Közgazdasági és Gazdaságinformatikai Doktori Iskola

TÉZISGYŰJTEMÉNY

Kovács László

Változószelekciós algoritmusok vizsgálata általánosított additív modellekben

Egy új, hibrid metaheurisztika elemzése

című Ph.D. értekezéséhez

Témavezetők:

Dr. Láng Blanka
egyetemi docens

Dr. Racskó Péter
tudományos tanácsadó

Budapest, 2021

Számítástudományi Tanszék

TÉZISGYŰJTEMÉNY

Kovács László

Változószelekciós algoritmusok vizsgálata általánosított additív modellekben

Egy új, hibrid metaheurisztika elemzése

című Ph.D. értekezéséhez

Témavezetők:

Dr. Láng Blanka
egyetemi docens

Dr. Racskó Péter
tudományos tanácsadó

Tartalomjegyzék

1. Kutatási előzmények és a téma indoklása

A felügyelt gépi tanulás során az elemző célja, hogy egy jól definiált eredményváltozóra minél nagyobb pontosságú becslést adjon bizonyos magyarázóváltozók értékének ismeretében. Napjainkban a feladat számtalan összetett algoritmus segítségével megoldható. Pl. mélytanuló neurális hálózatok, véletlen erdők, támaszvektor – gépek stb. Azonban egyre több szerző, pl. Molnar (2020) és Du et al. (2019) hívja fel a figyelmet arra, hogy a legpontosabb becslést szolgáltató modellekben a használt magyarázóváltozók hatásai az eredményváltozóra nehezen, vagy egyáltalán nem visszafejthetők. Viszont, bizonyos gyakorlati szituációkban a gépi tanulás legfontosabb eredménye nem feltétlenül a minél pontosabb becslés elkészítése, hanem az egyes magyarázóváltozók hatásának megállapítása. Például, egy banknak egyértelműen meg kell indokolnia, hogy mi alapján utasít el egy hitelkérelmet. Ilyen esetekben nem előre jelző, hanem magyarázó modellek építése az elemző célja.

Napjaink „big data" környezetében, amikor egy adott becslési feladathoz rengeteg potenciális magyarázóváltozó könnyen az elemző rendelkezésére áll, még egy egyszerű lineáris regressziós modell alkalmazása esetén is problémás lehet a magyarázóváltozók hatásainak megállapítása. Molnar (2020) és James et al. (2013) egyik javaslata a probléma áthidalására, és a különböző felügyelt tanulási modellek értelmezhetővé tételére a változószelekció.

Hall (1999) szerint a változószelekció legfontosabb alapelve, hogy a kiválasztott magyarázóváltozók szorosan korreláljanak a becsülendő eredményváltozóval, de egymáshoz képest legyenek függetlenek. Lineáris esetben az elv a káros multikollinearitás elkerülését jelenti. Ez az elv gyakorlatilag megegyezik az ökonometriában gyakran alkalmazott parszimónia elvével (Wooldridge, 2016).

Doktori értekezésemben egy új, hibrid genetikus-harmónia kereső metaheurisztikus algoritmus (továbbiakban HGHK algoritmus) viselkedését vizsgálom meg általánosított additív modellek változószelekciós feladatának megoldására. Az értekezésben megfogalmazott K1-K5. kutatási kérdések az algoritmus általánosított additív modellek körében történő alkalmazásának technikai lehetőségeit, az algoritmus által javasolt modellek minőségét, valamint az algoritmus várható futásidejének csökkentési lehetőségeit vizsgálják.

1.1. Kutatási előzmények

Az értekezésben bemutatott kutatás legfontosabb előzményeinek két társszerzős publikációm tekinthető: Láng – Kovács (2014) és Láng et al. (2017). Az idézett két tanulmányban

gazdaságinformatikus szakos BSc-hallgatóként működtem közre. A publikációk BSc-hallgatóként önállóan készített korábbi TDK dolgozat és szakdolgozat továbbfejlesztése.

Ebben a két tanulmányban társszerzőimmel bemutatjuk a HGHK algoritmust a változószelekciós feladat megoldására. Az algoritmus legjobb részhalmaz elvű szelekciót valósít meg. Legfontosabb újdonsága a szakirodalomban szereplő hasonló megoldásokhoz képest, hogy új korlátozó feltételek segítségével igyekszik a modellben szereplő magyarázóváltozók redundanciáját meggátolni. Lineáris esetben ez a modellben szereplő magyarázóváltozók korrelálatlanságát jelenti, ami a lineáris modellek gyakorlati alkalmazásának egyik legfontosabb feltevése. Az algoritmus fejlesztése során saját hozzájárulásom a harmónia kereső és genetikus algoritmus ötvözésének ötlete volt. Az ötlet mögött az a felismerés állt, hogy a változószelekciós feladatban szükség van az egyedek nagyobb fokú véletlenségét biztosító rekombinációs operátorokra, amiket a harmónia kereső algoritmus tud biztosítani, ám futásidő szempontjából szükséges a genetikus algoritmus párhuzamos egyedkezelésének megőrzése is. Az algoritmus optimalizált C# implementációját társszerzőimmel közösen végeztük. Az algoritmus első működő implementációja önálló munka eredménye, amely a gazdaságinformatikus BSc szakdolgozatom része. Az idézett tanulmányokban megmutatjuk, hogy az algoritmus alkalmazásával elérhető a modellbe válogatott magyarázóváltozók megfelelő mértékű tapasztalati korrelálatlansága, ami nagyban segíti a magyarázóváltozók marginális hatásának értelmezhetőségét. Hiszen a marginális hatás vizsgálata során feltesszük, hogy a vizsgált magyarázóváltozó változása esetén a modellben szereplő összes többi magyarázóváltozó értéke változatlan marad. Továbbá, a magyarázóváltozók közti korrelálatlanság sérülése hatással van a lineáris modellek paraméterbecslésének stabilitására is (Hastie et al., 2011).

Doktori értekezésem közvetlen előzményének tekinthető az aktuárius MSc szakdolgozatomban végzett saját kutatásom alapján készült Kovács (2019) önálló publikációm. Ebben a tanulmányban a HGHK algoritmust kiterjesztettem a klasszikus lineáris modellekről a Cox-féle arányos hazárd modellekben végzett változószelekcióra is. A kiterjesztett algoritmussal életbiztosítási szerződések lemorzsolódási görbéit befolyásoló magyarázóváltozók körét határozom meg. A lineáris modellek esetéhez hasonlóan az arányos hazárd modellek esetében is érvényesült a HGHK azon tulajdonsága, hogy a modellkeretben alkalmazott egyéb változószelekciós algoritmusokhoz képest hasonló becslési pontosságú megoldást szolgáltat, ám ezt korrelálatlan magyarázóváltozóhalmaz segítségével éri el. Ennek köszönhetően egy életbiztosítási szerződésállományban előforduló lemorzsolódási okok könnyebben feltárhatóak

a HGHK segítségével, mint más változószelekciós algoritmus által szolgáltatott modelleket felhasználva. A kutatás legfontosabb hozadéka doktori értekezésem szempontjából, hogy ekkor készítettem el a HGHK algoritmus R nyelvű implementációjának első változatát. Ez az implementáció a közvetlen kiindulása a jelen értekezéshez tartozó R nyelvű programkódoknak, melyek szintén teljes mértékben saját munka eredményei.

Jelen doktori értekezésben megvizsgálom, hogy a HGHK algoritmus milyen módon terjeszthető ki az általánosított additív modellek (Generalized Additive Model, GAM) körébe. A GAM-ok esetében a magyarázóváltozók függetlensége fontos feltétel, hiszen a modellben magyarázóváltozók additív módon kapcsolódnak egymáshoz. Ez azt jelenti, hogy a modell célváltozóra adott becslése az egyes magyarázóváltozókhoz tartozó nem-lineáris $\hat{f}_1, \hat{f}_2, \dots, \hat{f}_m$ függvények összegeként áll elő (Hastie – Tibshirani, 1990) Ezzel a HGHK algoritmus korlátja, ami arra törekszik, hogy a modellben csak független (vagy legalábbis a függetlenség mérésére alkalmazott metrika szerint nem redundáns) magyarázóváltozók szerepeljenek, továbbra is hasznos, ha az elemző célja egy GAM modellel inkább értelmező modell építése, és az eredményváltozóra nem-lineárisan ható legfontosabb tényezők azonosítása.

1.2. Kutatási kérdések

Az ismertetett kutatási előzmények alapján jelen értekezésben a K1-K5. kutatási kérdésekre keresem a választ a HGHK algoritmus GAM-ok körére kiterjesztett verziójával kapcsolatban.

K1. Megvalósítható-e a HGHK algoritmus kiterjesztése lineáris modellekről GAM keretbe anélkül, hogy az algoritmusban lineáris esetben alkalmazott döntési változókon és bináris egyedreprezentáción változtatni kellene?

K2. Képes-e valós adatbázisokon a HGHK algoritmus olyan modelleket azonosítani, amelyek kevesebb és nem redundáns magyarázóváltozó segítségével nagyságrendileg hasonló becslési pontosságot nyújtanak, mint az értekezés során vizsgált egyéb korszerű változószelekciós algoritmusok GAM keretben?

K3. A HGHK algoritmusban az új memória/populáció létrehozása során a véletlen vagy a kontrollált rekombinációs operátorokat érdemesebb előnyben részesíteni GAM keretben történő alkalmazás esetén?

K4. Milyen eszközzel érdemes a HGHK algoritmusban a kezdeti populáció jó minőségét biztosítani: több véletlen kezdőpopulációból lefuttatni a HGHK algoritmust kis populáció méret és alacsony maximális generációszám mellett, vagy inkább nagy

populáció méretet és magas maximális generációszámot célszerű alkalmazni és az algoritmust csak egyszer lefuttatni?

K5. Milyen párhuzamosítási stratégiával javítható jobban a HGHK algoritmus futásideje: több egyedhez tartozó modell egyszerre történő, párhuzamos kiszámításával, vagy egy modell kiszámításának párhuzamosításával? A kiválasztott párhuzamosítási stratégiával az algoritmus milyen hatékonyan skálázható? A párhuzamosítással elért empirikus gyorsítás a HGHK algoritmusban hogyan viszonyul a párhuzamosítás elviekben elérhető maximális gyorsításához?

2. A felhasznált módszerek

A K1-K5. kutatási kérdések megválaszolásához további kutatásomat a Design Science módszertan Hevner et al. (2004)-féle hét pontos ajánlása alapján tervezem meg.

A Design Science módszertan alkalmazása egyaránt jelenti informatikai eredménytermék kifejlesztését és kutatási kérdések megválaszolását. Az értekezésben a K1-K5. kutatási kérdéseinek megválaszolását egy saját algoritmus (HGHK) továbbfejlesztésével, implementációjával és annak numerikus hatékonyságvizsgálatával tervezem elérni, ami konkrét informatikai eredményterméknek tekinthető. A módszertan hét általános irányelve Hevner et al. (2004)-ben található.

Doktori értekezésemben elvégzett kutatásomban egy működő implementációját fejlesztem ki a HGHK algoritmusnak, ami képes a változószelekciós feladat megoldására GAM-ok esetén, valós adatbázisokon. Ezzel a 3. fejezetben bemutatott HGHK algoritmus megfelel a Design Science módszertan 1. ajánlásában szereplő termék kritériumainak.

Kutatásomban bemutatom a vizsgált GAM-ok elméleti és módszertani kereteit a legfrissebb szakirodalmi források alapján. Kiemelem, hogy a GAM-ok paraméterbecslési eljárásai érzékenyek a modellben szereplő magyarázóváltozók redundanciájára, a concurvity jelenségre. Bemutatom, hogy a concurvity jelenség a modellek eredményeinek üzleti értelmezhetőségét is megnehezíti. A GAM keretben alkalmazható legkorszerűbb változószelekciós algoritmusok működésének elemzésével rávilágítok arra, hogy a GAM változószelekció során alkalmazott legnépszerűbb algoritmusok a concurvity jelenséget nem, vagy csak korlátozott mértékben veszik figyelembe. Tehát, irodalomkutatás segítségével bemutatom, hogy az értekezésben szereplő kutatás eredményterméke fontos és releváns üzleti problémára nyújt megoldást a Design Science módszertan 2. ajánlásának megfelelően.

Részletes irodalomkutatással alátámasztom, hogy a HGHK algoritmus teljesítményének numerikus összehasonlítása az értekezésben megvizsgált, egyéb GAM keretben alkalmazható korszerű változószelekciós algoritmusok teljesítményével korszerű validációs eljárások és teljesítménymetrikák segítségével történik. Ezzel figyelembe véve a Design Science módszertan 3. ajánlását.

Kutatási eredményemként bemutatom, hogy az általam javasolt HGHK algoritmus maradéktalanul figyelembe veszi a concurvity jelenséget a változószelekció során. Továbbá, két eltérő paraméterekkel rendelkező gyakorlati problémán alkalmazom a HGHK-t és az algoritmus teljesítményét az egyéb vizsgált változószelekciós eljárásokkal szemben összehasonlítható módon visszaméretem. A numerikus kísérletek megerősítik, hogy a HGHK képes elérni, hogy az eredményül kapott modell concurvity jelenségtől mentes legyen, továbbá a becslési pontossága is nagyságrendileg megközelíti a többi vizsgált algoritmus pontosságát a legtöbb kiértékeléshez alkalmazott metrika alapján. A kutatás numerikus eredményei igazolják a HGHK elfogadható várható futásidejét is az algoritmus memóriájában/populációjában lévő egyedekhez tartozó GAM-ok párhuzamos kiszámítása esetén. Ezen numerikus eredmények alapján igazolom, hogy a kutatásom terméke, a HGHK algoritmus kutatási értéket hordoz, és ezzel kutatásom megfelel a Design Science 4. ajánlásának.

A HGHK algoritmus teljesítményének optimalizálását a paraméterek szigorú ceteris paribus elvű érzékenységvizsgálatával végzem el. Az eredmények függvényében a HGHK algoritmust nagyobb méretű feladatokon új módon, több, kisebb méretű kezdeti populációval alkalmazom a nagyobb méretű változószelekciós feladat hatékony megoldása érdekében. Ezen túl vizsgálom a HGHK skálázhatóságát több párhuzamosított végrehajtásra alkalmaz virtuális hardver architektúra alkalmazásával. A HGHK empirikusan mért teljesítményét mindig visszamérem a párhuzamosítással az adott architektúrán elérhető elvi maximális gyorsításhoz képest. Továbbá, az eredmények kiértékelése során szigorúan egységes validációs módszertant követek. Mindezekkel biztosítom a megfelelést a Design Science módszertan 5. ajánlásában szereplő kutatási szigornak és az összes elérhető algoritmusfejlesztési eszköz-környezet törvényszerűségei megfelelő alkalmazásának (6. ajánlás).

A kutatás aktív kommunikációja (Design Science 7. ajánlás) során doktori kutatásom eredményeit bemutattam több hazai és nemzetközi konferencián is, valamint referált tudományos folyóiratokban is megjelentek publikációim. A kutatási eredmények bemutatásra kerültek a Budapesti Corvinus Egyetem „Adatelemzés a gyakorlatban” előadássorozatának részeként és a 2020 januári Kutatási héten is, ahol az eredményeket az Egyetem hallgatóival és

olyan kutatókkal osztottam meg, akik a HGHK algoritmust saját kutatásaik során maguk is alkalmazni tudják.

Kutatási eredményeim reprodukálhatóságát szem előtt tartva az értekezés részét képezi az értekezésben vizsgált és továbbfejlesztett HGHK algoritmus forráskódja R nyelven, valamint a numerikus eredményeket előállító R szkript. Az R szkriptek, a vizsgált algoritmusok futtatásához használt, *Rda* formátumú tanító és teszt adatbázisok, továbbá a numerikus eredményeket részleteiben tartalmazó *csv* és *xlsx* formátumú táblázatok a https://github.com/KoLa992/Hybrid-algorithm-for-GAMs tárhelyről érhetők el. Minden R nyelven írt szkript az R 3.5.3-as verziójában került futtatásra, 64 bites Windows 10 operációs rendszeren. A teszteléshez használt hardver konfiguráció (a skálázhatóság vizsgálatán kívül) egy Intel Core i7-8750H 2,20 GHz processzorral (12 mag) és 8 GB 2666 MHz DDR4 RAM-mal rendelkező személyi számítógép.

3. Az értekezés eredményei

A 3.1. fejezetben ismertetem a HGHK algoritmus által megoldandó változószelekciós feladatot. A 3.2. fejezetben irodalomkutatásom eredményei alapján rávilágítok arra, hogy a HGHK algoritmus miben egyedi a korábbi GAM keretben működő változószelekciós algoritmusokhoz képest. A 3.3. fejezetben bemutatom a HGHK algoritmus működését és a hatékonyságát befolyásoló paraméterek körét. A fejezet eredményei alapján választ fogalmazok meg a K1 kutatási kérdésre. A 3.4. és 3.5. fejezetekben a HGHK algoritmus numerikus hatékonyságvizsgálatának és a szakirodalom által javasolt változószelekciós algoritmusokkal vett összevetésének eredményeit mutatom be. A fejezetek alapján a K2, K3 és K4 kutatási kérdésekre adok válaszokat. A 3.6. fejezetben a HGHK skálázhatóságát vizsgáló K5 kutatási kérdésre válaszolok, további numerikus kísérletek segítségével. Végül, a 3.7. fejezetben a kutatási kérdésekre adott válaszokat összefoglalom és ezek segítségével minősítem a HGHK algoritmus gyakorlati alkalmazhatóságát.

3.1. Változószelekciós feladat GAM keretben

Legyen $Y = [y_1, y_2, \dots, y_n]^T \in \mathbb{R}^n$ megfigyelt mintaelemek vektora egy exponenciális eloszláscsaládba tartozó eloszlású valószínűségi változóból. Egy GAM-ban Y várható értéke $X_j = \left[x_{j1}, x_{j2}, \dots, x_{jn}\right]^T$ megfigyelt magyarázóváltozók értékének ismeretében (1) módon becsülhető (Hastie – Tibshirani, 1990).

$$h\big(E(Y)\big) = \varepsilon + \sum_{j=1}^{p} f_j\big(X_j\big) \qquad (1)$$

Ahol $h(\cdot)$ az Y eloszlásához tartozó link függvény, $\varepsilon = [\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n]^T$ a modell hibavektora és $f_j(\cdot)$ a j-edik magyarázóváltozóhoz tartozó transzformációs függvény.

A doktori értékezésben az f_j függvényeket thin plate spline-ok segítségével reprezentálom, amelyek paramétereit büntetőtagos iteratív súlyozott legkisebb négyzetek módszerrel becsülöm meg (Wood, 2017). A reprezentáció előnye a szakirodalomban javasolt alternatívákhoz képest, hogy képes az f_j-k formájának automatikus kiválasztására egy büntetőtag segítségével, ami f_j második deriváltjától függ. A reprezentációban egyedül a spline bázisok maximális számának meghatározása szükséges, mint külső paraméter. Ezt a maximális számot k_j-vel jelölöm a j-edik magyarázóváltozóra. Az optimális választás k_j-re a legkisebb egész szám, ami elég nagy ahhoz, hogy a hozzá tartozó f_j thin plate spline függvény X_j varianciájának legnagyobb részét megőrizze. Szerencsére, Augustin et al. (2012) javasol egy statisztikai próbát, amelynek nullhipotézise, hogy $Var\left(f_j(X_j)\right)$ nem tér el szignifikánsan $Var(X_j)$-től. A próba segítségével a HGHK-ban automatizálni lehet k_j megválasztását. A módszert a 3.3. fejezetben részletezzük.

Legyen $X = \{X_1, X_2, \dots, X_m\}$ a lehetséges magyarázóváltozók halmaza egy GAM-ban. Változószelekció során a feladatunk egy olyan $\tilde{X} = \{X_1, X_2, \dots, X_p\} \subseteq X$ részhalmaz meghatározása, ami a legjobb általánosító képességű GAM-ot eredményezi. A GAM általánosító képessége azt mutatja meg, hogy a modell mekkora pontossággal tudja az n elemű mintán kívüli populációt is jellemezni a minta alapján kinyert információk alapján. A megfelelő általánosító képesség eléréséhez kompromisszumra van szükségünk a mintainformációk felhasználásának tekintetében. Ha túl kevés mintainformációt használunk fel, akkor nem nyerünk elég jó képet a valóságról. Ha túl sokat használunk fel, akkor túlságosan „ráfókuszálunk" a mintánkra, és ez rontja a mintán kívüli elemek leírásának pontosságát (Wooldridge, 2016).

Egy modell általánosító képességének mérésére több eszközt is kínál a szakirodalom. A doktori értekezésben elsősorban a McFadden-féle korrigált R-négyzet mutatót ($\bar{R}^2$) alkalmazom. A mutató részletes bemutatását McFadden (1974) tartalmazza. Ez alapján a változószelekciós feladat úgy is megfogalmazható, hogy egy olyan $\tilde{X}$ részhalmazt keresünk, amire $\bar{R}^2$ maximális. Ez egy legjobb részhalmaz kiválasztási probléma, ami NP-nehéz, hiszen az üres halmaz kizárásával is $2^m - 1$ db lehetséges megoldást kell vizsgálni (Huo – Ni, 2007).

Az $\tilde{X}$-be kiválasztott magyarázóváltozók redundanciájának elkerülése érdekében a változószelekciós feladatot érdemes egy extra korlátozó feltétellel bővíteni. A korlát

bevezetéséhez a multikollinearitás jelenségének nem-lineáris kiterjesztését, a concurvity-t szükséges mérni. Wood (2017) javaslata alapján a thin plate spline függvényeket alkalmazó GAM-ok esetében a concurvity egy $[0,1]$ intervallumon adott index segítségével mérhető. Az index 0 értéke a j-edik magyarázóváltozóra azt jelenti, hogy a változó varianciája egyáltalán nem leírható a többi modellben szereplő magyarázóváltozó nem-lineáris, additív kombinációja segítségével. Ellenben, az 1 érték azt jelenti, hogy a j-edik magyarázóváltozó tökéletesen reprodukálható a mintában a többi modellben szereplő magyarázóváltozó segítségével. Az index alapötlete az, hogy minden f_j függvényt fel lehet bontani olyan g_j spline bázisokra, amelyek más $f_{c\neq j}$ modellben szereplő transzformációs függvényeknek is részei. Amennyiben g_j az f_j varianciájának nagy hányadát teszi ki, akkor a j-edik magyarázóváltozót káros concurvity jelenség jellemzi. Ezek alapján a Wood-féle concurvity index $\left(\frac{\|g_j\|}{\|f_j\|}\right)^2$ módon adott. A képletben $\|\cdot\|$ az euklideszi normát jelöli. Az értekezésben Wood (2017) ajánlása alapján járunk el, amely szerint amennyiben az X_j-hez rendelt concurvity mérték 0,5-nél nagyobb, akkor a változó hatásának értelmezését károsan befolyásoló concurvity van a GAM modellben.

3.2. Változószelekciós algoritmusok GAM keretben

Irodalomkutatásom eredményei alapján a GAM keretben működő változószelekciós algoritmusok három nagyobb csoportra bonthatók.

Az első csoport a klasszikus stepwise logikát követi: egyszerre csak egy változót ad hozzá vagy vesz el a modellből, és ezt a műveletet addig folytatja, amíg a kiválasztott változószelekciós célfüggvény már nem javul szignifikánsan. A konkrét algoritmusok között a stepwise lépések finomságában találhatunk különbségeket. A klasszikus Stepwise algoritmus binárisan kezeli a változókat: egy változó vagy benne van $\tilde{X}$-ben vagy nem. Lásd pl. Venables – Ripley (2002) és Hastie et al. (2011). A GAMBoost algoritmus viszont bázis-spline függvények alkalmazásával egy lépésben mindig azon változó „súlyát” növeli a modellben, amivel a legjobb változást tudja elérni a kiválasztott változószelekciós célfüggvényben. Az algoritmus végén kapott modellben a 0 súllyal (minden bázisfüggvény együtthatója 0) szereplő változók azok, amiket szelektált a módszer. Tehát ez az algoritmus is a stepwise logikát követi, hiszen egy lépésben csak egy változón módosít a modellben, csak ezt nem bináris, hanem folytonos módon teszik meg. A részletek Binder – Tutz (2008) és Schmid – Hothorn (2008) tanulmányokban olvashatók. A GAMBoost algoritmus alapötletét a módosított backfitting eljárás (Belitz – Lang, 2008)

fejleszti tovább oly módon, hogy lehetővé teszi egy lépésben több változó „súlyának” növelését is a modellben, ezzel jól párhuzamosíthatóvá téve az algoritmust (Umaluf et al., 2015).

A második csoport a regularizációs módszerek családja. Ezek a módszerek a Tibshirani (1996)-féle Lasso algoritmus általánosításainak tekinthetők nem lineáris esetre. A regularizációs módszerek alapötlete, hogy a változószelekciót a modellek paramétereinek becslési eljárásába beépítik, mint extra korlátozó feltételeket. Ezen extra korlátok azt eredményezik, hogy bizonyos f_j-k már a becslés során azonosan 0-nak adódnak. A csoport képviselői: COSSO (Lin – Zhang, 2006), kétféle büntetőtagos thin plate spline (Marra – Wood, 2011), Nemnegatív Garrotte (Cantoni et al., 2011).

Megjegyzendő, hogy sem a stepwise, sem a regularizációs módszerek nem biztosítják közvetlenül a multikollinearitás, vagy nem-lineáris esetben a concurvity jelenség elkerülését a modellben. Sőt, több szerző kimondottan felhívja a figyelmet arra, hogy a stepwise és regularizációs modellek konzisztenciája sérül multikollinearitás, vagy nem-lineáris esetben concurvity jelenség fennállása esetén a lehetséges magyarázóváltozók körében: Chong – Jun (2005), Zhao – Yu (2006), Signoretto et al. (2008), Jia – Yu (2010). A konzisztencia hiánya GAM esetben azt jelenti, hogy a változószelekció során olyan változók kerülnek be $\tilde{X}$, amikhez a teljes populációra felírt regresszióban minden bázisfüggvény 0 együtthatóval szerepel, és olyan változók bázisfüggvényeinek lesz kivétel nélkül 0 súlya a végső modellben, amik a teljes populációra felírt modellben nem csupa 0 együtthatójú bázisfüggvényekkel szerepelnek.

A változószelekciós módszerek harmadik csoportját az eredményváltozó és az egyes magyarázóváltozók közös információtartalma alapján végez változószelekciót. Az ide tartozó algoritmusok olyan szempontból speciálisak, hogy nem feltételeznek GAM (és semmilyen más modell) keretet a változószelekció során. Ezek az algoritmusok definiálnak egy $I(x, y)$ mértéket x és y valószínűségi változók közös információtartalmának mérésére, majd különböző optimalizáló algoritmusok segítségével maximalizálnak egy egyedi változószelekciós célfüggvényt. Ezek a célfüggvények arra az elvre épülnek, hogy olyan X_j változókat válasszanak be a modellbe, melyekre $I(Y, X_j)$ magas, de a $k \neq j$ esetekre $I(X_k, X_j)$ mérték alacsony. Ez utóbbi szempont beemelésével ezek az algoritmus célfüggvényükben már védekeznek a modellbe válogatott változók közötti redundancia, tehát a multikollinearitás vagy a concurvity ellen. Ugyanakkor, az $I(x, y)$ jellegű mértékek alkalmazása a magyarázóváltozók közti kapcsolat szorosságának mértékét csak páronként vizsgálja. Az algoritmusok nem kezelik azt az esetet, amikor egy magyarázóváltozó több más magyarázóváltozó többváltozós

függvényeként áll elő. Emiatt az $I(x, y)$ mértéket alkalmazó algoritmusok segítségével nyert GAM-ok esetében továbbra is fennállhat a concurvity jelenség. A csoport legkorszerűbb képviselői az mRMR (Song et al., 2012) és HSIC-Lasso (Climente-González, et al., 2019).

A változószelekciós algoritmusok friss irodalmában elég gyakori a metaheurisztikus megoldások alkalmazása a feladat megoldására. Néhány példa a közelmúltból: Wang et al. (2015), Krömer – Platoš (2016), Mafarja – Mirjalili (2017), Sayed et al. (2019) és Abdel-Basset et al. (2020). Az idézett tanulmányok közös pontja, hogy a változószelekciós feladatban a modellt nem GLM-nek vagy GAM-nak tekintik, hanem valamilyen egyéb felügyelt gépi tanuló algoritmusnak (pl. k-NN, támaszvektor-gép, neurális hálózat stb.). Részben ebből adódóan az eredményváltozó szempontjából a változószelekciót csak osztályozó modellek körében értelmezik, mint a legjellemzőbb felügyelt tanulási feladatot. Az eredményül kapott modellekben a magyarázóváltozók eredményváltozóra gyakorolt hatásának visszafejthetősége nem szempont az idézett munkákban. Ebből adódik, hogy egyéb korlátok (pl. multikollinearitás vagy concurvity jelenség elkerülése) megfogalmazására nem kerül sor a változószelekció során.

Tudomásom szerint a javasolt HGHK algoritmus az egyetlen metaheurisztikus megoldás a változószelekciós feladatra, ami korlátozó feltételként a végső modellben szereplő magyarázóváltozók redundancia mentességét is biztosítja.

3.3. A HGHK algoritmus működése GAM keretben

Korábbi munkáimban (Láng et al., 2017) és (Kovács, 2019) egy hibrid genetikus – harmónia kereső algoritmust (HGHK algoritmus) javaslok a változószelekciós feladat megoldására lineáris modellek esetében. Az algoritmus a magyarázóváltozók VIF értékén keresztül a szelekciós folyamat során nem csak a változók közti páronkénti káros korrelációkra szűr. Ebben a fejezetben megadom az algoritmus kiterjesztését GAM keretre, thin plate spline-ok segítségével reprezentálva a magyarázóváltozók f_j transzformáló függvényeit. A HGHK algoritmus célfüggvénye a McFadden-féle $\bar{R}^2$ mutató. A genetikus algoritmus működését Goldberg (1989), míg a harmónia kereső algoritmusét Mahdavi et al. (2007) mutatja be részletesen. A HGHK folyamatábráját az értekezés 7. ábrája tartalmazza.

A HGHK algoritmus a változószelekciós feladat során a lehetséges megoldásokat egy m hosszú bitsorozat segítségével reprezentálja. Tehát csak arról dönt, hogy egy változót beemeljen-e a modellbe vagy sem. A HGHK algoritmus működésében megőrzésre kerül a genetikus algoritmus párhuzamosítható populáció (a harmónia kereső algoritmus terminológiájában memória) kezelése, ám az algoritmus keresztezés nevű rekombinációs operátorát lecserélem a

harmónia kereső algoritmus valószínűségi alapú rekombinációs operátorára. A cserére azért van szükség, mivel a genetikus algoritmus keresztezési operátora alapvetően olyan problémák esetén alkalmazható hatékonyan, ahol az egyedek minőségét érdemes részenként, bitcsoportonként javítani, és az egyes részmegoldások keresztezésével új megoldásokat létrehozni. Azonban, mivel az adatbázisok többségében a változók sorrendje véletlenszerű, így általában nincsenek csak az adott egyed bitsorozatának elején vagy végén kialakuló megfelelő mintázatok.

A korábbi memória átlagosnál jobb egyedeiből történő választás valószínűsége ($HMCR$, az angol harmony memory consideration rate kifejezésből) a futás során nő, míg a mutáció (módosítás) valószínűsége (bw) a futás során csökken. Ezzel a finomhangolással elérhető, hogy az algoritmus futásának elején agresszívebb lesz a populáció fennmaradó helyeire az egyedek teljesen véletlen generálása. Ahogy az algoritmus futása során egyre inkább közelebb kerül az optimumhoz, a keresési tér egyre kisebb részét kell bejárnunk az új egyedek generálása során, így nagyobb hangsúly tehető az egyedek öröklésére a korábbi generációkból. A HGHK algoritmus korai kilépési feltétele akkor érvényesül, ha az utolsó valahány lépésben a célfüggvény értéke nem változik a populációnk legjobb egyedének esetében.

A HGHK algoritmus akkor alkalmazható GAM keretben is, ha a megoldásokat továbbra is bitsorozat segítségével lehet reprezentálni. Amennyiben a változóra illesztett spline rendjét és a szakaszhatárokat is meg kell határozni, akkor már összetettebb reprezentáció szükséges a gének szintjén. Szerencsére, a 3.1. fejezetben ismertetett thin plate spline-ok alkalmazásával az egyedek bináris reprezentációja és a változószelekciós feladat keresési tere nem változik meg a lineáris esethez képest.

Az algoritmusban azt a technikát követem, hogy ha egy X_j magyarázóváltozót bekerül a GAM modellbe, akkor az alapértelmezés szerint $k_j = 10$. Amennyiben X_j értékkészlete kisebb, akkor k_j az X_j magyarázóváltozó lehetséges értékek számával lesz egyenlő. Amennyiben az Augustin et al. (2012)-féle próbák p-értéke $\alpha = 0{,}01$ alatti, akkor a k_j értékét 5-ösével növelem addig, amíg a változóhoz megfelelően nagy k_j-t nem választottam.

A concurvity jelenség szűrése miatt el kell érni, hogy a memória frissítése során csak olyan megoldásokat vigyen át az algoritmus az új memóriába, melyekre a 3.1. fejezetben meghatározott concurvity mérték 0,5 alatti. Ha az i-edik egyed teljesíti a korlátot minden változóra, akkor a C_i bináris változó 1 értéket vesz fel, egyébként 0-t. Továbbá, az is szempont, hogy olyan modelleket preferáljon, amikben csak olyan változók szerepelnek, amelyek spline

függvényében elutasítható azt a nullhipotézist, hogy $f_j \equiv 0$. A megadott nullhipotézis tesztelésére Marra – Wood (2011) ad meg egy χ^2-próbát, az algoritmusban ezt alkalmazom. Ha a felhasználó által megadott szignifikancia-szinten minden változóra elutasítható a nullhipotézis az i-edik egyedben, akkor S_i bináris változó 1 értéket vesz fel, egyébként 0-t.

Technikailag a korlátok beépítése úgy valósul meg, hogy amikor a memória frissítése során kiválogatom az átlagosnál jobb egyedeket az aktuális memóriából, akkor a memória átlagos célfüggvény értékére a (2) formulával adott súlyozott átlagot alkalmazom és csak olyan i egyed minősülhet átlagosnál jobb egyednek, amire $C_i = S_i = 1$.

$$\bar{R}_M^2 = \frac{\sum_{i=1}^{N} \bar{R}_i^2 \cdot C_i \cdot S_i}{\sum_{i=1}^{N} C_i \cdot S_i} \tag{2}$$

Ahol N a memória méretét, $\bar{R}_i^2$ az i-edik egyed célfüggvény értékét jelöli. Amennyiben $\sum_{i=1}^{N} C_i \cdot S_i = 0$, akkor $\bar{R}_M^2$ egyszerűen az $\bar{R}_i^2$ értékek számtani közepe és minden aktuális memóriában lévő egyed minősülhet átlagosnál jobb egyednek.

Az algoritmusba épített korlátok miatt könnyen előfordulhat, hogy a véletlenszerűen generált kezdeti memóriában nem lesz olyan egyed, amit a HGHK tovább enged az új memóriába a frissítés során és megfelel a $C_i = S_i = 1$ korlátoknak. Így könnyen lehet, hogy több generációig tart, míg talál olyan megoldásokat, amelyek minden korlátot kielégítenek. Tehát az algoritmus futásideje a kezdeti memória minőségétől függ. Emiatt érdemes lehet az algoritmust úgy futtatni, hogy egy futásidőben könnyen kezelhető memória méretet választok, és több véletlen kezdeti memóriából kiindulva lefuttatom az algoritmust egy alacsonyabb maximális generációszám mellett, ahogyan a K4 kutatási kérdés is ezt megfogalmazza. Ekkor a több futási eredmény legjobb célfüggvény értékkel rendelkező megoldását tekintem optimumnak. Alternatívaként a K4 kutatási kérdésben a nagy memóriaméret mellett történő egyszeri futtatás merülhet még fel a kezdeti memória rosszabb minőségének kezelésére.

A K1 kutatási kérdés kapcsán elmondható, hogy arra pozitív válasz adható a thin plate spline-ok jelen fejezetben bemutatott alkalmazási módjának segítségével. Hiszen, ezzel a GAM modellekben a spline függvények rendjének és a szakaszhatárok helyének megválasztása automatizálható. Így a lineáris esetben alkalmazott HGHK döntési változókon és bináris egyedreprezentáción nem szükséges változtatni.

3.4. A HGHK numerikus eredményei egy kis méretű feladaton

A HGHK algoritmus összevetése az egyéb GAM keretben működő változószelekciós algoritmusokkal két valós adatbázison történik. Az első adatbázis forrása Yeh (1998), és 1030 db betongerenda 9 ismérvét tartalmazza. Itt a feladat a gerendák nyomószilárdságának becslése a gerendák hét összetevője és a koruk függvényében. A változók részletes leírása szintén Yeh (1998)-ban szerepel. A változószelekciós feladat kicsi ($m = 8$), így a globális optimum könnyen megadható a lehetséges magyarázóváltozók összes részhalmazának legenerálásával. Az adatbázis használatának a célja, hogy megvizsgáljam, hogy az ismert globális optimumot milyen hatékonysággal azonosítják a vizsgált algoritmusok. A megtisztított adatbázist 70%-30% arányban osztottam fel tanító- és tesztmintákra.

A GAM keretben működő változószelekciós algoritmusok és a HGHK eredményeit a betongerendák adatbázison az értekezés 2. és 3. táblázatai tartalmazzák. Az algoritmusok sztochasztikus jellege miatt minden eljárást 30-szor futtattam az adatbázison, és minden esetben a legjobb modell eredményeit közlöm. Ezek alapján a K2 kutatási kérdés kapcsán kijelenthető, hogy a HGHK algoritmus modelljeinek tesztmintákon mért becslési pontossága nagyságrendileg nem marad el az értekezésben vizsgált egyéb algoritmusok által javasolt modellek teljesítményétől. A HGHK mellett egyedül az mRMR algoritmus képes concurvity jelenségtől teljesen mentes változóhalmaz azonosítására (2. táblázat), ám a javasolt modell becslési pontossága elmarad a HGHK algoritmus modelljétől. A betongerendák adatbázis esetében az eredmények robusztusnak tekinthetők a vizsgált teljesítménymetrikák körében. Az egyetlen másik concurvity korlátot nem sértő mRMR modellhez képest minden vizsgált metrikában (tehát nem csak R^2-ben) pontosabb becsléseket szolgáltat a HGHK által javasolt GAM (3. táblázat).

A betongerendák adatbázis kis mérete lehetőséget ad a HGHK algoritmus paramétereinek finomhangolásához. A HGHK-ban az értekezés 4. táblázatában szereplő paraméterek ceteris paribus optimalizálását végeztem el. A teljesen véletlen új egyed generálásra érdemes nagyobb súlyt helyezni, de nem érdemes teljesen elhagyni a memóriából történő kontrollált egyed generálást sem (5-ről 35%-ig emelhető a $HMCR$). Ezt támasztja alá az a tény is, hogy a memóriából történő új egyed előállítás esetén is érdemes az algoritmus első lépéseiben magas mutációs (bw) valószínűséget alkalmazni (90%), ám az induló érték nagyon alacsonyra (10%) csökkentése az iterációk során kifizetődő. A 4. táblázat alapján tehát a K3 kutatási kérdés kapcsán kijelenthető, hogy GAM keret esetén a HGHK algoritmusban a teljesen véletlen új

egyed generálásra érdemes nagyobb súlyt helyezni, de nem érdemes teljesen elhagyni a memóriából történő kontrollált egyed generálást sem.

További fontos tapasztalat, hogy az optimális paraméterek alkalmazása mellett a futtatások valamivel több, mint harmadában (12/30 esetben) úgy találja meg az algoritmus a globális optimumot vagy a második legjobb megoldást, hogy ez a megoldás már az algoritmus 5. iterációja előtt a memóriába került (20-as memória méret mellett). Azaz a keresési térnek kb. $\frac{5*20}{2^8} = 0{,}4$ részét vizsgálja csak át az algoritmus, mire megtalálja a két legjobb megoldás egyikét. A tapasztalat alapján választ tudunk adni a K4 kutatási kérdésre: a GAM keretből adódó rosszabb kezdeti memória (populáció) minőséget az optimális paraméterezésben az algoritmus több véletlen kezdeti memóriából történő futtatásával preferált kezelni az egyszer, ám nagy memóriamérettel történő futtatással szemben.

Az optimális paraméterek ceteris paribus keresésének részletes eredményei az értekezés mellékletében találhatók meg.

3.5. A HGHK numerikus eredményei nagy méretű feladat esetén

A második vizsgált adatbázisban a feladat annak megbecslése, hogy egy tajvani bank ügyfelei közül ki fog a lekérdezés időpontjától számított 1 hónapos időtartamon belül csődöt jelenteni hitelkártya adósságára. Az adatbázis 30 000 rekordot tartalmaz, 26 lehetséges magyarázóváltozóval. Az adatok forrása Yeh – Lien (2009), a változók leírása is ebben a tanulmányban található. A változószelekciós feladat jelen esetben az összes részhalmaz generálásával már nem oldható meg, az alkalmazott algoritmusok legjobb modelljeit csak önmagukban tudjuk vizsgálni, nincs referencia pont. A megtisztított adatbázist 70%-30% arányban osztottam fel tanító- és tesztmintákra. A vizsgálat során benchmarkként alkalmazom Yang – Zhang (2018) GLM és LightGBM modelljeinek eredményeit, valamint egy döntési fa és Recursive Feature Elimination változószelekcióval kombinált véletlen erdő algoritmust is.

A banki ügyfelek adatbázison futtatott benchmark és GAM változószelekciós algoritmusok részletes eredményei az értekezés 6. és 7. táblázataiban tekinthetők meg. Az algoritmusok sztochasztikus jellege miatt minden eljárást 20-szor futtattam az adatbázison, és minden esetben a legjobb modell eredményeit közlöm. A betongerendák adatbázison szerzett tapasztalatokat is hasznosítva az algoritmus futtatását több véletlenszerű kezdeti memóriából indítom, kisebb generációszám mellett. Az egyéb paraméterek megállapítása során a 4. táblázat optimális értékeit követtem. A paraméterezés részletezése az értekezés 10.4. fejezetében olvashatók. A banki ügyfelek adatbázison tapasztalt eredmények megerősítik a K2 kutatási kérdésre adott

korábbi választ, amely szerint a HGHK algoritmus modelljeinek tesztmintán mért becslési pontossága nagyságrendileg nem marad el az értekezésben vizsgált egyéb algoritmusok által javasolt modellek teljesítményétől. Sőt, a 6. táblázatban közölt eredmények alapján az is kijelenthető, hogy HGHK algoritmus modelljének banki ügyfelek adatbázison elért becslési pontossága AUC szerint a Yang – Zhang (2018)-féle változószelekció nélküli GLM modell teljesítményét meghaladja, és a szerzők által legpontosabbnak tartott LightGBM modell teljesítményétől sem marad el jelentősen. Továbbá, a HGHK algoritmus az egyetlen a vizsgált GAM algoritmusok közül a banki ügyfelek adatbázison, ami a concurvity jelenségtől teljesen mentes változóhalmazt javasol. Ez utóbbi eredményt még a concurvity jelenségre kontrolláló mRMR és HSIC-Lasso algoritmusok sem tudják biztosítani, mivel a jelenséget az algoritmusok csak magyarázóváltozó-páronként vizsgálják.

A banki ügyfelek adatbázison a teljesítménymetrikák függvényében változatosabb eredményeket tapasztalhatók, mint a 3.4. fejezet betongerendák adatbázisán. Itt, ha a két concurvity korlátot egyik változóban sem sértő modellt, a HGHK-t és a döntési fát vizsgáljuk, akkor elmondható, hogy becslési pontosság szempontjából kiegészítik egymást: a HGHK jobb „recall-jellegű" mutatókkal rendelkezik, míg a döntési fa a „precision-jellegű" metrikákban nyújt erős teljesítményt. Azonban, a banki csődkockázati elemzésekben a döntőshozóknak a „Recall-jellegű" metrikák a relevánsabbak (Moula, 2017), amiben a HGHK lényesen jobban teljesít, mint a döntési fa modell. Emiatt azt javasolható, hogy HGHK algoritmus modellje alapján érdemes azonosítani a nem redundáns hitelkockázati tényezőket.

A HGHK algoritmus várható futásideje a 6. táblázat alapján elfogadható a többi vizsgált algoritmuséhoz képest (az RFE változószelekcióval kombinált véletlen erdőjével összemérhető, míg a GAMBoost értékénél lényegesen gyorsabb), így megerősíthető a K4 kutatási kérdésre adott válasz is: érdemes a HGHK GAM keretben tapasztalt rossz minőségű kezdeti memóriáját (populációját) az algoritmus több véletlen kezdeti memóriából történő futtatásával kezelni kisebb generációszám mellett.

3.6. A HGHK skálázhatóságának vizsgálata

A HGHK algoritmus vertikális skálázhatóságát a memóriában lévő megoldásokhoz tartozó modellek szimultán kiszámítására allokált processzormagok számának függvényében vizsgálom. A vizsgálatot a banki ügyfelek adatbázison végzem el, hiszen az értekezés 6. táblázatának eredményei alapján a várható futásidő egyértelműen ebben, a nagyobb méretű adatbázisban meghatározó kérdés a HGHK algoritmus hatékonysága szempontjából.

A 3.4. és 3.5. fejezetekben elvégzett numerikus kísérletek mögötti hardverkonfiguráció az 1. fejezetben megadott személyi számítógép volt. A skálázhatóság numerikus vizsgálatához használt hardvererőforrások körét Microsoft Azure Data Science Virtual Machine (DSVM) segítségével bővítem. A platformon egyetemi környezetben elérhető hardverkonfigurációk segítségével 4, 8, 12 és 16 maggal rendelkező processzorok alkalmazhatók (Etaati, 2019).

A skálázhatóság numerikus vizsgálata előtt elengedhetetlen megállapítani a legfontosabb viszonyítási alapot, a párhuzamosítás által elérhető maximális gyorsítás mértékét, korlátlan számú elérhető processzormag esetén. Ezen felső határ megállapításához Amdahl-törvénye alkalmazható (Bryant et al., 2016). Amdahl-törvénye a teljes feladat/algoritmus párhuzamosítás miatti sebességnövekedésének felső határát határozza meg. A törvény pontos leírása Amdahl (1967)-ben található meg.

Az Azure DSVM környezetben végzett numerikus kísérletek során a HGHK algoritmust továbbra is 20-szor futtattam, és a várható futásidőt, valamint azok relatív szórását vizsgáltam processzormagok számának függvényében, majd megadtam a gyorsítás mértékét az egy magot alkalmazó architektúra esetéhez. Az így kapott numerikus gyorsítási eredményeket összevetettem az Amdahl-törvény alapján számolt maximális gyorsítás mértékével. Az eredményeket az értekezés 9. táblázata és 20. ábrája tartalmazza. Az Amdahl-törvénye szerint párhuzamosítással elérhető maximális gyorsítás mértékét a HGHK 4 mag esetén eléri, míg a többi esetben némileg elmarad tőle, aminek hátterében az algoritmus véletlen elemeit preferáló optimális paraméterezés következtében bekövetkező holtidő áll. Ez a holtidő magas relatív szóródást is okoz a 16 magos hardver konfiguráción tapasztalt futásidőkben.

A HGHK-ban tapasztalt holtidő viselkedésének megértéséhez 100 szimulált egyedhez tartozó GAM-ok kiszámítási idejeinek eloszlását érdemes tanulmányozni az értekezés 21. ábráján található hisztogram segítségével. Ez alapján a különböző változókombinációk esetén a szimulációban vizsgált 100 GAM számítási ideje enyhén jobbra elnyúló eloszlású. A szimuláció alapján továbbá azt mondhatjuk, hogy 1% körüli a tapasztalati valószínűsége annak, hogy egy jelentős mértékben kilógó futásidejű GAM-mal rendelkező változóhalmaz kerüljön be egy populáció egyedei közé. Ami azért tud problémákat okozni a párhuzamosítás során, mert azt a processzormagot, amin e GAM számítása fut, a többi processzormagnak „be kell várnia” mielőtt elkezdődik a szelekciós művelet a populációban. Mindez azért szükséges, hogy az átlagos célfüggvény érték (2) formulával megadható legyen a populációban. (2) viszont csak az összes egyed ismeretében számítható egy populációban.

A leghosszabb futásidővel rendelkező egyed bevárása miatt a HGHK minden iterációjában így lesz egy holtidő, amikor egy processzormag kivételével a többi nem dolgozik. Ha nincs nagyon kilógó számítási idővel bíró egyed több iteráció populációjában sem, akkor a holtidők elég rövidek tudnak lenni, így az algoritmus egészének futásidejében meg lehet közelíteni az Amdahl-törvény szerinti maximális gyorsítást. Azonban, ha sok populációban található nagy mértékben kiugró számítási idejű egyed, akkor az ezek miatt bekövetkező holtidő kumulálva a teljes algoritmuson végzett gyorsítást is messzire teheti az elméleti maximumtól. Mivel a HGHK optimális paraméterezésében az algoritmus véletlenszerű elemei dominálnak, így a kilógó számítási idejű egyedek jelenléte a populációkban könnyen magas ingadozást mutathat. Ezen ingadozás hatása azért a 16 magos esetekben jelenik meg dominánsan. A holtidő miatti futásidő veszteség az alkalmazott processzormagok függvényében nő, mivel a kihasználatlan processzormagok számát növelik az extra magok.

3.7. A kutatási kérdésekre adott válaszok összefoglalása

A 3. fejezet korábbi alfejezeteiben bemutatott irodalomkutatási eredmények, algoritmustervezés és numerikus hatékonyságvizsgálatok eredményei alapján a doktori értekezés K1-K5. kutatási kérdéseire tételesen is válaszolni lehet.

K1. Az első kutatási kérdés a HGHK algoritmus GAM keretre történő kiterjesztését vizsgálta a döntési változók és egyedreprezentáció szempontjából. Az 3.1. fejezetben bemutatott thin plate spline függvények segítségével a 3.3. fejezetben megmutattam, hogy GAM keretben a spline függvények rendjének és a szakaszhatárok helyének megválasztása automatizálható. Ezzel pedig a lineáris esetben alkalmazott döntési változókon és bináris egyedreprezentáción nem szükséges változtatni.

K2. A második kérdés a HGHK algoritmus által javasolt modellek minőségére koncentrált. A 3.4. és 3.5. fejezetben bemutatott két numerikus példa eredményei alapján elmondható, hogy a HGHK algoritmus modelljeinek tesztmintákon mért becslési pontossága nagyságrendileg nem marad el az értekezésben vizsgált egyéb algoritmusok által javasolt modellek teljesítményétől. Továbbá, a HGHK algoritmus az egyetlen a vizsgált GAM algoritmusok közül, ami a concurvity jelenségtől teljesen mentes változóhalmazt javasol mindkét numerikus példa esetében. Ez utóbbi eredményt még a concurvity jelenségre kontrolláló mRMR és HSIC-Lasso algoritmusok sem tudják biztosítani, mivel a jelenséget az algoritmusok csak magyarázóváltozó-páronként vizsgálják. A betongerendák adatbázison az mRMR algoritmus képes concurvity jelenségtől teljesen mentes változóhalmaz azonosítására

(értekezés 2. táblázat), ám a javasolt modell becslési pontossága elmarad a HGHK algoritmus modelljétől. A betongerendák adatbázis esetében az eredmények robusztusnak tekinthető a vizsgált teljesítménymetrikák körében (értekezés 3. táblázat). A banki ügyfelek adatbázison a teljesítménymetrikák függvényében változatosabb eredményeket tapasztaltunk. Ezen az adatbázison, ha a két, concurvity korlátot nem sértő modellt, a HGHK-t és a döntési fát vizsgáljuk, akkor elmondható, hogy becslési pontosság szempontjából kiegészítik egymást. Azonban, a banki csődkockázati elemzésekben a döntőshozóknak a „Recall-jellegű" metrikák a relevánsabbak, amiben a HGHK lényesen jobban teljesít, mint a döntési fa modell (értekezés 7. táblázat). Emiatt azt javasolhatjuk, hogy HGHK algoritmus modellje alapján érdemes azonosítani a nem redundáns hitelkockázati tényezőket.

K3. A harmadik kutatási kérdés a HGHK algoritmus rekombinációs operátorainak vizsgálatát célozza. A 3.4. fejezet eredményei alapján elmondható, hogy GAM keret esetén a HGHK algoritmusban a teljesen véletlen új egyed generálásra érdemes nagyobb súlyt helyezni, de nem érdemes teljesen elhagyni a memóriából történő kontrollált egyed generálást sem. Azaz, a rekombináció során a korábbi memória egyedei közül történő választásra kis súlyt érdemes helyezni, de teljesen elhagyni nem érdemes ezt az elemet. Sőt, az algoritmus futtatása során a korábbi memóriából történő választást érdemes növelni is!

K4. A negyedik kutatási kérdésben a HGHK algoritmus optimális memória (vagy populáció) méretét és kilépési feltételét vizsgálom. A 3.4. fejezetben, a kisebb méretű betongerendák adatbázison elvégzett ceteris paribus elvű érzékenységvizsgálat rávilágított arra, hogy az optimális paraméterek alkalmazása mellett a futtatások valamivel több, mint harmadában úgy találja meg az algoritmus a globális optimumot vagy a második legjobb megoldást, hogy a keresési térnek csak kb. 0,4 részét vizsgálja át. A tapasztalat így arra enged következtetni, hogy előnyösebb a kezdeti memóriák rosszabb minőségének kezelésére egy futásidőben könnyen kezelhető, kisebb memória méretet választani, és több véletlen kezdeti memóriából (azaz populációból) lefuttatni az algoritmust egy alacsonyabb maximális generációszám mellett.

K5. Az utolsó kutatási kérdésben a HGHK algoritmus párhuzamosítási stratégiáira helyeztük a fókuszt. A 3.5. és 3.6. fejezetek alapján a HGHK algoritmusban kifizetődik több egyedhez tartozó modellek egyszerre történő kiszámítása. Párhuzamosítás segítségével a HGHK algoritmus várható futásideje a nagyobb

méretű banki ügyfelek adatbázison a RFE változószelekcióval kombinált véletlen erdő algoritmus várható futásidejével nagyságrendileg összemérető. Továbbá, a HGHK jelentősen kedvezőbb várható futásidővel bír a GAMBoost értékénél (értekezés 6. táblázat). Az algoritmus párhuzamosításának skálázhatóságát 4, 8, 12 és 16 processzormaggal rendelkező virtuális architektúrákon végeztük el. Az empirikus vizsgálatok alapján az Amdahl-törvénye szerint párhuzamosítással elérhető maximális gyorsítás mértékét a HGHK 4 mag esetén eléri, míg a többi esetben némileg elmarad tőle, aminek hátterében az algoritmus véletlen elemeit preferáló optimális paraméterezés következtében bekövetkező holtidő áll. Ez a holtidő magas relatív szóródást is okoz a 16 magos hardver konfiguráción tapasztalt futásidőkben (értekezés 9. táblázat és 20. ábra).

Az értekezésben elvégzett szakirodalmi áttekintés és diszkusszió alapján az értekezésben bemutatott HGHK algoritmus az egyetlen metaheurisztikus megoldás a változószelekciós feladatra, ami GAM modellkeretben működik és korlátozó feltételként a végső modellben szereplő magyarázóváltozók redundancia mentességét többváltozós (és nem csupán páronkénti) összefüggéseket vizsgálva is biztosítja.

A HGHK algoritmus kapcsán GAM keretben az értekezés eredményei alapján elmondható, hogy ha a cél egy takarékos, magyarázó jellegű modell építése az eredményváltozóra és a futásidő sem komoly korlát, akkor a HGHK algoritmus alkalmazása javasolt az egyéb vizsgált változószelekciós algoritmusokkal szemben.

4. Főbb hivatkozások

Abdel-Basset, M., El-Shahat, D., El-henawy, I., de Albuquerque, V. H. C., & Mirjalili, S. (2020). A new fusion of grey wolf optimizer algorithm with a two-phase mutation for feature selection. Expert Systems with Applications, 139, 112824.

Amdahl, G. M. (1967, April). Validity of the single processor approach to achieving large scale computing capabilities. In Proceedings of the April 18-20, 1967, spring joint computer conference (pp. 483-485).

Augustin, N. H., Sauleau, E. A., & Wood, S. N. (2012). On quantile quantile plots for generalized linear models. Computational Statistics & Data Analysis, 56(8), 2404-2409.

Belitz, C., & Lang, S. (2008). Simultaneous selection of variables and smoothing parameters in structured additive regression models. Computational Statistics & Data Analysis, 53(1), 61-81.

Binder, H., & Tutz, G. (2008). A comparison of methods for the fitting of generalized additive models. Statistics and Computing, 18(1), 87-99.

Bryant, R. E., David Richard, O. H., & David Richard, O. H. (2016). Computer systems: a programmer's perspective. Third Edition. Upper Saddle River: Prentice Hall.

Cantoni, E., Flemming, J. M., & Ronchetti, E. (2011). Variable selection in additive models by non-negative garrote. Statistical modelling, 11(3), 237-252.

Chong, I. G., & Jun, C. H. (2005). Performance of some variable selection methods when multicollinearity is present. Chemometrics and intelligent laboratory systems, 78(1-2), 103-112.

Climente-González, H., Azencott, C. A., Kaski, S., & Yamada, M. (2019). Block HSIC Lasso: model-free biomarker detection for ultra-high dimensional data. Bioinformatics, 35(14), i427-i435.

Du, M., Liu, N., & Hu, X. (2019). Techniques for interpretable machine learning. Communications of the ACM, 63(1), 68-77.

Etaati, L. (2019). Data Science Virtual Machine and AI Frameworks. In Machine Learning with Microsoft Technologies (pp. 273-285). Apress, Berkeley, CA.

Goldberg, D. E., (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Boston, Kluwer Academic Publishers.

Hall, M. A. (1999). Correlation-based feature selection for machine learning. Doktori értekezés. University of Waikato.

Hastie, T. J., & Tibshirani, R. J. (1990) Generalized Additive Models. London: Chapman and Hall.

Hastie, T. J., Tibshirani, R., & Friedman, J. (2011). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). NYC, NY: Springer.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. Management Information Systems Quarterly, 28(1), 6.

Huo, X., & Ni, X. (2007). When do stepwise algorithms meet subset selection criteria?. The Annals of Statistics, 870-887.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning: with applications in R. New York :Springer.

Jia, J., & Yu, B. (2010). On Model Selection Consistency of the Elastic Net. Statistica Sinica, 20, 595-611.

Krömer, P., & Platoš, J. (2016, July). Genetic algorithm for entropy-based feature subset selection. In 2016 IEEE Congress on Evolutionary Computation (CEC) (pp. 4486-4493). IEEE.

Lin, Y., & Zhang, H. H. (2006). Component selection and smoothing in multivariate nonparametric regression. The Annals of Statistics, 34(5), 2272-2297.

Mahdavi, M., Fesanghary, M., & Damangir, E. (2007). An improved harmony search algorithm for solving optimization problems. Applied Mathematics and Computation, 188, 1567-1579.

Mafarja, M. M., & Mirjalili, S. (2017). Hybrid whale optimization algorithm with simulated annealing for feature selection. Neurocomputing, 260, 302-312.

Marra, G., & Wood, S. N. (2011). Practical variable selection for generalized additive models. Computational Statistics & Data Analysis, 55(7), 2372-2387.

McFadden, D. (1974). Conditional logit analysis of qualitative choice behaviour, in: P. Zarembka (ed.), Frontiers in Econometrics, Academic Press, New York, 105-142.

Molnar, C. (2020). Interpretable machine learning. Leanpub.

Moula, F. E., Guotai, C., & Abedin, M. Z. (2017). Credit default prediction modeling: an application of support vector machine. Risk Management, 19(2), 158-187.

Sayed, G. I., Tharwat, A., & Hassanien, A. E. (2019). Chaotic dragonfly algorithm: an improved metaheuristic algorithm for feature selection. Applied Intelligence, 49(1), 188-205.

Schmid, M., & Hothorn, T. (2008). Boosting additive models using component-wise P-splines. Computational Statistics & Data Analysis, 53(2), 298-311.

Signoretto, M., Pelckmans, K., & Suykens, J. A. (2008). Functional ANOVA Models: Convex-concave approach and concurvity analysis (No. 08-203). Internal Report.

Song, L., Smola, A., Gretton, A., Bedo, J., & Borgwardt, K. (2012). Feature selection via dependence maximization. Journal of Machine Learning Research, 13(5), 1393-1434.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1), 267-288.

Umlauf, N., Adler, D., Kneib, T., Lang, S., & Zeileis, A. (2015). Structured Additive Regression Models: An R Interface to BayesX. Journal of Statistical Software, 63(21), 1-46.

Venables, W. N. & Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth Edition. New York :Springer.

Wang, Y., Liu, Y., Feng, L., & Zhu, X. (2015). Novel feature selection method based on harmony search for email classification. Knowledge-Based Systems, 73, 311-323.

Wood, S. N. (2017) Generalized Additive Models: An Introduction with R (2nd edition). Chapman and Hall/CRC.

Wooldridge, J. M. (2016). Introductory econometrics: A modern approach. Nelson Education.

Yang, S., & Zhang, H. (2018). Comparison of several data mining methods in credit card default prediction. Intelligent Information Management, 10(05), 115.

Yeh, I. C. (1998). Modeling of strength of high-performance concrete using artificial neural networks. Cement and Concrete research, 28(12), 1797-1808.

Yeh, I. C., & Lien, C. H. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. Expert Systems with Applications, 36(2), 2473-2480.

Zhao, P., & Yu, B. (2006). On model selection consistency of Lasso. The Journal of Machine Learning Research, 7, 2541-2563.

5. A témakörrel kapcsolatos saját publikációk jegyzéke

Referált szakmai folyóiratok

Láng, B. & Kovács, L. (2014). Linear Regression Model Selection using Improved Harmony Search Algorithm. *SEFBIS Journal*, 9(1), pp. 15-22.

Láng, B., Kovács, L. & Mohácsi, L. (2017). Linear Regression Model Selection using a Hybrid Genetic - Improved Harmony Search Parallelized Algorithm. *SEFBIS Journal,* 11(1), pp. 2-9.

Kovács, L. (2019). Applications of Metaheuristics in Insurance. *Society and Economy,* 41(3), pp. 371-395.

Kovács, L. (2021). Változószelekció általánosított additív modellben metaheurisztika segítségével. *SZIGMA Matematikai-közgazdasági folyóirat*. (Megjelenés folyamatban)

Lektorált konferenciakötet

Kovács, L. (2021). Performance Testing of Feature Selection Algorithms for Generalized Additive Models. Proceedings of the 16th International Symposium on Operational Research: SOR '21. (Megjelenés folyamatban)