

Fogyasztásbecslés neurális hálókkal

Bevezetés

Manapság egyre elterjedtebb a személygépkocsik használata, mivel biztosítják a gyors és kényelmes közlekedést. Azonban ennek több hátránya is van gazdaságosság és környezettudatosság szempontjából. Az üzemanyag ára általában nő, ezért egyre drágábbá válik az autótulajdonosok számára a közlekedés. Emellett a XXI. század egyik nagy problémájáért, a klímaváltozásért is részben felelősek az autósok. Kutatások szerint ugyanis „*a közlekedés felelős a teljes uniós szén-dioxid-kibocsátás majdnem 30 százalékáért, amelynek 72 százaléka a közúti közlekedésből származik*”¹. Viszont ezeknek a problémáknak a mértéke csökkenthető lenne, ha tudnánk csökkenteni a járművek fogyasztását. Ezáltal kevesebb üzemanyagra lenne szükség, így kevesebb pénzből és károsanyag-kibocsátásból jöhetne létre egy utazás.

Napjainkban léteznek elektromos autók, amelyek nem bocsátanak ki káros anyagokat, és jelenleg ingyenes a töltésük. Azonban ezek jóval drágábbak a benzines vagy dieseles változatoknál, így csak nagyon kevés ember engedheti meg magának a megvásárlásukat. Továbbá azzal is számolnunk kell, hogy az autók működéséhez szükséges áram előállítása nem feltétlenül környezetbarát. Például Németországban végeztek egy kutatást, ami alapján „*az elektromos autók 11-28 százalékkal nagyobb mértékben terhelik a környezetet, mint a dízelautók, ha figyelembe vesszük az akkumulátorok gyártásából eredő szén-dioxid-kibocsátást és a Németországban felhasznált elektromos energiában a nem megújuló energia alapú módszerek létét*”². Ezért a fogyasztáscsökkentés az elektromos autók esetében is hasznos lenne.

Célkitűzés:

Kutatásunk során fő célunk az, hogy mesterséges intelligencia segítségével létrehozzunk egy programot, amely előrejelzi, hogy egy adott utazás milyen paraméterekkel valósítható meg a legkevesebb fogyasztás mellett. Hosszabb utazások tervezésénél ez hasznos lehet, mivel ezekre a paraméterekre (forgalom, hőmérséklet, átlagsebesség stb.) mi is hatással lehetünk. A paraméterek sokfélesége és összetettsége miatt egy neurális hálóra van szükségünk, amely képes komplex összefüggések modellezésére.

1. Adatok gyűjtése

Ahhoz, hogy a neurális hálónk minél pontosabban működjön, sok adatra van szükségünk, így első célunk minél több adat összegyűjtése. Ehhez létrehoztunk egy honlapot³, amelyen az önkéntesek meg tudják adni korábbi utazásaik adatait, amelyek egy adatbázisban kerülnek mentésre. Ehhez PHP és HTML programnyelveket használtunk. Így első célunk minél több adat összegyűjtése.

¹ <https://www.europarl.europa.eu/news/hu/headlines/society/20190313STO31218/amit-erdemes-tudni-a-gepjarmuvek-szen-dioxid-kibocsatasarol-az-eu-ban>

² <https://www.origo.hu/tudomany/20190418-nemet-kutatok-szerint-az-elektromos-autok-karosabbak-a-kornyezetre-mint-a-dizel-autok.html>

³ <http://biointelligence.hu/blanka/fogyasztas.1.html>

2. Adatok rendszerezése

A megfelelő adatmennyiség összegyűjtése után az adatok szűrése következik. Az adatok közül kiválogattunk egy meghatározott útvonalon történő utazásokat (Budapest és Dunaújváros között), majd kiválasztottuk azokat az adatokat, amelyeket bementként szeretnénk használni. Ezek az utazás kezdetkori hőmérséklet, az utazás befejezésénél a hőmérséklet, a légkondi állapota (bekapcsolva/kikapcsolva), a forgalom erőssége (alacsony, közepes, magas, illetve dugó), az átlagsebesség és az üzemanyag típusa (ebből 2 fordult elő: Shell Fuel Save 95 és Shell V-Power 95). Ezeket a bemeneti adatokat úgy konvertáltuk, hogy mindegyik szám legyen, amelynek nagy része -1 és 1 közé esik, ezáltal könnyebben kezelhetők a neurális háló számára. Kétféle módszert használtunk. Az elsőt az alapvetően számértékeken (kezdeti hőmérséklet, végső hőmérséklet, átlagsebesség) alkalmaztuk: az eredeti értékből kivontuk az összes megegyező típusú adat átlagát, majd elosztottuk a maximális és minimális érték különbségével. Tehát például ha a kezdeti hőmérsékletünk 18, akkor abból kivontuk az adatbázisunkban meglevő kezdeti hőmérsékletek átlagát (ami 20,11 volt), majd elosztottuk a legnagyobb és a legkisebb kezdeti hőmérséklet különbségével (esetünkben $33-7=26$ -tal). Így végeredményként -0,081154-et kaptunk. Az alapvetően nem számértékeket (forgalom, légkondi, üzemanyag) úgy módosítottuk, hogy különböző számértékeket rendeltünk az értékekhez. A forgalomnál az alacsony értéke 0, a közepesé 0,5, a magasé és a dugóé 1. A légkondi bekapcsoltságát 0 és 1 értékekre állítottuk, ahogy a kétféle üzemanyagot is. Ezután két csoportra osztottuk az adatokat: tanítóadatokra (összesen 100 adatsor) és tesztadatokra (összesen 52 adatsor).

3. A neurális háló létrehozása

A tanítóadatok segítségével létrehoztunk különböző neurális hálókat. Ehhez Python Numpy és Scikit-learn szoftverkönyvtárakat használtunk. A pontosságukat úgy teszteltük, hogy betanítottuk őket, majd kiszámoltuk a valós fogyasztás és a tesztadatokra becsült értékek különbségét, majd vettük ezek számtani közepét. Ezt megtettük 100-szor, úgy, hogy minden betanításnál növeltük a kezdeti (iniciális) súlyokat 1-gyel (0-tól 99-ig). Végül pedig átlagoltuk az így kapott 100 db számtani közepet.

A neurális hálóknak sok paraméterük van, amelyek változtatásával nagy eltérések keletkezhetnek az eredményekben. Ebben a fázisban célunk az volt, hogy kísérletezzünk ezekkel a paraméterekkel, és megtaláljuk azokat a hálókat, amelyek becslése megfelelő a kutatáshoz. Vizsgálódásunk során az alábbiakat észleltük.

Neuronok és rétegek

A neurális hálónál a legmeghatározóbb változásokat a megadott neuronok és rétegek száma adta. Az alapbeállítások, amelyekkel ezeket a beállításokat teszteltük, az alábbiak voltak: `activation="relu"`, `solver="adam"`, `max_iter=200000`, `learning_rate_init=0.001`, `alpha=1e-3`, `tol=1e-4`, `random_state=i` (i 0-tól 99-ig változott minden lefuttatásnál). Az egy rétegből és neuronból álló háló átlagosan 0,577 l/100 km-es eltéréssel jósolt, úgy, hogy mindig egy konstans értéket vett fel. Így ennél mindenképpen jobb hálót akartunk keresni.

Így elkezdtek növelni a neuronok számát, és nagyjából 100 neuronig javultak az eredmények, aztán romlani kezdtek (1. ábra). A legjobb eredményt 100 neuronnal értük el, ott az átlagos eltérés 0,469 l/100 km volt.

2 réteggel nem sikerült jobb eredményt elérnünk az 1x100-as hálónál, itt a legjobb eredmény a 2x10-es háló 0,495 1/100 km-es átlagos hibával történő becslése volt. A 2x100-as háló 0,708-as átlagos eltéréssel dolgozott, ami nagyobb, mint az 1x1-es hálóé, itt valószínűleg a túltanulás jelensége jött létre (2. ábra).

A 3 réteges kísérleteknél a legjobb eredmény 0,471 1/100km lett a 3x8-as hálóval, ami már megközelítette az 1x100-as háló eredményét (3. ábra).

4 rétegnél sikerült találnunk 2 olyan hálót, amelyek jobban becsülnek az 1x100-as hálónál. A 4x7-es, ami 0,464, és a 4x6-os, amely 0,464 átlagos eltéréssel becsül (4. ábra).

A későbbiekben ezeknél nem találtunk jobb hálókat, 5 rétegnél az 5x5-ös háló érte el a legjobb eredményeket 0,466-os átlagos eltéréssel (5. ábra), azonban ahogy növeltük a rétegek, az eredmények egyre rosszabbak lettek (6. ábra), így kiválasztottuk azokat a hálókat, amelyek a legjobban becsültek (7. ábra, a hibasávok a szórást jelölik), majd tovább kísérleteztünk a paraméterekkel.

Aktiválási függvény

4 féle aktiválási függvényt lehet beállítani ebben a programban: identity ($f(x) = x$), logistic ($f(x) = \frac{1}{1+e^{-x}}$), tanh ($f(x) = \tanh(x)$) és relu, az alapbeállítás ($f(x) = \max(0, x)$).

A kiválasztott legjobb 4 hálóval kísérleteztünk, és az összes esetben az identity adta ki a legjobb eredményeket, a logistic az 1x100-as kivételével pedig a legrosszabbakat (8. ábra). A hálókat a tanh függvény is javította, de nem olyan mértékben, mint az identity.

Solver

Ezután a solverekkel kezdtem kísérletezni, amik a súlyok beállításában vesznek részt. Összesen 3 féle van: lbfgs, sgd és adam (az alapbeállítás). 2 neurális hálón vizsgáltam meg ezeket, különböző aktivációs függvényekkel.

Az 1x100-as hálónál az esetek nagy részében romlottak az eredmények az adam-hez képest (különösen a relu-lbfgs és a tanh-lbfgs párosításban), de egy alkalommal javulást figyeltünk meg az identity-lbfgs párosításnál. Itt az átlagos eltérések átlaga 0,381 volt (9. ábra).

A 4x7-es hálónál már az sgd solvernél 4-ből 2-szer javultak az eredmények az adamhez képest, azonban az igazi áttörés itt is az identity-lbfgs párosításnál történt: szintén 0,381 volt az átlagos eltérés (10. ábra).

Azt tapasztaltuk, hogy (a miénkhez hasonló adatmennyiséggel) relu függvényhez adam solvert, identityhez lbfgs, és logistichoz és tanh-hoz adam vagy sgd solvert érdemes használni.

Ezután elkezdtünk kísérletezni az identity-lbfgs beállítással, és attól függetlenül, hogy hány réteget és neuront használtunk, az átlagos eltérés mindig 0,381 1/100 km lett, csupán a százazredekben voltak eltérések. Így valójában 1 neuron is elég ahhoz az eredményhez, amelyet 100 vagy akár 1000 neuron nyújt ezekkel a beállításokkal, egyedül a szórásokban van eltérés, de az szintén százazred nagyságrendű, így elhanyagolható. Ezért, mivel kevesebb súlyt kell beállítani, és mivel az lbfgs solver eredetileg is gyorsabb a többinél, sokkal rövidebb időre van szükségünk a hálók betanításához, mint más beállításokkal, ami szintén előnyös.

Tanulási ráta és iterációk maximuma

A tanulási rátát adam és sgd solvereknél használjuk, és meghatározza, hogy milyen léptékben halad a függvény a minimum felé. Ez összefügg az iterációk maximumával, mert például kis tanulási rátához több iteráció kell, és fordítva. 2 neurális hálón kísérleteztem az alapbeállításokkal, és az alábbi eredmények jöttek ki.

A 4x7-es hálónál nem javultak az eredmények (11. ábra), és jól látszódik, hogy az 1 nagyon nagy tanulási rátának. Az 1x100-as hálónál szintén ezt tapasztaltuk (12. ábra). A 0,0001-es tanulási rátánál kipróbáltuk azt is, hogy 500000-re növeltük az iterációk maximumát, de az eredmény hasonló volt a kisebb iterációmaximumos értékhez. Emellett kísérleteztünk kisebb mértékben változtatni a tanulási rátát a 0,001-es nagyságrendben, de az alapbeállítás bizonyult a legjobbnak.

Alpha

Az alpha feladata a túltanulás („overfitting”) megakadályozása. Minél nagyobb az alpha, annál egyszerűbb függvény jön létre. Először egy olyan hálón kísérleteztünk, amely esetén túltanulás történt, ez pedig a 2x100-as háló volt, ami alapbeállításokkal 0,708-as átlagos eltéréssel becsült.

Elkezdünk növelni az alphát, hogy megakadályozzuk a túltanulást, és valóban jobb eredményeket kaptunk (13. ábra). 1 és 5 között részletesebben megvizsgáltuk (14. ábra), és 3-as alphával sikerült elérnünk 0,385-ös átlagos eltérést, így 54%-ára csökkentettük a háló becslési hibáját.

Azonban kísérleteink során rájöttünk, hogy az alpha növelésével nem csak a „túltanított” hálókön lehet javítani. Az alpha változtatásával kísérleteztünk a 4x7-es alapbeállításokkal működő hálóval, és ahogy nőtt az alpha, úgy az eredményeken is javultak (15. ábra). 1 és 8 között részletesebben megvizsgáltuk, és kirajzolódott, hogy az eltérések 3 és 3,5-ös alphánál a legkisebbek (16. ábra). Az 1x100-as hálóval is hasonló tapasztalataink voltak, ott a legkisebb eltérések 3,5 és 4-es alphánál voltak (17. ábra).

Több hálóval is kísérleteztünk, és a kutatásunk alatt a legjobb eredményeket az 1x1000-es hálóval értük el (18. ábra). Bár alapbeállításokkal 0,51 l/100 km-es átlagos eltéréssel becsült, az alpha növelésével ez jóval kisebb lett. A legkisebb eltérést 2 és 2,5-ös alphával értük el, ami 0,375 l / 100 km volt. A szórása is minimális, csupán 0,001.

4. Fogyasztásbecslő honlap létrehozása

Projektünk során fontosnak véltük, hogy ne csupán létrehozzunk egy neurális hálót, de azt bárki számára elérhetővé és használhatóvá tegyük. Így, miután megtaláltuk a legmegfelelőbb hálót, aminek az 1x1000-es bizonyult, szeretnénk volna létrehozni egy weboldalt. Azonban a neurális háló Pythonban íródott, így Pythonnal generáltunk egy JavaScript kódot, ami ugyanazokat a számításokat végzi el, mint a neurális hálónk. Ezután HTML, és a generált JavaScript kód használatával létrehoztunk egy honlapot⁴, ahol bárki beírhatja utazásának várható paramétereit, és a neurális háló elvégzi a fogyasztásbecslést. Ez pedig könnyen elvégezhető, mivel manapság már nagyon pontos előrejelzéseket tudunk kapni az időjárásról és a forgalomról egyaránt. Így, ha a lehető legkevesebb fogyasztást szeretnénk elérni, csak be kell írunk a lehetséges útvonalakat (amelyeket pl. Google Maps segítségével könnyen megkereshetünk), és azt

⁴ <http://biointelligence.hu/blanka/fogyasztasbecsles.html>

választani, amelyhez a háló a legkevesebb fogyasztást becsüli. Emellett, ha már van egy meghatározott útvonalunk, a várható értékek alapján megbecsülhetjük a fogyasztást, és így ki tudjuk számolni, mennyibe fog kerülni az utazás.

5. Jövőbeni tervek

Terveink között szerepel a jövőben egy GPS-hez hasonló mobilapplikáció létrehozása (19. ábra), amely a kiindulópont és a cél bevitelével automatikusan elvégzi a várható adatok lekérdezését, és ha szükséges, útközben is módosítja azokat. Így még egyszerűbbé és kényelmesebbé válna a fogyasztásbecslés a felhasználók számára. Emellett ezeknek az utazásoknak az adatait eltárolnánk, és megkérdeznénk a felhasználókat, hogy adják meg a valós fogyasztást. Ezáltal még több adatunk lesz, és tovább fejleszthetjük a hálónkat, hogy az még pontosabb adatokat adjon meg.

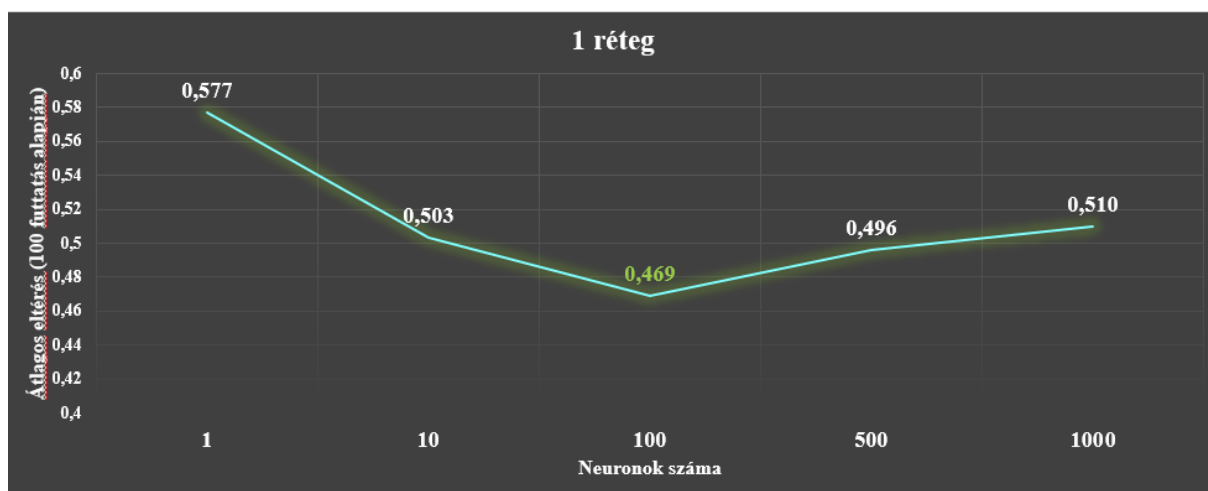
Összefoglalás

Ebben a pontban összegezni szeretném a kutatásunk során elért főbb eredményeket.

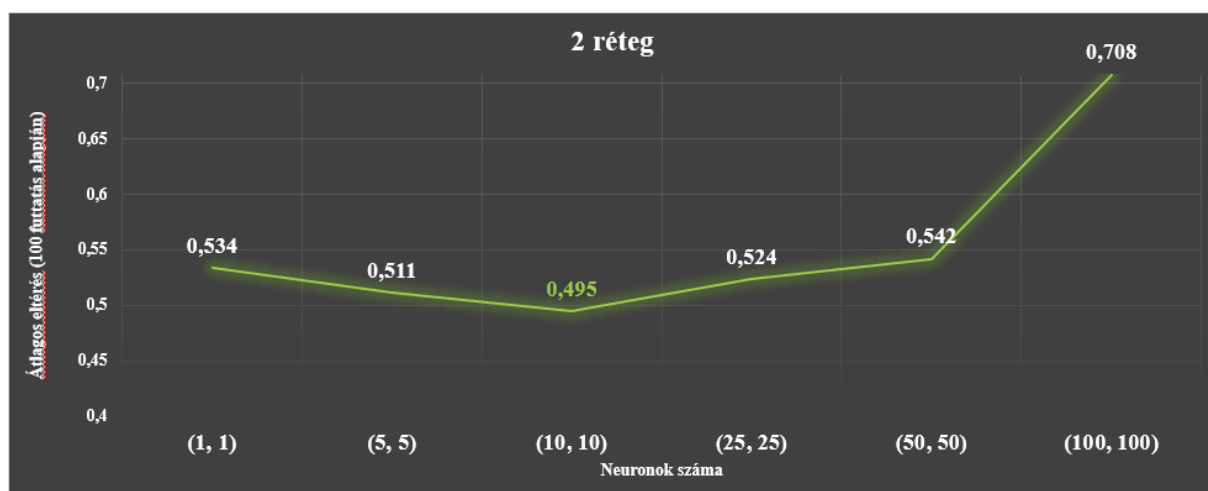
- készítettünk egy adatgyűjtő weboldalt
- tanulmányoztuk, hogy a különböző paraméterek megváltoztatása milyen hatással van a neurális háló pontosságára
- létrehoztunk egy átlagosan 0,375 l / 100km-es pontossággal becslő neurális hálót
- a neurális háló alapján létrehoztunk egy weboldalt, ahol bárki által elvégezhető a fogyasztásbecslés

Melléklet

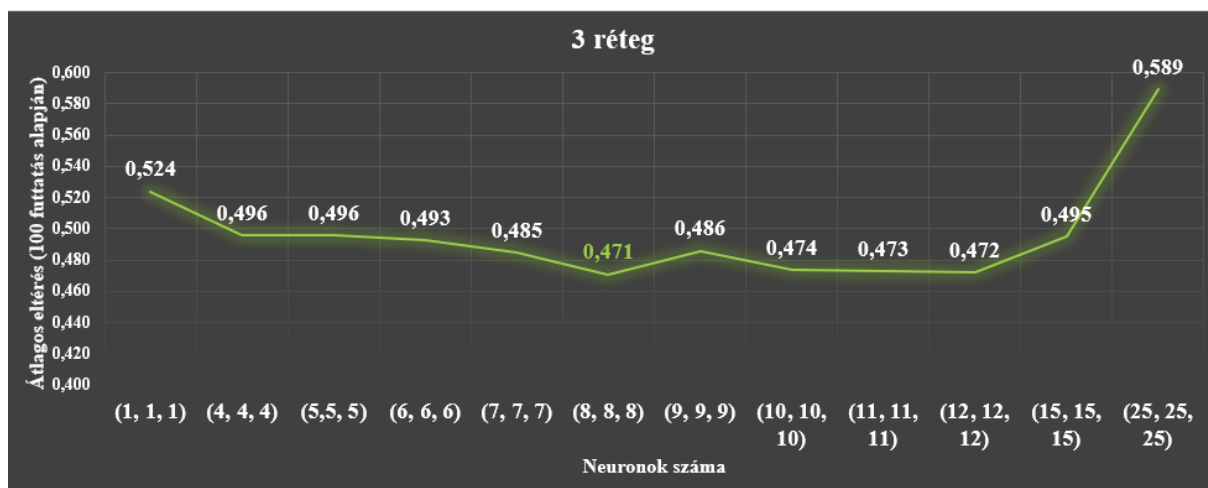
1. ábra: Egyrétegű neurális hálók átlagos eltérései



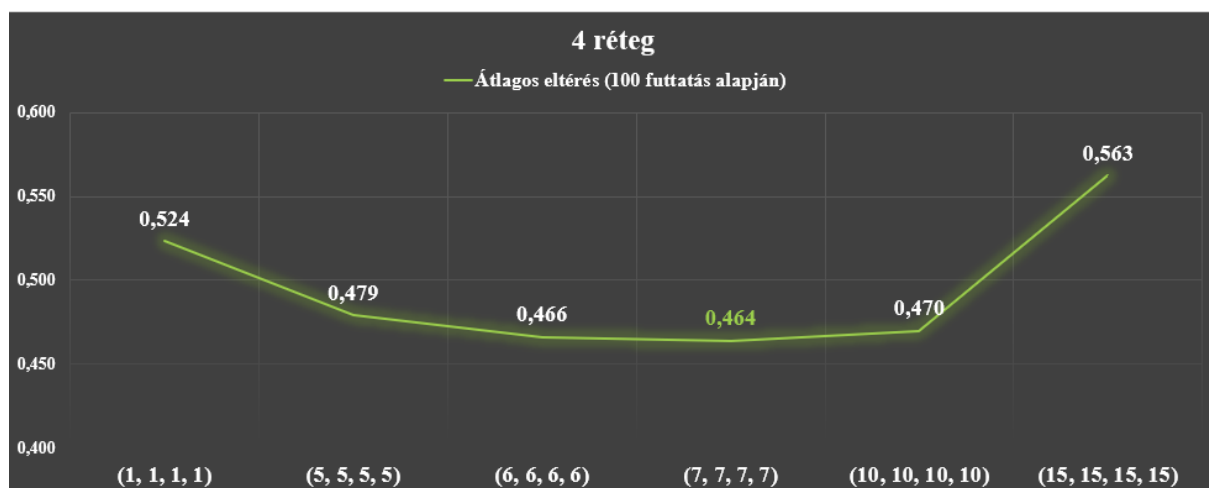
2. ábra: Kétrétegű neurális hálók átlagos eltérései



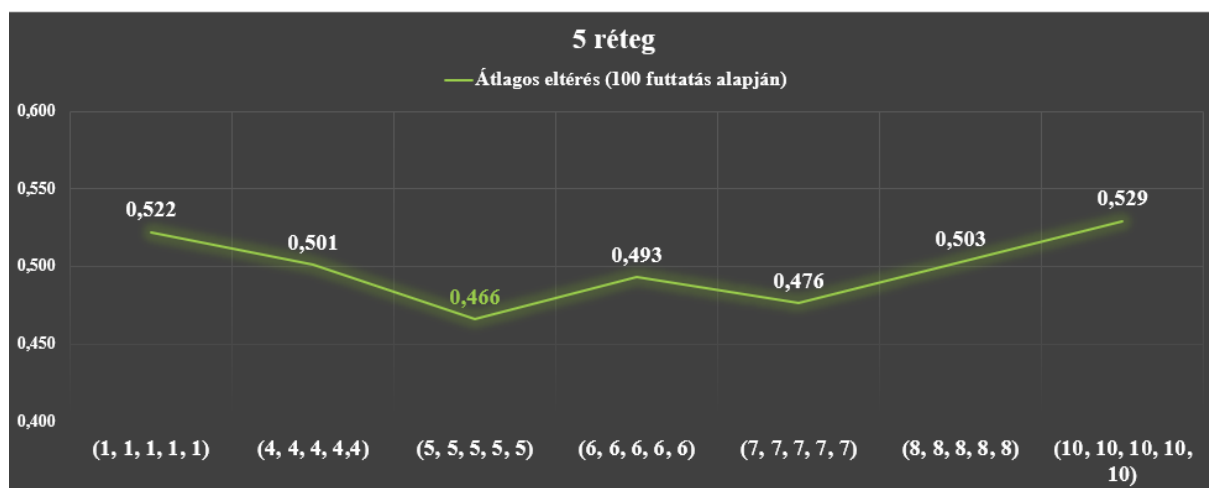
3. ábra: Háromrétegű neurális hálók átlagos eltérései



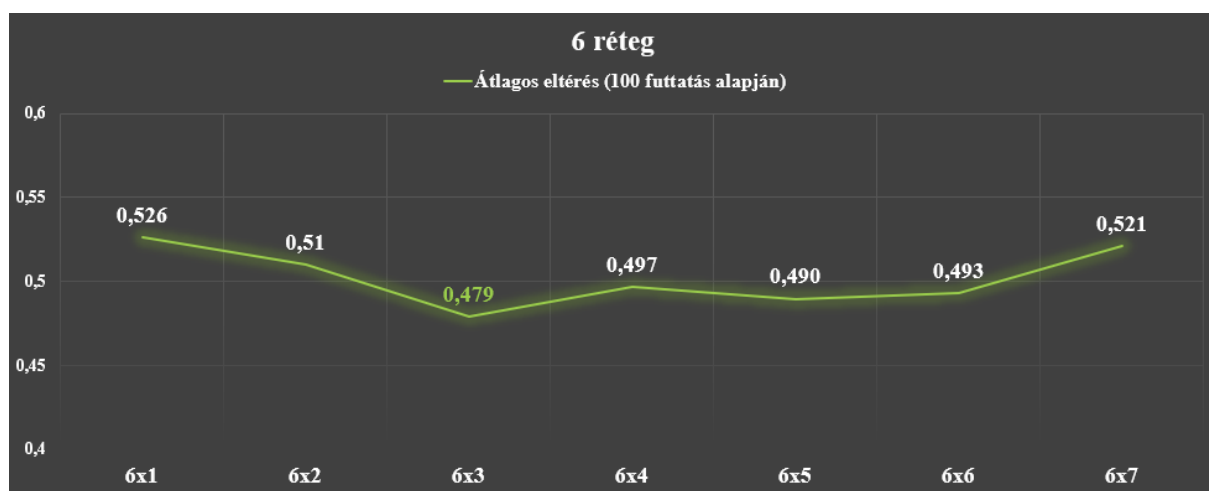
4. ábra: Négyrétegű neurális hálók átlagos eltérései



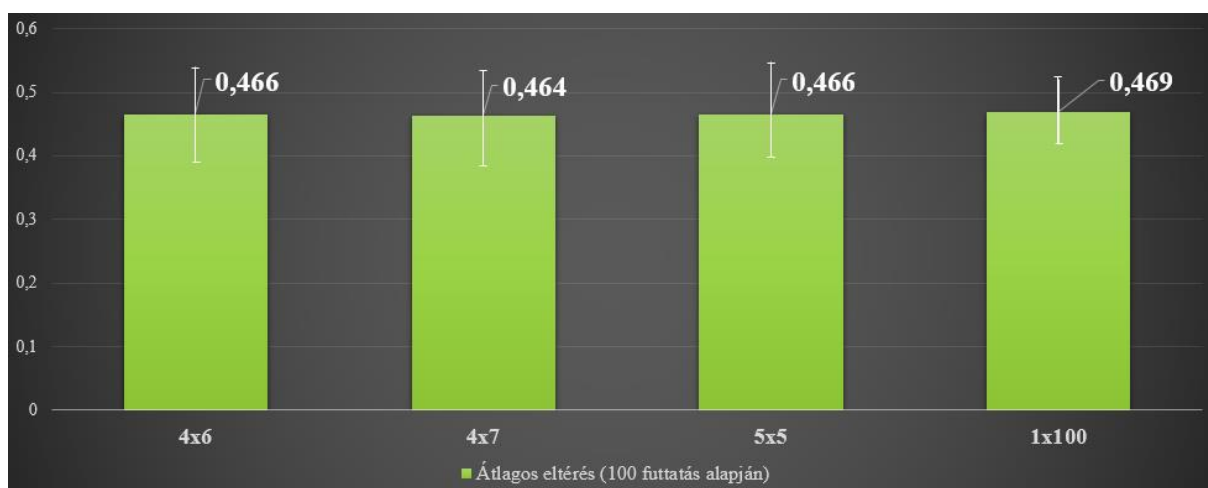
5. ábra: Ötrétegű neurális hálók átlagos eltérései



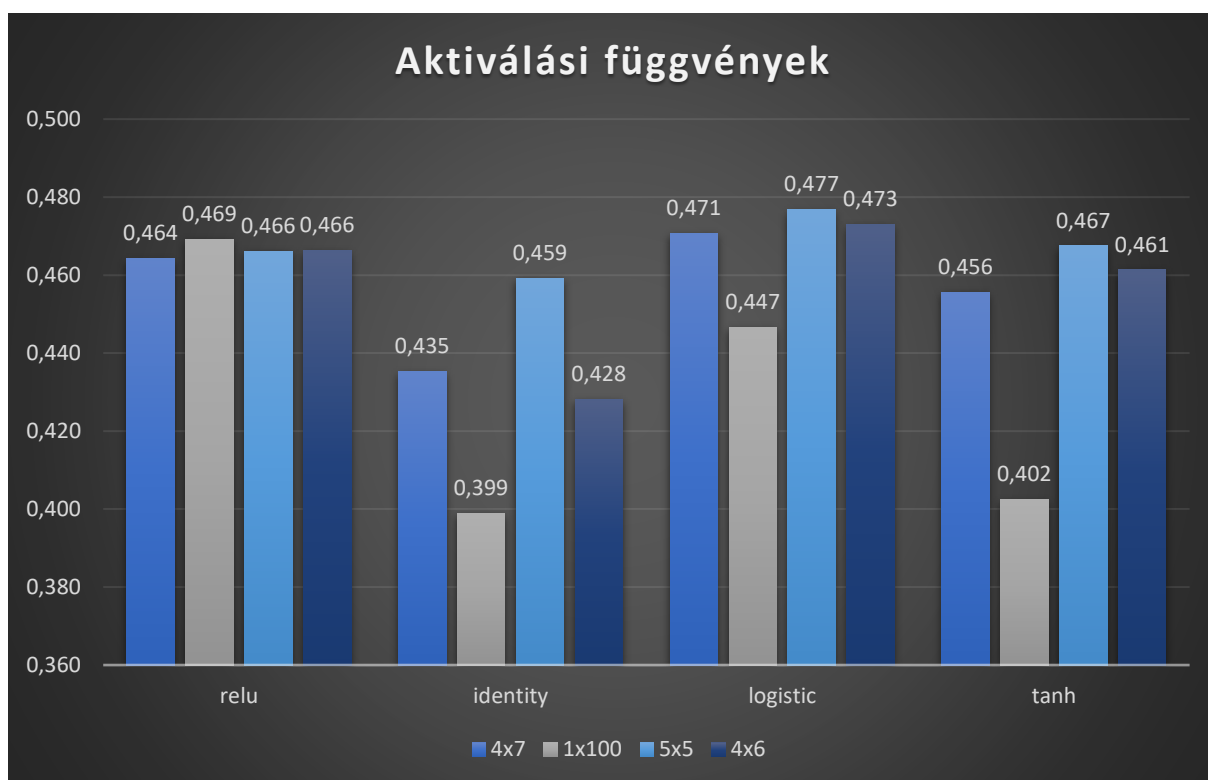
6. ábra: Hatrétegű neurális hálók átlagos eltérései



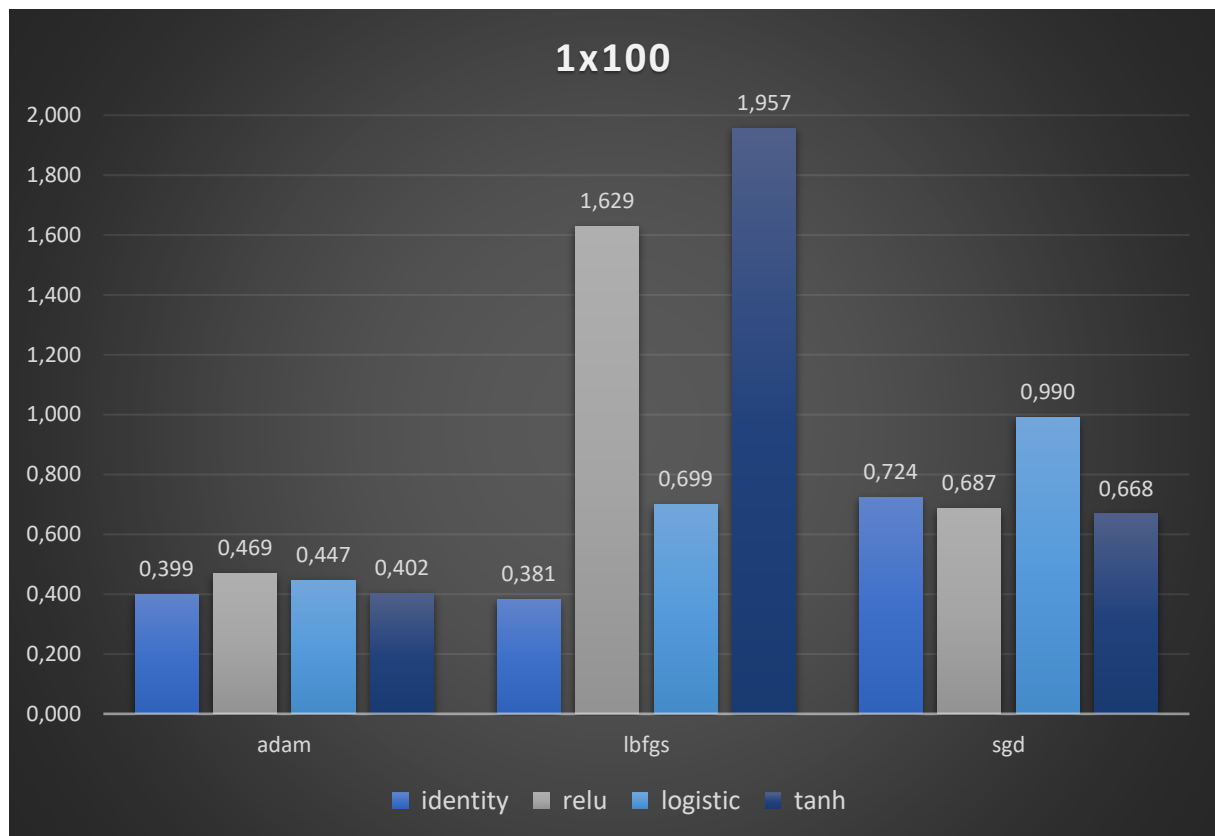
7. ábra: Az első kísérletek alapján legpontosabban becslő hálók



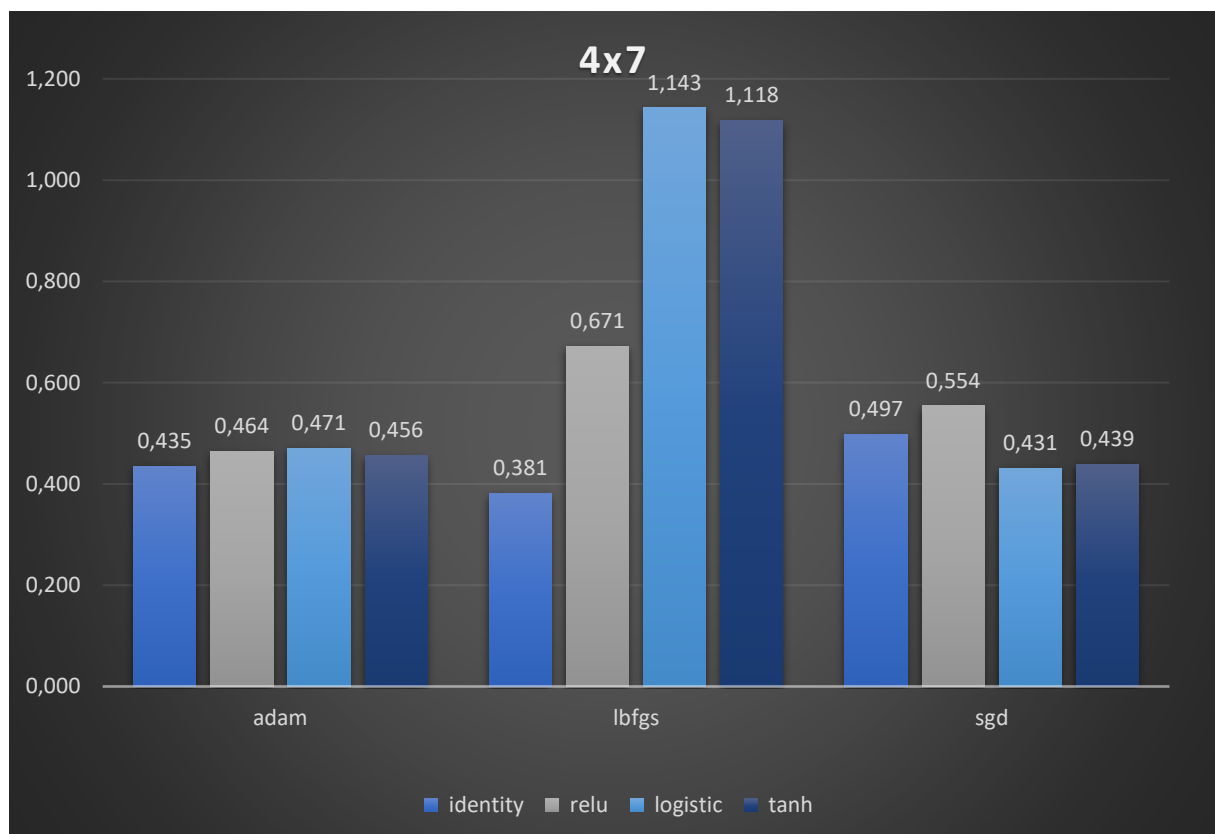
8. ábra: A különböző aktiválási függvények eltérései az első kísérletek alapján 4 legjobbnak ítélt neurális hálókkal



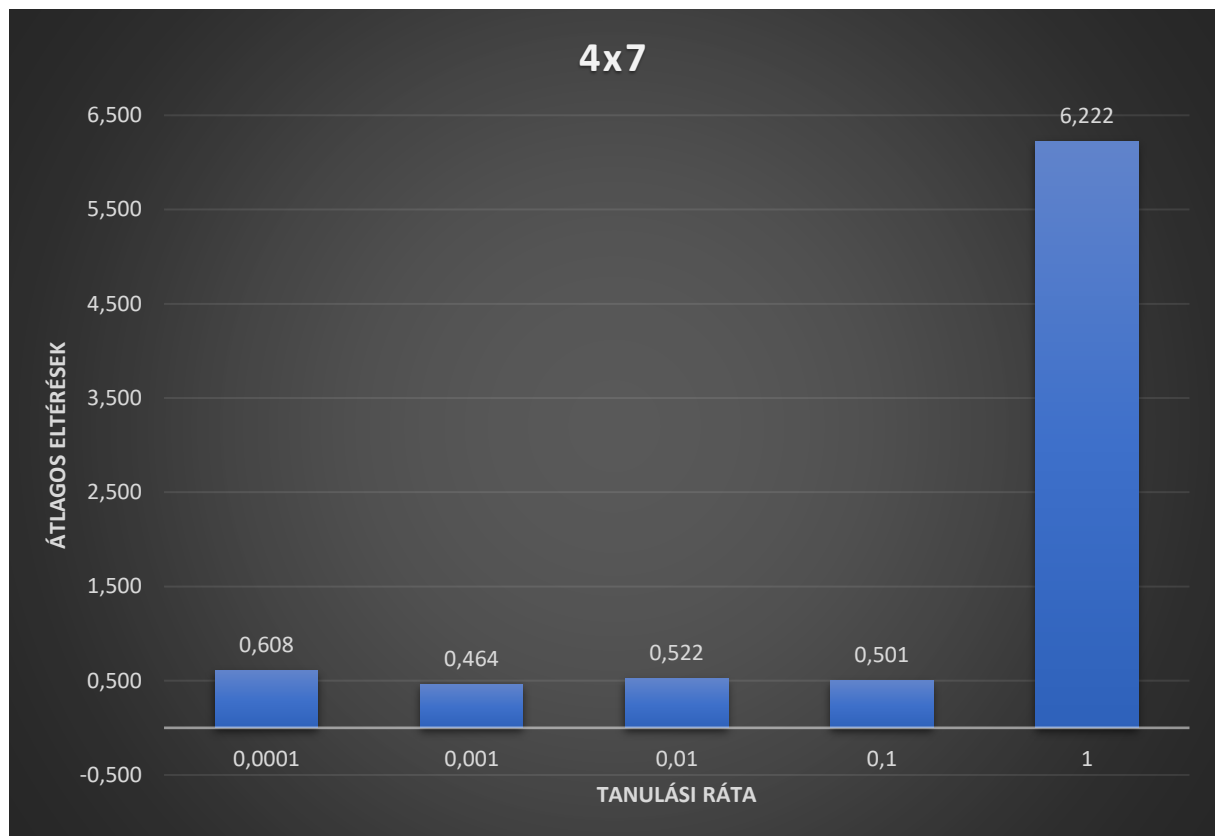
9. ábra: A különböző solverek eltérései többféle aktiválási függvénnyel az 1x100-as hálón



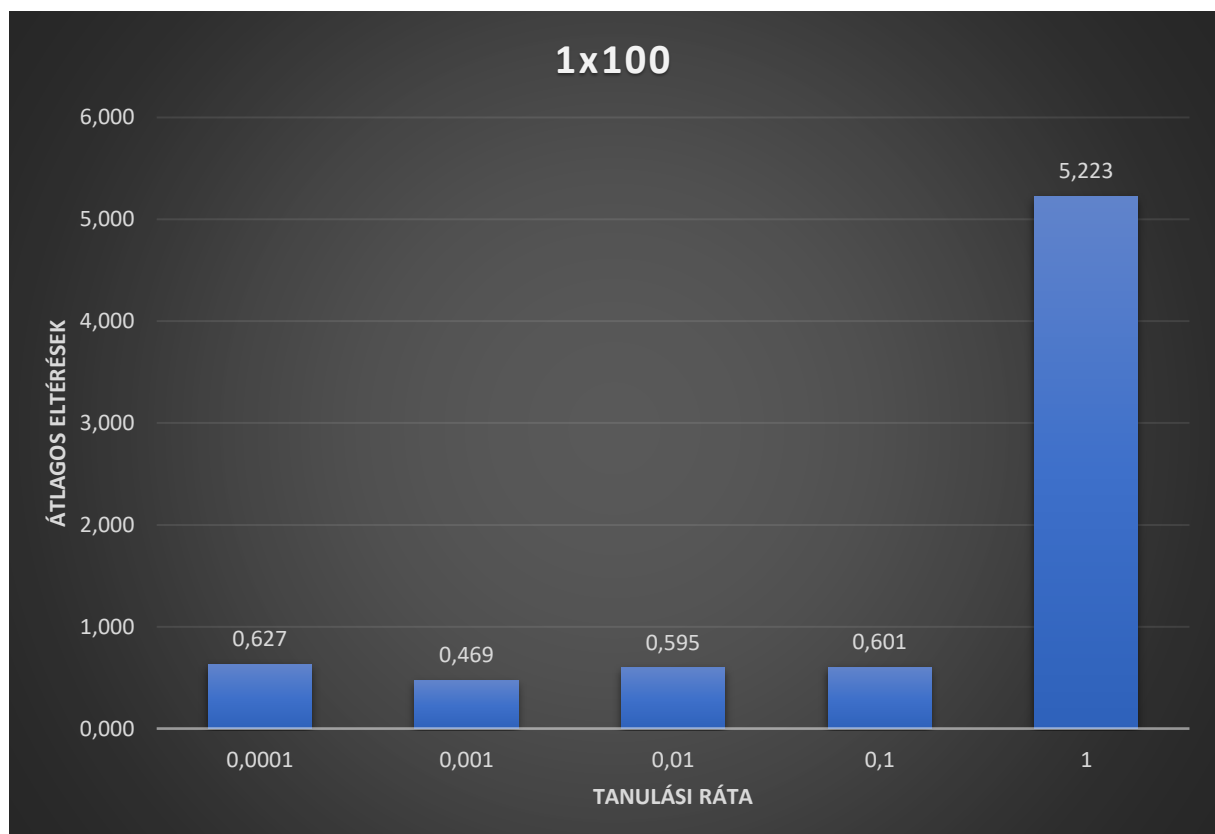
10. ábra: A különböző solverek eltérései különböző aktiválási függvényekkel a 4x7-es hálón



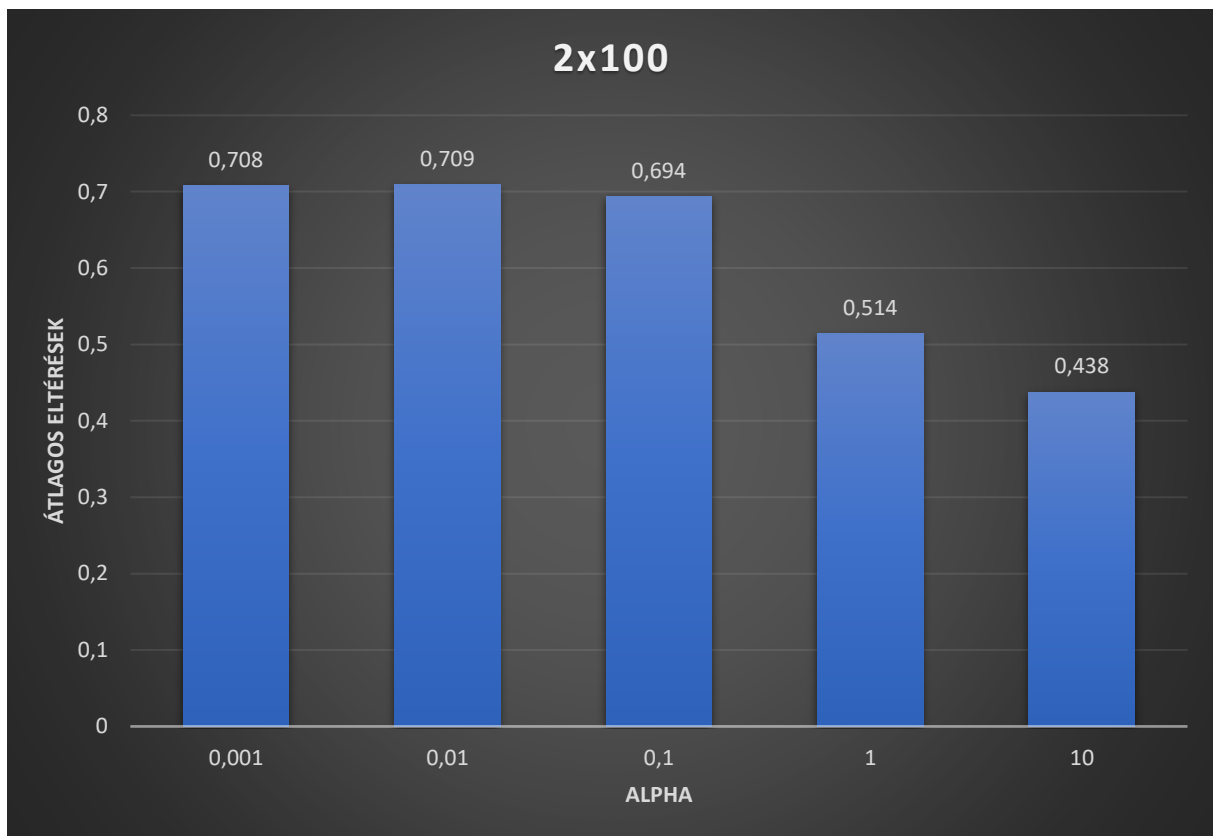
11. ábra: 4x7-es háló átlagos eltérései különböző tanulási rátákkal



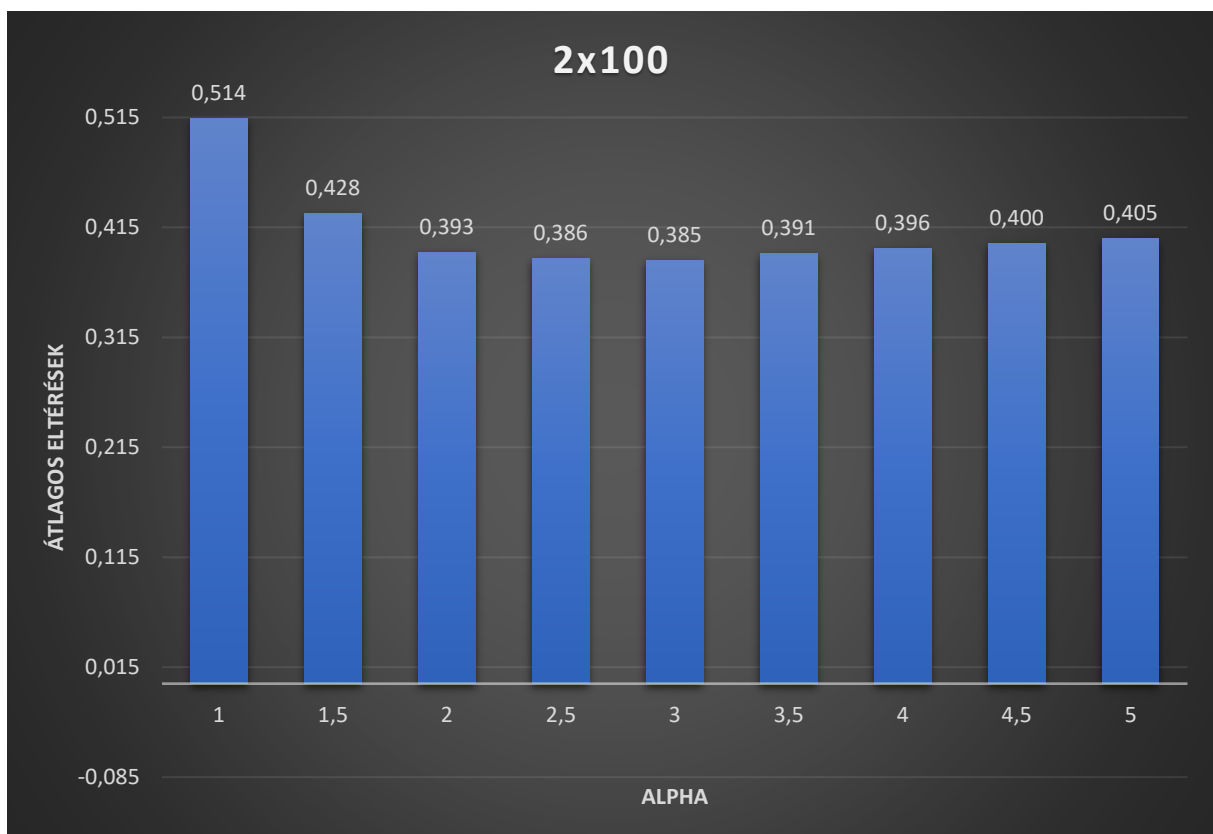
12. ábra: 1x100-as háló átlagos eltérései különböző tanulási rátákkal



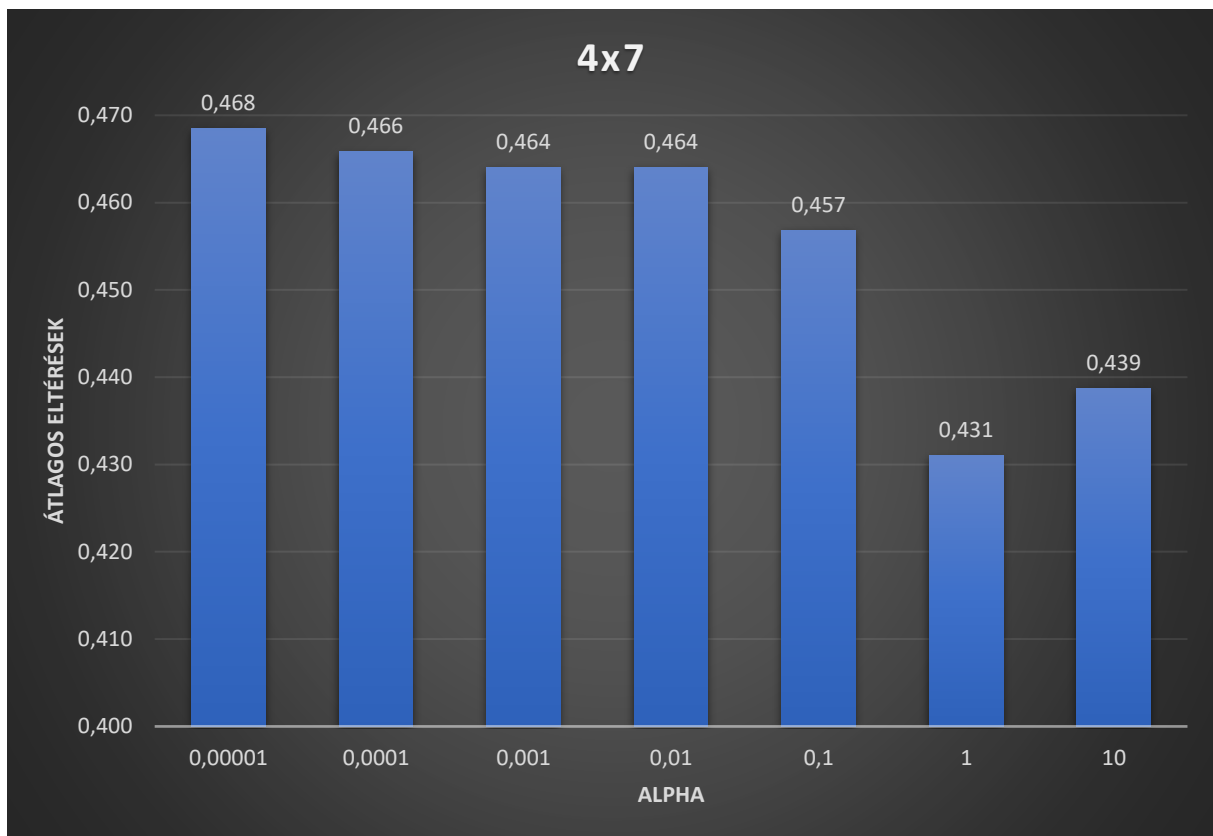
13. ábra: Túltanulás kiküszöbölése az alpha növelésével a 2x100-as hálón



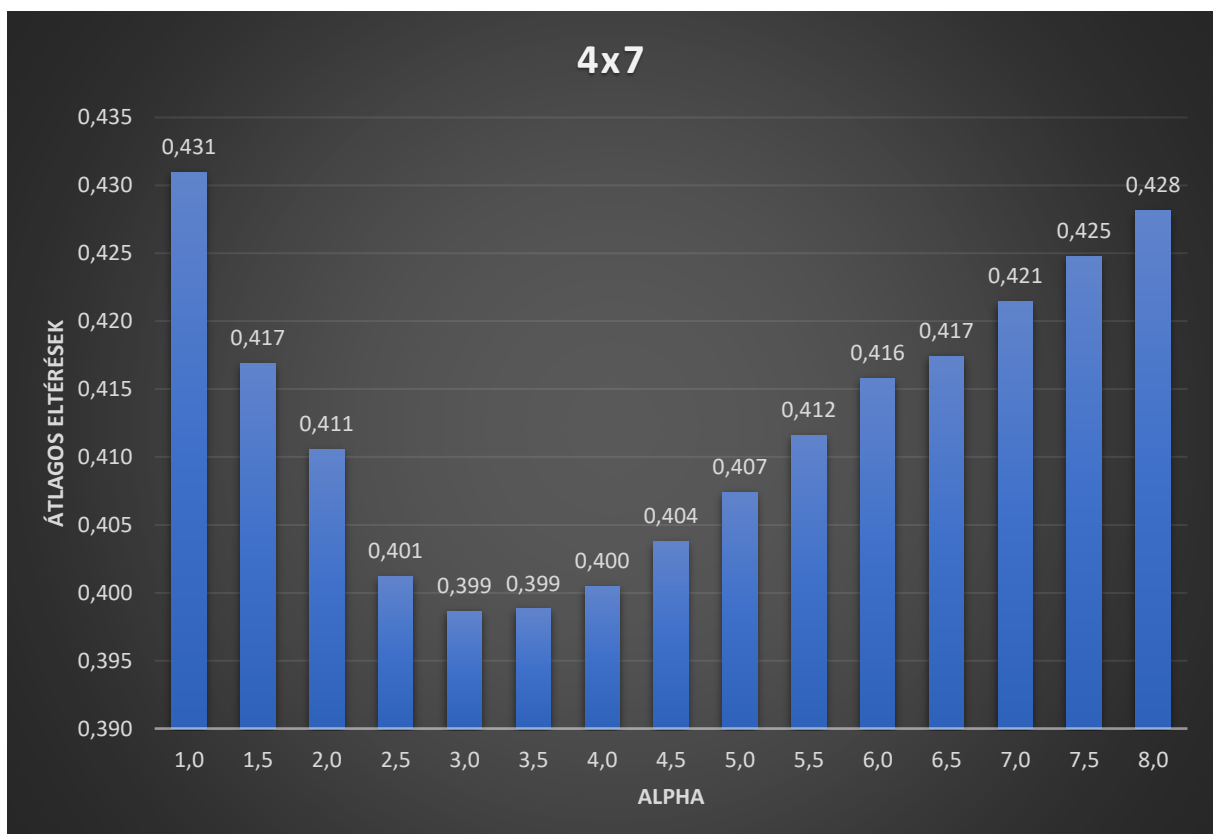
14. ábra: A 2x100-as háló átlagos eltérései az 1 és 5 közötti alphák függvényében



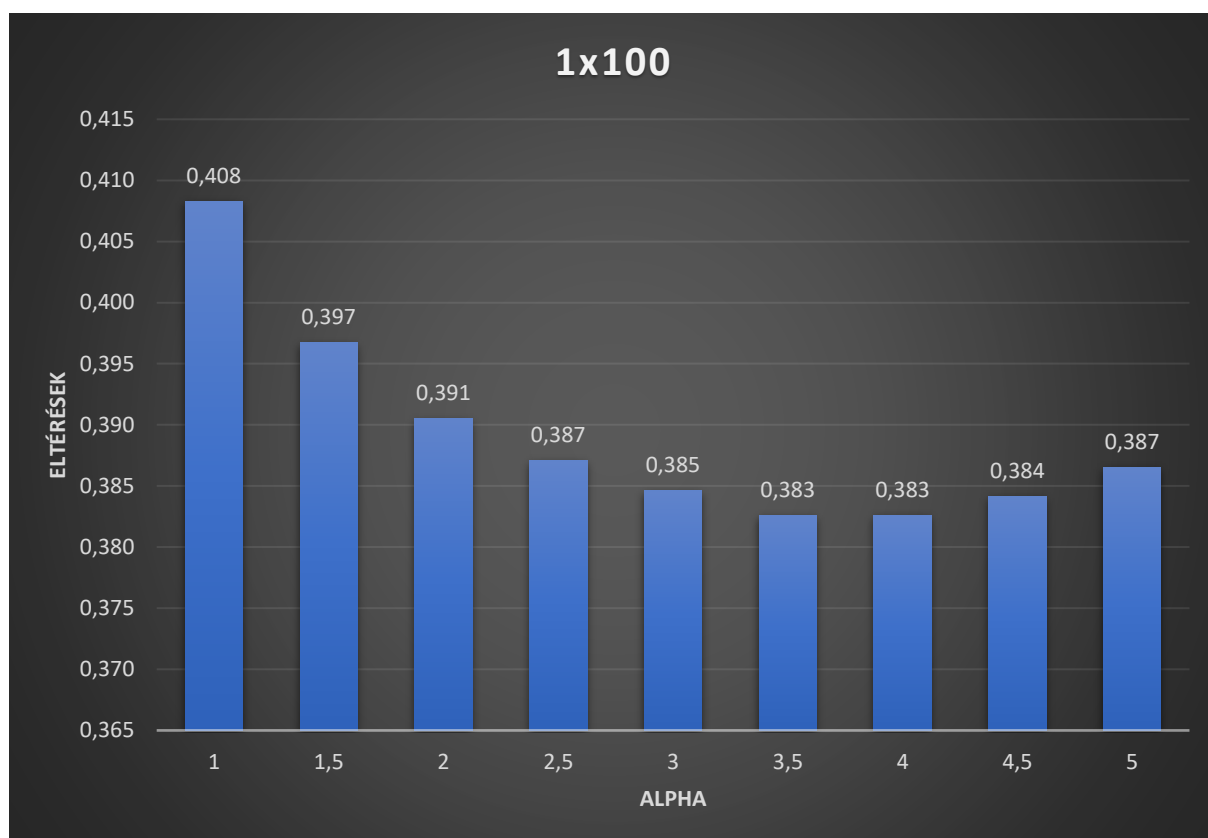
15. ábra: A 4x7-es háló átlagos eltérései az alpha függvényében



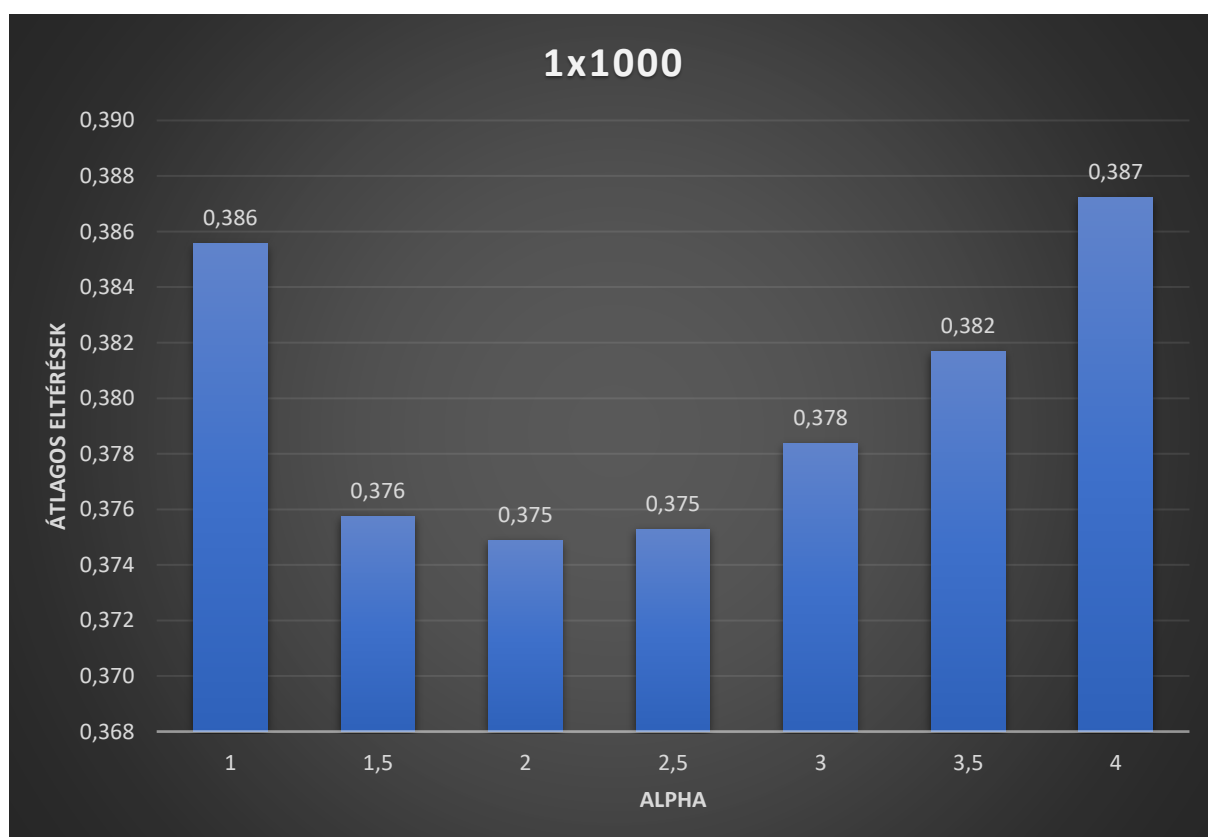
16. ábra: A 4x7-es háló átlagos eltéréseinek vizsgálata az 1 és 8 közötti alphák függvényében



17. ábra: Az 1x100-as háló átlagos eltérései megnövelt alphákkal



18. ábra: Az 1x1000-es háló átlagos eltérései megnövelt alphákkal



19. ábra: A jövőbeni alkalmazás

