változók szintjei között nagy számbeli különbség van, akkor ún. osztálykiegyensúlyozatlanság (class imbalance) jön létre. Ez azért jelentős probléma, mert ha a tanító állományban csak nagyon kevés példa van egy-egy kategóriára, akkor az algoritmus – a ritka előfordulásnak megfelelően – csak nagyon ritkán fogja ezt a kategóriatagságot prediktálni. Ez a probléma többféleképpen kezelhető. Az egyik lehetőség, hogy kidobjuk azon megfigyeléseinket, amelyek a gyakoribb kategóriába tartoznak. Ilyenkor véletlenszerűen tartunk meg megfigyeléseket, úgy, hogy végül az egyes kategóriák elemei kiegyenlítettek legyenek (downsampling). Ekkor azonban elveszítjük az adataink jelentős részét, ami az előrejelzés pontosságát fogja csökkenteni, vagy a modell túlillesztetté válhat (Yarkoni és Westfall, 2017). Egy másik lehetőség, hogy annak a kategóriának az elemeiből, amelyből aránylag kevés van, úgy veszünk mintát, hogy ugyanazt az elemet többször is választhatjuk (over-sampling). Ennek a továbbfejlesztett formája, amikor a kisebbségi kategóriából ún. szintetikus adatokat hozunk létre, ezzel kiegyensúlyozva a kategóriatagságokat. Ennek az egyik módja az ún. SMOTE algoritmus (Synthetic Minority Over-sampling Technique), amely k-legközelebbi szomszéd algoritmussal számít távolságokat a kisebbségi osztály egyes tagjai között, majd ezen távolságvektorok alapján képez új, szintetikus adatpontokat.

Adatainkon rétegzett mintavételt is alkalmazhatunk (stratified sampling). Ez biztosítja, hogy a minta bizonyos jellemzők vagy változók tekintetében reprezentálja a sokaságot. A módszer

a sokaságot bizonyos jellemzők alapján rétegekre vagy alcsoportokra osztja, majd az egyes rétegekből a sokaságon belüli méretük arányában vesz mintát. Ez biztosítja, hogy a minta reprezentatív legyen a sokaságra nézve a releváns feature-ök tekintetében. A rétegzett mintavétel hasznos lehet olyan esetekben, amikor a populációban bizonyos jellemzők nem egyenletes eloszlásúak, és fontos annak biztosítása, hogy a minta reprezentatív legyen a populációra nézve.

LEGFONTOSABB FELÜGYELT GÉPI TANULÁSI ALGORITMUSOK

A felügyelt gépi tanulási folyamat általános áttekintése és az általánosíthatóság biztosításának eszközei után – a teljesség igénye nélkül – rövid áttekintést nyújtunk a leggyakrabban használt felügyelt gépi tanulási módszerekről (Hindman, 2015). Számos gépi tanulási algoritmus létezik, amelyek különböznek működési módjukban, erőforrás-igényességükben, és hatékonyságukban. A problémakör, a rendelkezésre álló adatok és az eredmény elvárt tulajdonságai alapján érdemes választanunk ezek közül. Az alábbiakban felsorolt módszerek, vagy azok változatai egyaránt alkalmasak osztályozási és regressziós problémák megoldására is.

Regresszió: lineáris, generalizált lineáris, regularizált

A legismertebb, pszichológusok által is gyakran használt gépi tanulási algoritmus a lineáris regresszió (Stanton, 2001). Ez az algoritmus egy egyenest (illetve több prediktor esetén egy hipersíkot) illeszt az adatokra több iteráción keresztül, hogy ezáltal megtalálja azokat a paramétereket, amelyek a legkisebb hibával tudják prediktálni a kimeneti változó értékét. Ez a módszer önmagában nem képes a prediktorok hatékony kiválasztására, és az algoritmus működéséből fakadóan sok prediktor esetén túlillesztés léphet fel. Az algoritmus általánosított formája az ún. generalizált lineáris modell (GLM; Stanton, 2001), amelynek lényege a modell általánosítása nemlineáris „kapocsfüggvényen" (link function; pl. logaritmikus, logit, probit) keresztül. Ez lehetővé teszi, hogy ne folytonos változó legyen a kimenet, hanem egyéb mérési szintű és különleges eloszlású változókat is modellezhessünk. Például modellezhetők megszámlált diszkrét mennyiségek (pl. epilepsziás epizódok száma egy héten), valószínűségek (pontosság a Stroop teszten), dichotóm adatok (szenved-e valaki alvászavarban vagy nem), sorba rendezhető ordinális értékek (5 fokú Likert-skálán a fájdalom mértéke) és diszkrét kategóriák (melyik pártra fog valaki szavazni).

A regresszió korlátja, hogy minél több prediktort használunk, annál inkább megnő a multikollinearitás és a túlillesztés esélye. Ennek a kiküszöbölésére hozták létre a regularizált regressziót (Tikhonov, Goncharsky, Stepanov és Yagola, 1995). Ezek a továbbfejlesztett algoritmusok egy „büntetőparaméter" segítségével mesterségesen lecsökkentik az egyes prediktorok hatását a kimeneti változóra, így lehetővé válik az egymással korreláló változók együttes vizsgálata. Ezek közül a módszerek közül az egyik az ún. ridge regresszió, amely többszörös regressziós modellek együtthatóinak becslésére szolgáló módszer olyan esetekben, amikor a független változók nagymértékben korrelálnak egymással (Hoerl és Kennard, 1970). Egy másik regularizált regresszió fajta, a LASSO (least absolute shrinkage and selection operator; Santosa és Symes, 1986) regresszió képes arra, hogy sok prediktor közül csak azokat tartsa meg, amelyek valóban fontosak a predikció szempontjából. Ezt szintén egy büntetéssel éri el, viszont a LASSO esetén végül csak a jelentős prediktorok maradnak a modellben, a többi prediktor együtthatója 0-ra csökken. A harmadik változata a regularizált regressziónak az ún. elastic net (Zou és Hastie, 2005), amely a ridge és a LASSO regressziók keveréke: mindkét korábban említett módszer

büntetőparaméterét használja. Ezek a modelltípusok jól használhatók számos különleges tulajdonságú kimeneti változó esetén is. A folytonos kimeneti változón túl a módszer alkalmazható bináris (kétszintű) és multinomiális (többszintű) klasszifikációra, valamint speciális eloszlású kimeneti változókra, mint az ordinális, megszámlált (count) kimenetek, továbbá létezik többszintű változata is, amely alkalmassá teszi hierarchikusan beágyazott adatok és ismételt méréses adatok elemzésére. Kombinálva a megfelelő prediktor transzformációkkal (pl. logaritmikus vagy polinomiális), képes nemlineáris kapcsolatokat is modellezni.

K-legközelebbi szomszéd (k-nearest neighbours, kNN)

A k-legközelebbi szomszéd algoritmus (Cover és Hart, 1967) egy olyan gépi tanulási algoritmus, amelyet regressziós és osztályozási feladatokra egyaránt használnak. Az algoritmus úgy működik, hogy a gyakorló adathalmazban megtalálja a tesztállomány adott adatpontjához a k-legközelebbi adatpontot. A teszthalmaz adott adatpontja és az összes tanítóállományban lévő adatpont közötti távolságot kiszámítjuk, és a k-legközelebbi adatpontot a tesztadatponttól való távolság alapján választjuk ki. A tesztadatpont prediktált értékét regressziós feladatok esetén a k-legközelebbi szomszéd átlagával, osztályozási feladatok esetén pedig a k-legközelebbi szomszédok közül a leggyakoribb osztály kiválasztásával határozzuk meg. A k értéke fontos paraméter a kNN algoritmusban, mivel túl kicsi érték esetén a predikció zajos lesz, túl nagy érték esetén pedig a modell túlillesztetté válhat.

Döntési fa alapú módszerek

A *döntési fák* (decision trees) a prediktorok jellemzői alapján a döntések faszerű modelljét hozzák létre (Breiman, Friedman, Stone és Olshen, 1984). A fa levelei (leaf) a kategóriacímkéket, a gyökértől a levelekig vezető ösvények pedig az osztálycímkékhez vezető döntéseket jelképezik. A fa minden egyes belső csomópontja, nódusa egy prediktorra vonatkozó „tesztet" jelent, és minden egyes ág a teszt eredményét. Például, hogy a szétbontás előtt és után hány elem kerül helyes kategóriába. Egy új mintára vonatkozó előrejelzés elkészítéséhez az algoritmus a gyökércsomópontból kiindulva végigjárja a fát, és a minta jellemzőinek értékei alapján döntéseket hoz, amíg el nem ér egy levélcsomópontot, amely maga a klasszifikáció.

A döntési fák viszonylag könnyen érthetők és értelmezhetők, és folytonos és kategorikus adatokat egyaránt képesek kezelni. Hajlamosak lehetnek azonban a túlillesztésre, különösen, ha a fa túl mélyre nő. A túlillesztés megelőzése érdekében a döntési fákat meg lehet metszeni (pruning), hogy eltávolítsák azokat az ágakat, amelyek nem javítják jelentősen a pontosságot. A döntési fa alapú módszerek kiválóan alkamasak interakciók és nemlineáris összefüggések vizsgálatára. A döntési fáknak több továbbfejlesztett változata van, amelyek jellemzően több fa eredményeit összesítik. Ezek általában csökkentik a túlillesztést, amely egyetlen döntési fa képzésekor előfordulhat, és a végső modellt robusztusabbá teszik az adatokban lévő zajjal szemben.

Az *erősített fák* (*boosted trees;* Hastie és mtsai, 2009) sok gyenge tanuló (például egyetlen nódust tartalmazó döntési fák) előrejelzéseit kombinálják, hogy létrehozzon egy erős tanulót, amely pontosabb, mint bármelyik gyenge tanuló. A gyenge tanulók egymás után jönnek létre, és minden egyes tanuló megpróbálja kijavítani az előző tanuló által elkövetett hibákat. A végső előrejelzés az összes gyenge tanuló előrejelzéseinek kombinálásával készül.

A *véletlen erdő* (random forest; Breiman, 2001) nagyszámú döntési fát hoz létre a megfigyelések és prediktorok részhalmazain, és a végső előrejelzés az összes egyedi fa előrejelzéseinek átlagolásával készül. Ehhez hasonló a *bagged trees* módszer, aminek a neve a bootstrapped

aggregation szóból ered, amely több döntési fa létrehozását jelenti a megfigyelések véletlenszerű részhalmazain, ám a véletlen erdővel ellentétben az összes prediktort felhasználja a fák kialakításában, nemcsak egy véletlen részhalmazukat.

Neurális hálózatok (neural networks)

A neurális hálózatok a gépi tanulási algoritmusok egy olyan típusa, amelyet az emberi agy szerkezete és működése ihletett. Ezek a hálózatok egy vagy több rétegben összekapcsolt mesterséges neuronokból állnak, amelyek feldolgozzák és továbbítják egymásnak az információkat. A létrehozása során a hálózat a neuronok közötti kapcsolatok súlyait úgy állítja be, hogy minimalizálja a hibát a megjósolt kimenet és a valódi kimenet között. A neurális hálózatok története az 1940-es években kezdődött, amikor McCulloch és Pitts (1943) megalkotta a mesterséges neuronok fogalmát. A Frank Rosenblatt (1958) által az 1950-es évek végén kifejlesztett, és valószínűleg a pszichológusok számára is ismerős perceptron döntő mérföldkövet jelentett. Ez a hálózat bináris bemeneti adatok alapján bináris osztályozásra képes. A bemeneti adatokat megszorozzuk a súlyokkal, amelyek a bemenet és a kimenet közötti kapcsolat erősségét jelölik. Minden bemenethez tartozik egy súly, amely a kimenethez való hozzájárulását jelenti. Ezek a súlyok olyan paraméterek, amelyeket a perceptron a tanítás során tanul meg. Ezután a bemenetet és a súlyokat összegezzük egy összegzőfüggvény segítségével: ez valójában a két számsor skaláris szorzata. Ennek eredményét egy ún. aktivációs függvénybe vezetjük, amely döntést hoz két kategória között: ha adott küszöbérték alatt van az érték, akkor az egyik kategóriába sorol az algoritmus, ha felette, akkor a másikba. A perceptron, bár némely osztályozási feladat esetében igen hasznos, az összetett problémák kezelésében mutatkoznak korlátai. Ezek a korlátok a neurális hálózatok iránti lelkesedés csökkenéséhez vezettek. Később a visszaterjesztéses algoritmussal (back-propagation), amely lehetővé tette a többrétegű hálózatok hatékony tanítását, újjáéledt az érdeklődés a neurális hálók iránt (Werbos, 1994). Ez megnyitotta az utat a kifinomultabb architektúrák, például a radiális bázisfüggvény (RBF; Broomhead és Lowe, 1988) hálózatok kifejlesztése előtt. Kapcsolódó fogalom az ún. deep learning (Deng és Yu, 2014), amely olyan neurális hálózatok gyűjtőneve, amelyeknek a bemeneti és kimeneti rétege között legalább egy rejtett réteg van. Az 1990-es években Yann LeCun és mások által bevezetett konvolúciós neurális hálózatok (CNN) forradalmasították a képfeldolgozási feladatokat (LeCun és mtsai, 1989). A Hochreiter és Schmidhuber által bevezetett hosszú rövidtávú memóriájú (LSTM) hálózatok az eltűnő gradiens problémáját oldották meg, növelve a szekvenciális adatok hosszú távú függőségének megragadására való képességét (Hochreiter és Schmidhuber, 1997). A 2010-es évek első felében csatlakozott a neurális hálózati algoritmusok gazdag palettájához a generatív adverzális hálózat (GAN; Goodfellow és mtsai, 2014), amelyben két neurális hálózat verseng egymással egy zéró összegű játszmában.

A neurális hálózatok egyik ígéretes új módszere a *transzformerek* (Vaswani, Shazeer, Parmar és Uszkoreit, 2017) használata. A transzformerek szekvenciális adatok, például természetes nyelvi adatok feldolgozására szolgálnak. Ilyen elven működik a GPT (generative pre-trained transformers) architektúra, amely szöveg-kép előállítást vagy emberihez hasonló, igen összetett szöveg generálását tesz lehetővé (pl. chatGPT). A transzformerek lényeges eleme a figyelemmechanizmusok használata, amely lehetővé teszi, hogy a modell a bemeneti szekvencia feldolgozása során szelektíven annak különböző részeire fókuszáljon. Ez ellentétben áll a hagyományos rekurrens neurális hálózatokkal, amelyek a bemeneti szekvenciát sorrendben és egyesével dolgozzák fel (Vaswani és mtsai, 2017). Összességében a neurális hálózatok a gépi tanulási feladatok erőteljes és széles körben használt eszközei, és számos alkalmazásban, például képosztályozási, természetes

nyelvi feldolgozási és előrejelzési feladatokban bizonyultak sikeresnek. A többi bemutatott algoritmushoz képest a neurális hálózatok tanításához nagyságrendekkel több adat szükséges.

MODELL VÁLTOZÓINAK ELŐKÉSZÍTÉSE – FEATURE ENGINEERING

Gépi tanulásos terminológiában az adatok bizonyos számszerűsíthető tulajdonságait *feature*-öknek nevezzük; ezeket a *feature*-öket jellemzően változókba rendezzük, amelyek az elemzés alapját képezik majd. A *feature engineering* az a folyamat, amely során területspecifikus tudást felhasználva a nyers adatokból további feldolgozásra alkalmas adatállományt állítunk elő, amely az állomány számunkra releváns tulajdonságait megfelelő formában tartalmazza. Célja az, hogy javítsuk a majdani modellünk predikcióját: azt várjuk, hogy az újonnan előállított *feature*-ök segítségével pontosabb predikciókat kapunk, mint ha a nyers adatokkal dolgoztunk volna. További célja a *Feature engineering*nek, hogy uniformizáljuk a feldolgozási lépéseket annak érdekében, hogy azok automatikusan alkalmazhatók legyenek a tesztállományra és az új adatokra is, amelyekre predikciókat szeretnénk alkotni.

Miután eldöntöttük, hogy a feltett kérdésünkre milyen algoritmus segítségével szeretnénk válaszolni, a *feature engineering* eszközeivel elő kell készítenünk az adatállományunkat: először ki kell számítanunk azokat a változókat, amelyeket fel szeretnénk használni a predikcióra, de még nem megfelelő formában állnak rendelkezésre; ezután lehetséges, hogy szintetikus adatokat kell létrehoznunk; majd lehet, hogy valamilyen módon transzformálnunk kell már meglévő változókat; utolsó lépésként pedig ki kell választanunk azokat a változókat, amelyeket fel fogunk használni a modellillesztés során. A jó predikciók és az erőforrás-hatékonyság érdekében a *feature engineering*et a kiválasztott tanuló algoritmushoz érdemes illeszteni. A továbbiakban rövid áttekintést adunk a legfontosabb *feature engineering* lépésekről. Ezek nagyobb lélegzetű összefoglalása olvasható Kuhn és Johnson (2019) könyvében.

Új változók kiszámítása

Szembesülhetünk azzal a problémával, hogy az eredeti adatállományunk nem olyan formában tartalmazza a változóinkat, amely számunkra a legpraktikusabb – például egymással erősen korreláló változókat találunk az adatállományban, amelyeket érdemes lenne főkomponens-elemzéssel kevesebb, egymással nem korreláló változóra redukálni. Ilyenkor megtehetjük, hogy a rendelkezésre álló változókat, vagy egy részüket összevonjuk kevesebb változóvá, annak érdekében, hogy az adataink valamilyen nekünk jelentősnek tűnő *feature*-t reprezentáljanak; vagy fordítva, több új változót is létrehozhatunk. Mi döntjük el, hogy számunkra milyen információ releváns az összes elérhető közül, és hogy azt milyen módon tesszük láthatóvá. A modellillesztés során számos esetekben azok a *feature*-ök, amelyek nem járulnak hozzá a predikciós pontosság növeléséhez, elhagyásra kerülnek. A gépi tanulás nagy előnye, hogy a folyamat bármely részét annyiszor végezhetjük el, ahányszor szükséges, nem kell tartanunk az újrateszteléstől. Így például összehasonlíthatjuk, milyen módon hatnak az általunk definiált *feature*-ök a predikcióra.

Hiányzó adatok kezelése

Gyakran előfordul, hogy az adatállományunkból hiányoznak egy-egy, vagy akár több változó értékei is. Ez azért jelenthet problémát, mert számos olyan modelltípus létezik, amelyek

illesztésénél kizárólag a hiánytalan megfigyeléseket vehetjük figyelembe; ahol akár egyetlen hiányzó adat is van, azzal az adatsorral már nem tudunk számolni. A *feature engineering* része az is, hogy a hiányzó adatok problémáját valamilyen módon orvosoljuk. Az adatimputáció az a folyamat, amelynek során a hiányzó értékeket egy adatkészletben plauzibilis értékekkel helyettesítjük. Ehhez számos módszer áll rendelkezésre. A középérték imputálás során a hiányzó értékeket a kipótolandó, hiányos *feature* átlagával, mediánjával vagy móduszával helyettesítjük. Ez a módszer egyszerűen megvalósítható és számítási szempontból hatékony, de torzításhoz vezethet az adatállományban, ha a hiányzó értékek nem véletlenszerűen hiányoznak (Lang és Little, 2018). A kNN (lásd az algoritmusok leírásánál) imputálás esetén a hiányzó értékeket a többi változó alapján kiszámított k leghasonlóbb elem alapján kiszámított értékkel helyettesíti. Többszörös imputálás során több imputált adatállományt hozunk létre egy statisztikai modell segítségével, amely a hiányzó adatokra plauzibilis értékeket generál. Ez a módszer robusztusabb, mint az egyszerű átlag/medián/módusz imputálás, mivel figyelembe veszi az imputált értékek bizonytalanságát.

Adattranszformációk

Egyes esetekben a változóink a számunkra érdekes *feature*-öket reprezentálják, de nem a legmegfelelőbb formában; azaz, tartalmilag helyes, de formailag helytelen adatok állnak rendelkezésre. Ilyenkor adattranszformációkat kell végeznünk. A kategorikus változók esetében például ún. dummy-kódolással létrehozhatunk új, numerikus változókat: n-fokú kategorikus változó esetén ugyanazt az információt n-1 bináris változóval kódoljuk, amelyek az adott kategóriába való tartozást, vagy nem tartozást reprezentálják. Folytonos változók esetén végezhetünk minimum-maximum *feature* skálázást: ez a leggyakrabban azt jelenti, hogy az adott változó értékeit 0 és 1 közötti értékekké transzformáljuk, olyan módon, hogy az adott változó minimumát kivonjuk az egyes értékekből, majd ezen különbséget elosztjuk a változó terjedelmével. Egy másik adattranszformációs eljárás folytonos változók esetében a centrálás, amikor az adott változó átlagát kivonjuk minden értékből; valamint a standardizálás, amikor centrálás után elosztjuk az egyes értékeket az adott változó szórásával. Bizonyos esetekben célszerű lehet a változó négyzetgyökét vagy logaritmusát venni, és ezt használni a későbbi számításokban. Abban az esetben, ha valamilyen szöveges adatot kell számszerűsítenünk, hasznos lehet n-gramokat készíteni: ez a technika hasznos a szövegosztályozási és természetes nyelvi feldolgozási feladatokban, mivel lehetővé teszi, hogy a modell a szavak kontextusának figyelembevételével megragadja a szöveg jelentését. Az n értékének megválasztása az adathalmaz és az adott probléma sajátosságaitól függ. A beágyazás (embedding) az adatok – gyakran szavak – ábrázolásának egy olyan módja, amely lehetővé teszi, hogy a gépi tanulási modellek és algoritmusok könnyen felhasználhassák azokat. A beágyazás egy szöveg szemantikai jelentésének kompakt, numerikus reprezentációja. Minden beágyazás egy számvektor, úgy, hogy a vektortérben két beágyazás közötti távolság korrelál az eredeti fogalmak közötti szemantikai hasonlósággal. Ha például két szöveg hasonló, akkor a vektoros reprezentációiknak is hasonlónak kell lenniük (Mars, 2022).

Feature kiválasztás

Miután adatainkból feature-öket alakítottunk ki, szimulációval kipótoltuk a hiányzó adatokat, és végrehajtottuk a megfelelő adattranszformációkat, a következő lépés annak a vizsgálata, hogy vajon minden egyes kialakított feature-re szükségünk van-e a predikcióhoz. A *feature*-ök

kiválasztása kulcsfontosságú lépés a *feature engineeringben*, mivel segít azonosítani a legrelevánsabb és leginformatívabb *feature*-öket, amelyek pontosan meg tudják ragadni az adatok mögöttes mintáit. Korábban említettük, hogy a kutatók sok esetben nem használnak minden változót, ami rendelkezésükre áll, annak érdekében, hogy egy egyszerűbb, jobban értelmezhető modellel dolgozhassanak. A társadalomtudományokban ráadásul kifejezetten előnytelen gyakorlatok is elterjedtek a szelekcióval kapcsolatban (Gigerenzer, 2004; Thiese, Arnold és Walker, 2015); ilyen például a lépésenkénti (stepwise) regresszióanalízis is (Harrell, 2001). A gépi tanulás keretrendszerének fontos lépése a prediktorok közötti szelekció, amely a modellkomplexitás kérdésére is megoldást jelenthet. A *feature* kiválasztás célja az adatok dimenzionalitásának csökkentése és a redundáns vagy irreleváns jellemzők eltávolítása, ami javíthatja a modell teljesítményét, valamint könnyebben értelmezhetővé teheti azt. Például, ha személyiségvonások alapján szeretnénk a személyek viselkedésére predikciót adni, lehetséges, hogy redundáns lesz ugyanannak a vonásnak három különböző kérdőív alapján kiszámított értéke. Az is lehet, hogy mivel kicsit más szemszögből vizsgálják ugyanazt a jelenséget, mindhárom értékes információt tartalmaz, és ezeknek a megtartása javítja a predikciót. A *feature* kiválasztás az a folyamat, amelynek a végére el tudjuk dönteni, a rendelkezésre álló *feature*-ök közül melyekre van szükségünk a legjobb predikció eléréséhez.

Három nagy csoportba sorolhatjuk ezeket a kiválasztási módszereket: léteznek szűrő módszerek, *feature* részhalmaz-kiválasztó (wrapper) módszerek, valamint beágyazott módszerek (Jović, Brkić és Bogunović, 2015). A szűrő módszerek a jellemzők statisztikai tulajdonságain alapulnak, mint például a korreláció vagy a közös információtartalom, és függetlenek a modelltől. Az előre meghatározott határérték alatti pontszámot elérő *feature*-ök kiesnek, míg a magasabb pontszámot elérők kiválasztásra kerülnek. A *feature*-ök ezen részhalmaza ezután bemenetként megadható a választott algoritmusnak. Modellfüggetlenségük mellett nagy előnye a szűrő módszereknek, hogy kevésbé számításigényesek, mint a többi módszer, és ezáltal könnyebben skálázhatók magas dimenzionalitású adatállományra is. A *feature* részhalmaz-kiválasztó módszerek egy modellt használnak a különböző *feature*-csoportok predikciós teljesítményének értékelésére, és kiválasztják azt a *feature*-csoportot, amely a legjobb predikciós teljesítményt eredményezi. A *feature* részhalmaz-kiválasztó módszerek fő előnye, hogy figyelembe veszik a jellemzők és a modell közötti interakciókat, ami nagyobb pontosságot eredményezhet. Hátrányuk viszont, hogy nagy számítási igényük van, és az ilyen módon kiválasztott *feature*-ök más algoritmus esetén nem biztos, hogy a leghasznosabbak lesznek; például, ha egy *feature* részhalmaz-kiválasztó módszerrel iteratív módon kiválasztjuk azokat a *feature*-öket, amelyek egy lineáris regressziós modellben a legjobb prediktorok, akkor azok lehet, hogy egy véletlen erdő modell esetén sokkal rosszabb predikciót fognak adni (a szűrő módszerek esetén sem lehet garantálni, hogy minden modellben minden prediktor ugyanolyan hasznos lesz, de kisebb a különbség, mint ha *feature* részhalmaz-kiválasztó módszerrel egy adott típusú modellre optimalizáltuk volna a *feature*-ök kiválasztását. Ilyen *feature* részhalmaz-kiválasztó módszer például az előre irányuló szelekció (forward selection) és a visszafelé irányuló szelekció (backward selection). A beágyazott módszereknél a *feature* választás a modellalgoritmusba van integrálva. A tanítás során a modell beállítja belső paramétereit, és meghatározza az egyes *feature*-öknek adott megfelelő súlyokat a legnagyobb pontosság elérése érdekében. Ezért a beágyazott módszerben az optimális *feature* részhalmaz keresése és a modell felépítése egyetlen lépésben egyesül (Guyon, 2003). Ilyen módszer a regularizáció általi *feature* kiválasztás is, például a korábban említett LASSO modellek esetében.

PÉLDA ELEMZÉS: EGYETEMI HALLGATÓK LÉPCSŐ ÉS LIFT KÖZÖTTI VÁLASZTÁSÁNAK MODELLEZÉSE

A továbbiakban bemutatunk egy elemzést, amely során felügyelt gépi tanulási módszereket használtunk: Célunk az volt, hogy feltárjuk, milyen kontextuális tényezők befolyásolják a lépcső és a lift közötti választást, amikor az emberek egy felsőbb emeletre szeretnének eljutni, és a két opció közül mindkettő rendelkezésre áll. Ez az elemzés az alapját képező adatokkal együtt elérhető a GitHubon.[1]

Módszer

A kutatásban 523 (346 nő) legalább 18 éves egyetemi hallgató (M_{kor} = 21,9 év, SD_{kor} = 3,3) vett részt alanyként, akiket e-mailes hirdetés útján toboroztunk, és akik kompenzációként kurzus kreditpontokat kaptak. Az adatállomány egy részét egy másik publikációnkban is felhasználtuk (Hajdu, Szaszi és Aczel, 2020). A résztvevők egy e-mailben kapott linken keresztül férhettek hozzá online kérdőívünkhöz. A kérdőívben először arra kértük őket, hogy jelezzék, hogy az előző napon meglátogatták-e az egyetemi épületek valamelyikét. Ezután azt kérdeztük tőlük, hogy melyik épületben jártak utoljára, valamint hogy a liftet vagy a lépcsőt választották-e az emeletre való feljutáshoz. Ezután pedig arra kértük őket, hogy jelöljék meg a választás időpontjában a különböző fennálló kontextuális tényezők paramétereit. Ezek a kontextuális tényezők egy korábbi kutatás eredményei: *Társak választása, Célemelet, Környezettudatosság, Lustaság, Egészség, Lift sebessége, Feljutás sebessége, A liftre várakozók száma, Csomag, Egyes opciók vonzósága, Fáradtság* és *Hőmérséklet*. Például a *Csomag* faktor esetében egy 11 pontos Likert-típusú skálán (egyáltalán nem értek egyet – teljes mértékben egyetértek) kellett megjelölniük egyetértésük mértékét azzal az állítással, hogy a választás időpontjában nehéz poggyászt cipeltek. Kivételt képez a *Társak*, amely egy 3 választási lehetőséget tartalmazó item volt, ahol a résztvevők azt jelezték, hogy a lépcső és lift közötti választás idején 1) egyedül voltak, 2) társaik a liftet választották, vagy 3) társaik a lépcsőt választották.

A résztvevők jelezték, ha bármilyen fizikai vagy mentális állapot akadályozta őket abban, hogy a lépcsőt vagy a liftet használják (pl. sérülés, klausztrofóbia). Úgy döntöttünk, hogy nem vesszük figyelembe azoknak az adatait, akik arról számoltak be, hogy fizikai vagy pszichés állapotok korlátozzák őket a döntésükben, mert válaszaik alacsony variabilitást mutattak volna. A résztvevők tíz egymást követő hétköznapon kapták meg ugyanazt a kérdőívet, és mindig aznapra vonatkozóan kellett kitölteniük azt.

Eredmények

Az adatokon először szűrést végeztünk: egyedül azon személyekhez tartozó adatokat tartottuk meg, akik legalább nyolc napon adtak választ a kérdéseinkre. Ezt követően 75–25%-os arányban osztottuk fel tanító- és tesztállományokra, oly módon, hogy minden, ugyanahhoz a résztvevőhöz tartozó megfigyelés kizárólag az egyik állományban szerepelt. Így a tanítóállományban 2689, míg a tesztállományban 899 megfigyelés volt. A tanítóállományt a modell betanítására használtuk, a teszthalmazt pedig arra, hogy lássuk, mennyire jól teljesít a modell új adatokon.

[1] https://github.com/nandorhajdu/ml-for-psych-2022.

A tanítóállományban szereplő választások 61,7%-a lift volt. Ezt a lift- és lépcsőválasztás közti egyenlőtlenséget nem tartottuk olyan mértékűnek, hogy kezeljük. A hiányzó adatokat imputálással egészítettük ki: a *Társak* változó értékei közül 221 hiányzott. Ezeket móduszimputációval pótoltuk, azaz a hiányzó adatok helyére a *Társak* változó módusza került. Ezenkívül még egy változónál, az *Egészség* esetén találtunk hiányzó értékeket: itt 1390 megfigyelés hiányzott, ami a tanítóállomány összes megfigyelésének 51,7%-a. Ezt úgy kezeltük, hogy k-legközelebbi szomszéd imputációt alkalmaztunk, és k-nak 5-öt választottunk, azaz az egyes hiányzó *Egészség* adatpontokat úgy pótoltuk, hogy megvizsgáltuk a többi változó alapján az 5 legközelebbi szomszédot, és ezen szomszédok *Egészség* értékei közül a leggyakoribb értéket helyettesítettük be az adott hiányzó *Egészség* értékhez. A folytonos változóinkat ezek után standardizáltuk.

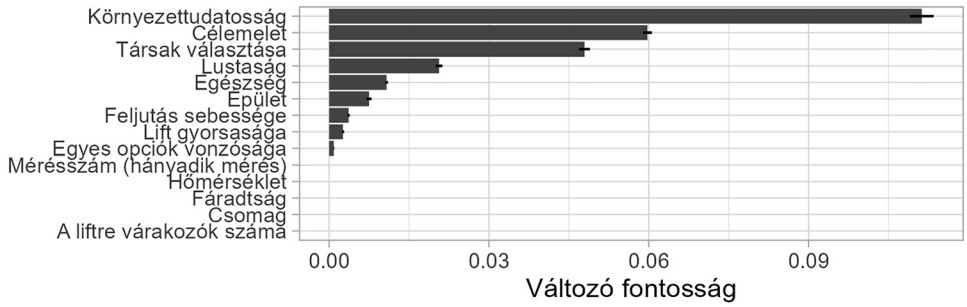
Ezután, annak megállapítására, hogy a kontextuális tényezők milyen mértékben befolyásolják a lépcső és a lift közötti választást, egy olyan modellt határoztunk meg, amely vegyíti a kevert hatású modellek képességét az ismételt mérések elemzésére és véletlenszerű hatások figyelembevételére, és a döntési fák képességét az interakciós összefüggések kezelésére: egy generalizált lineáris kevert hatású regresszió fa modellt (Fokkema, Smits, Zeileis, Hothorn és Kelderman, 2018). A módszer további előnye, hogy az ismételt méréseket, azaz a nem független varianciákat figyelembe tudja venni. A lépcső és a lift közötti választás volt a függő változó, a mért kontextuális tényezők pedig a független változók. A résztvevőket véletlen hatásként kezeltük.

Meg szerettük volna becsülni, hogy a modell mennyire jól magyarázza az egyének választásainak eltéréseit. Ehhez a tanítóállományon a lépcső vagy a lift választásának a modell által megmagyarázott varianciáját becsültük meg a prediktált értékek és a mért értékek közötti négyzetes korrelációs együttható kiszámításával, $R^2 = 0{,}91$. A résztvevőkből adódó, véletlenszerű hatást a modellünkben konstans tagokkal fedtük le. A modellünknek fontos eleme, hogy számoltunk azzal a hatással, hogy bizonyos adatpontok ugyanattól a személytől származnak, $M_{random} = -0{,}044$, $SD_{random} = 1{,}454$.

Annak vizsgálatára, hogy melyik változó járul hozzá a legnagyobb mértékben a predikciós pontossághoz, permutáció fontosság pontszámokat számoltunk ki. Ezek az értékek esetünkben azt mutatják meg, mennyivel csökken az ROC görbe alatti terület (ROC AUC), ha az adott változó értékeit összekeverjük, azaz megszüntetjük prediktív értéküket a függő változóra vonatkozóan. Eredményeink azt mutatják, hogy a legfontosabb változók a modellünk predikciójában a *Környezettudatosság,* a *Célemelet,* valamint a *Társak választása*. A permutációs fontosság pontszámokat a *4. ábrá*n szemléltetjük.

Megvizsgáltuk, hogyan teljesít a modell olyan adatok esetén, amelyek nem képezték a tanítóállomány részét, azaz új, ismeretlen adatokon. A modell predikcióit a tesztadatokon összehasonlítottuk a valós döntésekkel, hogy lássuk a modell pontosságát. Valószínűségi küszöbértéknek 0,5-et választottunk, ahol a 0,5-nél nagyobb előre jelzett valószínűségeket úgy kategorizáltuk, hogy valaki inkább a lépcsőt választotta, mint a liftet. Az eredmények azt mutatták, hogy a modell az új esetek 85%-át kategorizálta helyesen a tesztállományban. A tévesztési mátrixot a *2. táblázat* szemlélteti, az egyéb pontosságmutatókat a *3. táblázat*ban közöljük.

Az ROC görbe (*5. ábra*) vizsgálata alapján a modell jól teljesít a lépcsőzés és a liftezés predikciójában, a görbe alatti terület AUC = 0,91. A szenzitivitás alacsonyabb, mint a specificitás, ami azt jelzi, hogy a modell a lépcsőzést pontosabban prediktálja, mint a liftezést.



4. ábra. A prediktorok Permutációs fontosság pontszámai
Megjegyzés: Az oszlopok a változók átlagos fontosságát mutatják tíz permutáció alapján kiszámolva. A hibasáv az átlag standard hibáját jelzi. Azon változók, amelyekhez nem tartozik oszlop, nem járultak hozzá a modellhez.

2. táblázat. A generalizált lineáris kevert hatású regresszió fa modell tesztállományra adott predikcióinak tévesztési mátrixa

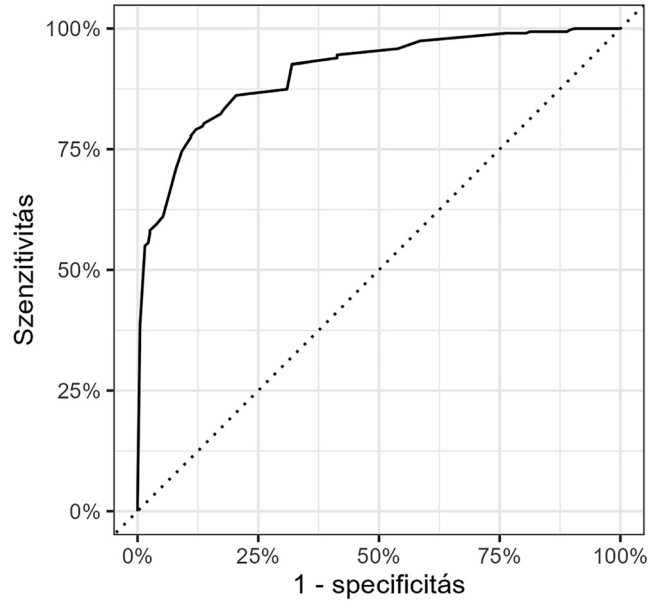
		Választás	
		Lépcső	Lift
Predikció	Lépcső	241	65
	Lift	70	523

3. táblázat. A generalizált lineáris kevert hatású regresszió fa modell tesztállományon mutatott predikciós teljesítménymutatói

Mutató	Érték
Pontosság (accuracy)	0,85
Szenzitivitás	0,78
Specificitás	0,89
Kiegyensúlyozott pontosság	0,83
ROC AUC	0,91

A GÉPI TANULÁS KORLÁTAI

A gépi tanulás használatának vannak korlátai is (Hullman, Kapoor, Nanayakkara, Gelman és Narayanan, 2022). Az egyik jelentős korlátozás a modell képzéséhez használt adatok minősége és reprezentativitása. A gépi tanulási modellek nagy mennyiségű adatra támaszkodnak a minták azonosításához és az előrejelzések készítéséhez, és ha az adatok torzak vagy hiányosak, a modell pontatlan vagy hiányos eredményeket adhat. Ez problémákhoz vezethet a kutatásban, ha a modellt olyan mintán tanítják be, amely nem reprezentatív a szélesebb populációra nézve, ami pontatlan előrejelzésekhez és ajánlásokhoz vezet.



5. ábra. A generalizált lineáris kevert hatású regresszió fa modell ROC görbéje
Megjegyzés: A szenzitivitás és a specificitás kapcsolatát mutatja be különböző döntési küszöbértékek esetén. A modell teljesítményének egyik mérőeszköze az ROC görbe alatti terület. A modell ROC AUC értéke a tesztadatokon: 0,91.

A gépi tanulás másik korlátja a pszichológiai kutatásban a „fekete doboz" problémája. A gépi tanulási modellek gyakran összetettek, és nehezen értelmezhetőek, így nehéz megérteni, hogy a modell hogyan készíti előrejelzéseit. Az átláthatóság hiánya bizalmatlansághoz vezethet az eredményekkel kapcsolatban, és korlátozhatja az előrejelzéseken alapuló, megalapozott döntések meghozatalának képességét. Emellett a modellek összetettsége megnehezítheti a predikciókat vezérlő változók azonosítását, ami problémás lehet, amikor a pszichológiai folyamatok mögöttes mechanizmusait próbáljuk megérteni Ennek a problémának a megoldására szolgál például a változók fontosságának becslése, amely alapján meg tudjuk állapítani, melyik változó járul hozzá leginkább a modell predikciójának pontosságához. Egy másik informatív mérőszámcsoport az ún. generalizált SHAP értékek (Bowen és Ungar, 2020). Ezek segítségével meghatározhatjuk, miért valószínűbb, hogy az adott megfigyelés inkább tartozik az egyik osztályba, mint egy másikba; hogy miért különböznek modellünk előrejelzései a megfigyelések csoportjai között; valamint hogy miért teljesít gyengén a modellünk egy adott mintán.

A gépi tanulási modellek inkább változók közötti együttjárásokon és asszociációkon alapulnak, mint oksági összefüggéseken, ami korlátozhatja hasznosságukat bizonyos pszichológiai kutatási kérdések megválaszolásában. A gépi tanulási modellek képesek mintákat azonosítani az adatokban, de nem tudnak ok-okozati összefüggéseket megállapítani a változók között. Ez azt jelenti, hogy bár a gépi tanulás hasznos lehet előrejelzések készítéséhez, nem biztos, hogy ez a legjobb megközelítés a pszichológiai folyamatok mögöttes mechanizmusainak megértéséhez.

Végül, vannak etikai megfontolások, amelyeket figyelembe kell venni, amikor a gépi tanulást a pszichológiai kutatásban használjuk (Lo Piano, 2020). Például nagy mennyiségű személyes

adat felhasználása esetén fennáll a magánélet védelmének megsértése és az adatbiztonság megsértésének kockázata. Emellett fennáll a meglévő előítéletek és diszkrimináció megerősítésének kockázata, ha a modell képzéséhez használt adatok elfogultak vagy hiányosak. Ez pontatlan előrejelzésekhez és ajánlásokhoz vezethet, amelyek negatív következményekkel járhatnak az egyénekre és közösségekre nézve.

Összefoglalva, bár a gépi tanulás értékes eszköz lehet a pszichológiai kutatásban, fontos felismerni a korlátait és az etikai megfontolásokat. A kutatóknak gondosan mérlegelniük kell a modell képzéséhez használt adatok minőségét és reprezentativitását, a modell átláthatóságát, valamint az ok-okozati összefüggések helyett a korrelációk és asszociációk használatának korlátait. E megfontolások figyelembevételével a kutatók felelősségteljes és hatékony módon használhatják a gépi tanulást, amely elősegíti a pszichológiai folyamatok megértését, miközben minimalizálja az egyének és közösségek sérülését.

A GÉPI TANULÁS HELYE A PSZICHOLÓGUSOK OKTATÁSÁBAN

A pszichológiát tanuló hallgatók statisztikai alapképzése jelenleg Magyarországon nem tartalmazza kimerítően a gépi tanulás témakörét; más európai és amerikai egyetemeken azonban már legalább választható formában elérhetőek kurzusok (pl. amerikai és brit egyetemekhez lásd Friedrich, Buday és Kerr, 2000; TARG Meta-Research Group, 2022). A feltáró kutatások és a gépi tanulás oktatásának hiánya több okra vezethető vissza. Egyrészt a pszichológiában – mint ahogy a bevezetőben arra kitértünk – arányaiban alacsony a feltáró kutatások mennyisége a megerősítő kutatásokhoz képest. Feltáró kutatásokkal így a hallgatók is kisebb valószínűséggel találkoznak a tanulmányaik során. Másrészt a gépi tanulás módszereinek ismerete számos olyan készséget igényel, amelyek oktatása általában nem része a pszichológushallgatók képzésének. Ezek a programozás – amit általában nem tanulnak –, valamint a matematikai statisztika, amit tanulnak ugyan, de csak korlátozottan. Harmadrészt, a hallgatók nagy része a pszichológiára humán tudományként tekint, és az egyetemi szak kiválasztásakor nem feltétlenül van tisztában a pszichológia intenzív kutatásmódszertani és statisztikai hátterével. Ennek megfelelően nem minden hallgató motivált a haladóbb kutatási és statisztikai módszerek elsajátítására (Tessler, 2013).

Azonban a feltáró kutatások módszertanát nem csak azért érdemes tanítani, hogy a hallgatók képesek legyenek ilyen kutatásokat önállóan lefolytatni, hanem azért is, hogy megértsék ezen kutatások lehetőségeit és korlátait (Ziegler és Garfield, 2018). A legnagyobb rugalmasságot a gépi tanulási módszerek használatában a rendelkezésre álló eszközök közül az elemző scriptek írása és a programozás adja. Több programnyelv is alkalmas lehet ezeknek az elemzéseknek az elkészítésére, de a leginkább felhasználóbarát infrastruktúra a Python (Van Rossum és Drake, 2009) és R (R Core Team, 2021) nyelveken áll rendelkezésre. Valószínűsíthető, hogy a jövőben a gépi tanulási módszerek egyre nagyobb része válik hozzáférhetővé programozást nem ismerő felhasználók számára is (Ferreira, Pilastri, Martins, Pires és Cortez, 2021). Így azok, akik most a technológiai készségek hiánya miatt szorulnak ki a gépi tanulási módszerek használatából, a jövőben hozzáférhetnek ezekhez. Például a regularizált regresszió és néhány egyéb gépi tanuló algoritmus már jelenleg is elérhető olyan felhasználóbarát kezelőfelülettel rendelkező programokban, mint például a JASP (pl. JASP Team, 2018), vagy a jamovi (The jamovi project, 2024). Ezen eszközök használatához szükséges, hogy az alapfogalmakat megértsék a felhasználók, ha nem is tudják a gépi tanulás lépéseit maguktól végrehajtani.

4. táblázat. Pszichológusok képzésében használható tematika és források

Téma	Források
Feltáró és igazoló kutatások jellemzői és funkciói	Weston, Ritchie, Rohrer és Przybylski (2019)
Gépi tanulás felhasználási területei a pszichológiában	Rosenbusch, Soldner, Evans és Zeelenberg (2021)
A gépi tanulás különböző fajtái és funkciói	Hastie és mtsai (2009)
Prediktív modellezés, modellértékelés	Bruce, Bruce és Gedeck (2020)
Bias-variance egyensúly a modellválasztásban, túlillesztés és alulillesztés	Yarkoni és Westfall (2017)
Túlillesztés elkerülése, gazdálkodás az adatokkal, test set, cross-validation	Kuhn és Silge (2021)
Legfontosabb gépi tanulási algoritmusok	Hindman (2015)
A gépi tanulási folyamat lépései	Kuhn és Silge (2021)
Feature-ök létrehozása és szelekciója	Kuhn és Johnson (2019)
A gépi tanulás korlátai	Hullman és mtsai (2022)
A gépi tanulás etikai kérdései	Lo Piano (2020)

A *4. táblázat*ban összegyűjtöttük azokat a fogalmakat, amelyeket egy pszichológusoknak szánt gépi tanulás kurzuson érdemes lehet érinteni. Az itt tárgyalt témákhoz nem feltétlenül szükséges, hogy a hallgatók programozást is tanuljanak.

KÖSZÖNETNYILVÁNÍTÁS

Nagy Tamást a Nemzeti Kutatási Fejlesztési és Innovációs Hivatal Kutatási Alapja (OTKA PD 131954) és az ELTE Eötvös Loránd Tudományegyetem Kiválósági Alapja támogatta. Szászi Barnabást az ELTE Egyetemi Kiválósági Alap pályázata támogatta.

IRODALOM

Bowen, D., & Ungar, L. (2020). *Generalized SHAP: Generating multiple types of explanations in machine learning*. http://arxiv.org/abs/2006.07155.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. Taylor & Francis. https://play.google.com/store/books/details?id=JwQx-WOmSyQC.

Broomhead, D. S., & Lowe, D. (1988). Radial basis functions, multi-variable functional interpolation and adaptive networks. *Royal Signals and Radar Establishment Malvern (United Kingdom)*, *25*(3), 1–8. https://apps.dtic.mil/sti/citations/tr/ADA196234.

Bruce, P., Bruce, A., & Gedeck, P. (2020). *Practical Statistics for Data Scientists, 2e: 50+ Essential Concepts Using R and Python* (2nd ed.). O′Reilly. https://www.amazon.co.uk/dp/149207294X?linkCode=gs2&tag=oreilly20-21.

Copeland, B. J. (2022). Artificial intelligence. In *Encyclopedia britannica*. https://www.britannica.com/technology/artificial-intelligence.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory/Professional Technical Group on Information Theory*, *13*(1), 21–27. https://doi.org/10.1109/TIT.1967.1053964.

Deng, L., & Yu, D. (2014). Deep learning: Methods and applications. *Foundations in Signal Processing, Communications and Networking*, *7*(3–4), 197–387. https://doi.org/10.1561/2000000039.

Ferreira, L., Pilastri, A., Martins, C. M., Pires, P. M., & Cortez, P. (2021). A comparison of AutoML tools for machine learning, deep learning and XGBoost. In *2021 International Joint Conference on Neural Networks (IJCNN)*. https://doi.org/10.1109/ijcnn52387.2021.9534091.

Fokkema, M., Smits, N., Zeileis, A., Hothorn, T., & Kelderman, H. (2018). Detecting treatment-subgroup interactions in clustered data with generalized linear mixed-effects model trees. *Behavior Research Methods*, *50*(5), 2016–2034. https://doi.org/10.3758/s13428-017-0971-x.

Friedrich, J., Buday, E., & Kerr, D. (2000). Statistical training in psychology: A national survey and commentary on undergraduate programs. *Teaching of Psychology*, *27*(4), 248–257. https://doi.org/10.1207/S15328023TOP2704_02.

Gigerenzer, G. (2004). Mindless statistics. *The Journal of Socio-Economics*, *33*(5), 587–606. https://doi.org/10.1016/j.socec.2004.09.033.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., … Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, *27*. https://proceedings.neurips.cc/paper_files/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html.

Guyon, I. (2003). An introduction to variable and feature selection. https://www.jmlr.org/papers/volume3/guyon03a/guyon03a.pdf?ref=driverlayer.com/web.

Hajdu, N., Szaszi, B., & Aczel, B. (2020). Extending the choice architecture toolbox: The choice context mapping. https://doi.org/10.31234/osf.io/cbrwt.

Harrell, F. E. (2001). *Regression modeling strategies*. New York: Springer. https://doi.org/10.1007/978-1-4757-3462-1.

Hastie, T., Friedman, J., & Tibshirani, R. (2009). *The elements of statistical learning* (2nd ed.). New York: Springer. https://doi.org/10.1007/978-0-387-21606-5.

Hindman, M. (2015). Building better models: Prediction, replication, and machine learning in the social sciences. *The Annals of the American Academy of Political and Social Science*, *659*(1), 48–62. https://doi.org/10.1177/0002716215570279.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics: A Journal of Statistics for the Physical, Chemical, and Engineering Sciences*, *12*(1), 55–67. https://doi.org/10.2307/1267351.

Hullman, J., Kapoor, S., Nanayakkara, P., Gelman, A., & Narayanan, A. (2022). The worst of both worlds: A comparative analysis of errors in learning from data in psychology and machine learning. *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, 335–348. https://doi.org/10.1145/3514094.3534196.

The jamovi project (2024). jamovi (Version 2.5) [Computer Software]. Retrieved from https://www.jamovi.org.

JASP Team (2018). *JASP* (Version 0.9). https://jasp-stats.org/.

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255–260. https://doi.org/10.1126/science.aaa8415.

Jović, A., Brkić, K., & Bogunović, N. (2015). A review of feature selection methods with applications. In *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1200–1205). https://doi.org/10.1109/MIPRO.2015.7160458.

Kuhn, M., & Johnson, K. (2019). *Feature engineering and selection: A practical approach for predictive models*. CRC Press.

Kuhn, M., & Silge, J. (2021). *Tidy modeling with R*. https://www.tmwr.org/.

Lang, K. M., & Little, T. D. (2018). Principled missing data treatments. *Prevention Science: The Official Journal of the Society for Prevention Research*, *19*(3), 284–294. https://doi.org/10.1007/s11121-016-0644-5.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, *1*(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541.

Lo Piano, S. (2020). Ethical principles in machine learning and artificial intelligence: Cases from the field and possible ways forward. *Humanities and Social Sciences Communications*, *7*(1), 1–7. https://doi.org/10.1057/s41599-020-0501-9.

Mars, M. (2022). From word embeddings to pre-trained language models: A state-of-the-art walkthrough. *NATO Advanced Science Institutes Series E: Applied Sciences*, *12*(17), 8805. https://doi.org/10.3390/app12178805.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, *5*(4), 115–133. https://doi.org/10.1007/BF02478259.

R Core Team (2021). *R: The R project for statistical computing*. https://www.r-project.org/.

Robinson, D. (2017). What's the difference between data science, machine learning, and artificial intelligence? *Variance Explained*. http://varianceexplained.org/r/ds-ml-ai/.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408. https://doi.org/10.1037/h0042519.

Rosenbusch, H., Soldner, F., Evans, A. M., & Zeelenberg, M. (2021). Supervised machine learning methods in psychology: A practical introduction with annotated R code. *Social and Personality Psychology Compass*, *15*(2). https://doi.org/10.1111/spc3.12579.

Sagiroglu, S., & Sinanc, D. (2013). Big data: A review. In *2013 International Conference on Collaboration Technologies and Systems (CTS)* (pp. 42–47). https://doi.org/10.1109/CTS.2013.6567202.

Santosa, F., & Symes, W. W. (1986). Linear inversion of band-limited reflection seismograms. *SIAM Journal on Scientific and Statistical Computing*, *7*(4), 1307–1330. https://doi.org/10.1137/0907087.

Stanton, J. M. (2001). Galton, Pearson, and the peas: A brief history of linear regression for statistics instructors. *Journal of Statistics Education: An International Journal on the Teaching and Learning of Statistics*, *9*(3). https://doi.org/10.1080/10691898.2001.11910537.

TARG Meta-Research Group (2022). Statistics education in undergraduate psychology: A survey of UK curricula. *Collabra: Psychology*, *8*(1). https://doi.org/10.1525/collabra.38037.

Tessler, J. (2013). On the importance of learning statistics for psychology students. *APSSC Undergraduate Update*. https://www.psychologicalscience.org/members/apssc/undergraduate_update/undergraduate-update-summer-2013/on-the-importance-of-learning-statistics-for-psychology-students.

Thiese, M. S., Arnold, Z. C., & Walker, S. D. (2015). The misuse and abuse of statistics in biomedical research. *Biochemia Medica: Casopis Hrvatskoga Drustva Medicinskih Biokemicara/HDMB*, *25*(1), 5–11. https://doi.org/10.11613/BM.2015.001.

Tikhonov, A. N., Goncharsky, A. V., Stepanov, V. V., & Yagola, A. G. (1995). *Numerical methods for the solution of Ill-posed problems*. Springer Netherlands. https://doi.org/10.1007/978-94-015-8480-7.

Van Rossum, G., & Drake, F. L. (2009). *Python 3 reference manual*. CreateSpace. https://dl.acm.org/doi/book/10.5555/1593511.

Vargha, A. (2022). *Személy-orientált többváltozós statisztika: Klasszifikációs módszerek*. Pólya Kiadó.

Vaswani, A., Shazeer, N., Parmar, N., & Uszkoreit, J. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*. https://papers.nips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Vélez, J. I. (2021). Machine learning based psychology: Advocating for A data-driven approach. *International Journal of Psychological Research*, *14*(1), 6–11. https://doi.org/10.21500/20112084.5365.

Werbos, P. J. (1994). *The roots of backpropagation: From ordered derivatives to neural networks and political forecasting (Adaptive and cognitive dynamic systems: Signal processing, learning, communications and control)* (1st ed.). Wiley-Interscience. https://www.amazon.com/Roots-Backpropagation-Derivatives-Forecasting-Communications/dp/0471598976.

Weston, S. J., Ritchie, S. J., Rohrer, J. M., & Przybylski, A. K. (2019). Recommendations for increasing the transparency of analysis of preexisting data sets. *Advances in Methods and Practices in Psychological Science*, *2*(3), 214–227. https://doi.org/10.1177/2515245919848684.

Yarkoni, T., & Westfall, J. (2017). Choosing prediction over explanation in psychology: Lessons from machine learning. *Perspectives on Psychological Science: A Journal of the Association for Psychological Science*, *12*(6), 1100–1122. https://doi.org/10.1177/1745691617693393.

Ziegler, L., & Garfield, J. (2018). Developing a statistical literacy assessment for the modern introductory statistics course. *Journal of Educational and Behavioral Statistics*, *17*(2), 161–178. https://doi.org/10.52041/serj.v17i2.164.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, *67*(2), 301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x.

Using supervised machine learning methods in psychological research

Nándor Hajdú, Barnabás Szászi, Balázs Aczél and Tamás Nagy

Supervised machine learning can be used in many areas of psychological research, enabling the analysis of more complex data. Our aim is to describe the types, operation and use of supervised machine learning in psychological research. We review the benefits of machine learning, as well as the concepts of overfitting, bias, and variance that help in model selection and ensure robustness of the results. We also briefly describe the most important supervised machine learning algorithms and describe the key steps in the preparation of variables and data. An example analysis is presented to illustrate how the choice between stairs and elevator of university students can be modelled using supervised machine learning. At the end of the paper, we discuss the limitations of machine learning and its place in the education of psychologists. We hope that the knowledge presented will help psychologists to use machine learning more effectively and creatively.

KEYWORDS

machine learning, statistics, exploratory data analysis, research methodology