



**Budapesti Műszaki és Gazdaságtudományi Egyetem
Építőmérnöki Kar
Általános és Felsőgeodézia Tanszék**

DIPLOMATERV

Neurális hálózatok alkalmazása a geodéziában

Készítette:

Zaletnyik Piroska

földmérő- és térinformatika szakos hallgató

Konzulensek:

Dr. Völgyesi Lajos

egyetemi docens

Általános és Felsőgeodézia Tanszék

Dr. Paláncz Béla

egyetemi docens

Fotogrammetria és Térinformatika Tanszék

Budapest

2003.

TARTALOMJEGYZÉK

ELŐSZÓ	5
1 BEVEZETÉS.....	6
1.1 Geoidfelület közelítése	8
1.2 EOVS-WGS84 koordináta-transzformáció	8
2 NEURÁLIS HÁLÓZATOK	9
2.1 Kialakulás, definíció.....	9
2.2 A neuronok felépítése.....	9
2.3 A hálózatok felépítése.....	12
2.4 Hálózatok tanítása.....	15
2.4.1 Ellenőrzött tanulás.....	16
2.4.1.1 Backpropagation hálózatok	16
2.4.1.2 Radiál Bázis Függvényű (RBF) hálózatok	17
2.4.1.3 Túltanulás	17
2.4.2 Nem ellenőrzött tanulás.....	19
2.5 Szoftverek a neurális hálózatokhoz.....	19
2.6 Összefoglalás	20
3 GEOIDFELÜLET KÖZELÍTÉSE	20
3.1 Bevezetés.....	20
3.2 Geoidfelület közelítése polinomokkal	21
3.3 Közelítés neurális hálózattal	23
3.3.1 Az alkalmazott hálózat	23
3.3.2 A tesztelés eredménye.....	26
3.3.3 Új neurális hálózat.....	30
3.4 Eredmények, további lehetőségek.....	32
3.5 A polinomos közelítés összehasonlítása a neurális hálózatokkal.....	33
4 EOVS – WGS-84 KOORDINÁTA-TRANSZFORMÁCIÓ	34
4.1 Bevezetés.....	34
4.2 A feladat ismertetése	34
4.3 A közelítés minősítésének kérdése	37
4.4 Általános sorokkal történő transzformáció.....	38
4.4.1 Általános sorokkal történő transzformáció segédvetületi rendszer (SVR) közbeiktatásával	39
4.4.1.1 Átszámítás WGS-84-ből EOVS-ba (transzformációs polinom együtthatóinak a meghatározása).....	39
4.4.1.2 EOVS-ból WGS-84-be történő átszámítás (transzformációs polinom együtthatóinak a meghatározása)	40
4.4.1.3 Eredmények értékelése	42
4.4.2 Általános sorokkal történő transzformáció segédvetületi rendszer nélkül.....	43
4.5 Transzformáció neurális hálózatokkal	45
4.5.1 Alkalmazott hálózatok	45

4.5.2	Transzformáció RBF hálózattal	46
4.5.3	Transzformáció Backpropagation (BPN) hálózattal.....	48
4.5.4	Skálázás.....	50
4.5.5	Polinomok közelítése neurális hálózattal.....	52
5	ÖSSZEFOGLALÁS	54
	HIVATKOZÁSOK	57
	MELLÉKLETEK	59
M8	Polinom közelítésének tanuló- és tesztpontjai.....	60
	+CD melléklet a számításokkal	

ÁBRAJEGYZÉK

1. ábra, Neuron felépítése	10
2. ábra Lépcsőfüggvény és telítéses lineáris aktivációs függvény	11
3. ábra, Tangens hiperbolikus és szigmoid aktivációs függvény.....	11
4. ábra, RBF neuron felépítése.....	11
5. ábra RBF aktivációs függvény	12
6. ábra Neurális hálózat szerkezete.....	13
7. ábra Ábrázolás gráffal.....	14
8. ábra Túltanulás	18
9. ábra Példa alultanításra (a), megfelelő tanításra (b) és túltanításra (c)	18
10. ábra Polinomos közelítés hibáinak hisztogramja	23
11. ábra A geoidfelület közelítésére alkalmazott hálózat	23
12. ábra A geoidfelület közelítése 1. rendű neurális hálózattal	26
13. ábra A geoidfelület közelítése 4. rendű neurális hálózattal	27
14. ábra 1. és 4. rendű hálózatok hibáinak hisztogramja	27
15. ábra A 4. rendű neurális hálózattal közelített felület eltérései a geoidfelülettől ...	29
16. ábra A romániai nagy eltérésű területet levágó egyenes	29
17. ábra Az új 1. és 4. rendű hálózatok hibáinak hisztogramja	31
18. ábra Eltérések az eredeti (HGTUB2000) geoid magasságok és a neurális hálózattal közelített magasságok között	32
19. ábra Polinomos (a) és neurális hálózatokkal való közelítés (b) hibáinak a hisztogramja	33
20. ábra Ellipszoidi földrajzi koordináta rendszer (Krauter, 1995).....	35
21. ábra Egységes Országos Vetület (EOV) (Krauter, 1995).....	36
22. ábra Azonos pontok az EOV rendszerben.....	37
23. ábra A tesztpontok elhelyezkedése az oda-vissza transzformációhoz	42
24. ábra Polinom és a közelítéséhez használt neurális hálózat meghatározandó paramétereinek a száma	54

Előszó

A dolgozatomban egy viszonylag új informatikai eszköznek, a mesterséges intelligencia témakörébe tartozó neurális hálózatoknak a geodéziai alkalmazhatóságát vizsgáltam meg.

Ezekre a hálózatokra jellemző, hogy tanulás útján képesek megoldást találni különböző problémákra, pl. osztályozás, függvényközelítés. Ez utóbbi tulajdonságát használtam fel én is a geodéziai problémák megoldása során és hasonlítottam össze ezt az új módszert a hagyományos megoldási eljárásokkal.

A dolgozatom bevezetésében röviden ismertetem a neurális hálózatok jelentőségét és a lehetséges geodéziai alkalmazások területeit.

A második részében áttekintést adok a neurális hálózatok kialakulásáról, felépítéséről, működéséről és fajtáiról, különös tekintettel az általam is alkalmazott hálózattípusokra.

A harmadik részben megvizsgálom milyen pontossággal illeszthető a rendelkezésre álló magyarországi geoid adatbázisra felületdarab polinomos közelítéssel, illetve neurális hálózatok felhasználásával. Ez tulajdonképpen egy felület közelítése, adattömörítés. Az itt leírt módszert ugyanígy lehet alkalmazni akár digitális domborzat modellezésre is, melynek a térinformatika szempontjából van nagy jelentősége.

A negyedik részben pedig az EOVS és a WGS-84 rendszerek közötti koordináta-transzformáció megoldását vizsgálom meg különböző neurális hálózatokkal és hasonlítom össze ezt a hagyományos hatványpolinomos eljárással, a rendelkezésemre álló 43 azonos pont alapján.

Végül összefoglalom az eredményeket és elemzem, hogy milyen esetekben érdemes ezt az új módszert használni, mikor lehet a segítségével a pontosságot jelentősen megnövelni. Ugyanakkor azt is ismertetem, hogy milyen veszélyei lehetnek a neurális hálózatok alkalmazásának (pl. kevés adatnál a túltanulás veszélye), és mikor jobb, ha a hagyományos módszereknél maradunk.

Szeretném megköszönni a két konzulensemnek a szakmai támogatását és segítségét a munkám során, Paláncz Béla egyetemi docensnek a Fotogrammetria és Térinformatika Tanszékről és Völgyesi Lajos egyetemi docensnek az Általános és Felsőgeodézia Tanszékről.

1 Bevezetés

Az informatika fejlődése nagy hatással van az összes tudományterület, így a geodézia fejlődésére is. Napjainkban egyre modernebb számítógépek jelennek meg, melyekkel korábban megoldhatatlannak látszó problémák is kezelhetővé válnak.

Az utóbbi évszázadban a geodéziának minden szakterülete tökéletesen átalakult az új technológiáknak köszönhetően. A hagyományos teodolitokat felváltották az elektronikus teodolitok. Ezzel jelentősen lecsökkent mind a mérés, mind a feldolgozás ideje. Már nincs szükség a terepen kézzel írott jegyzőkönyvekre, hiszen a műszer rögzít minden mérési adatot, melyek azután egyszerűbben és hatékonyabban feldolgozhatóak számítógéppel. Egyre nagyobb teret hódítanak a digitális térképek is, szemben a hagyományos papír alapú térképekkel. Egyszerűbb lett a térképek tárolása, sokszorosítása, különböző méretarányban történő megjelenítése. A földhivatalokban az adatokat már digitálisan tartják nyilván, nincs szükség hatalmas irattárakra, egyszerűbb az adatok lekérdezése, ellenőrzése, aktualizálása. Egyre jobban terjednek a térinformatikai rendszerek, melyeket a legkülönbözőbb területeken lehet felhasználni (pl. közművek nyilvántartása, környezetvédelem, mérnöki tervezés, közigazgatás, turizmus stb.)

Sorolhatnám még sokáig ezeket a változásokat, de céloom ezekkel a példákkal most csak az volt, hogy felhívjam a figyelmet arra mennyire fontos figyelemmel kísérni az informatika fejlődését, lépést tartani az új módszerekkel és eljárásokkal, hiszen ezek forradalmasíthatják a geodéziát.

Ennek a fontosságát szem előtt tartva igyekeztem ebben a dolgozatban megvizsgálni egy viszonylag új számítástechnikai eszközt, a mesterséges intelligencia körébe tartozó neurális hálózatoknak a geodéziai felhasználhatóságát, összehasonlítva hagyományos módszerekkel.

„A mesterséges intelligencia egyfelől egyfajta műszaki tudomány, amelynek feladata egy adott működés létrehozása minél jobb minőségben, mesterségesen megvalósítható és reprodukálható eszközökkel. Jobb híján erre a célra jelenleg a számítógépet használjuk. ...Másfelől a mesterséges intelligencia a természettudományos megismerés egy újfajta eszközének is tekinthető, amennyiben lehetőséget ad arra, hogy az ember gondolkodási mechanizmusairól szóló ismereteinket modellezzük a segítségével.” (Mérő, 1989) A tisztán műszaki gondolkodásmód alapján készített mesterséges intelligencia programok (pl. társalkodó programok) képtelenek voltak átlépni egy bizonyos határt, amit az emberi gondolkodás könnyen túlszárnyalt. Ez készítette a kutatókat arra, hogy tanulmányozzák az emberi gondolkodás mechanizmusait.

Így alakultak ki a mesterséges neurális hálózatok is. A nagyon egyszerű felépítésű idegsejtet tanulmányozva feltűnt, hogy ugyanolyan ill. hasonló sejtek különböző hálózatokba szerveződve nagyon sokféle feladatot képesek megoldani. Ennek az informatikai megvalósítása a mesterséges neurális hálózat, amely, akárcsak az ember, tanulás útján képes megoldást találni különböző problémákra.

Bevezetés

Ezek a rendszerek képesek olyan feladatokat megoldani, amelyek csak igen nehezen algoritmizálhatóak bonyolultságuk, vagy az összefüggések ismeretének a hiánya miatt. Megoldhatóak velük pl. olyan feladatok, amelyeknél nem ismerjük a kapcsolatot a bemenő és a kimenő adatok között csak sejtjük, hogy van valami összefüggés. Jellegzetes felhasználási területei pl. a szöveg-, beszéd- és hangfelismerés és optimalizálás.

Felhasználása a geodézián belül is igen széleskörű.

Neurális hálózatokat osztályozási feladatok megoldására is kiválóan lehet használni. A geodéziában például multispektrális távérzékelési képek feldolgozására, ahol a hálózat képes megtanulni, hogy mi tartozik az egyes osztályokba (pl. szántó, erdő, vízfelület stb), és képes az egész képet önállóan osztályokba sorolni. Persze az osztályozás pontosságának a minősítéséhez teszterületekre is szükség van.

Igen fontos jellegzetessége még ezeknek a hálózatoknak az approximációs vagy leképzést közelítő tulajdonság. Ennek segítségével bármilyen folytonos függvényt közelíthetünk velük ill. meghatározhatunk összetartozó be és kimeneti értékek (tanuló adatok) alapján ismeretlen leképzési függvényeket pl. a legkisebb négyzetek módszerét alkalmazva. Ezt a tulajdonságát lehet felhasználni a koordináta-transzformációk meghatározása során, ill. valamilyen felület, jelen esetben pl. a geoid felület közelítése esetén is, de a térinformatikában is igen hasznos lehet ez a digitális domborzatmodellek előállításakor.

Ezenkívül igen jól használhatók még optimalizálásra, alak és hangfelismerésre is. Ez a fotogrammetriában is alkalmazható a légi felvételeken az illesztőpontok, pl. az útkereszteződések felismerése során.

Már több geodéziai célú alkalmazás is készült. Pl. térbeli derékszögű koordináta-transzformáció meghatározására (Barsi 1999), osztályozási feladatra (Barsi, 1997), magasságok meghatározására (Veres 2002), és készült vizsgálat a neurális hálózatok térinformatikai (GIS) függvényként történő alkalmazására is (Sárközy, 1998).

Látható, hogy a geodézián belül is igen sokrétűen lehet felhasználni a neurális hálózatokat, azonban ügyelnünk kell arra, hogy ne tekintsük ezt a módszert egy univerzálisan mindenre használható eszköznek, mivel bizonyos esetekben előfordulhat, hogy olyan feladatnál alkalmazzuk, melyekre sokkal jobb, hagyományos módszerek is léteznek.

Én két aktuális problémán keresztül vizsgáltam meg a neurális hálózatok alkalmazását, és arra kerestem választ, hogy ténylegesen jobb eredményt érhetünk-e el ezzel az új módszerrel, mint a hagyományos megoldásokkal. Illetve, hol és mikor érdemes ezt a módszert alkalmazni és milyen veszélyeket rejthet magában, amire oda kell figyelni.

A vizsgált két probléma a magyarországi geoidfelület közelítése illetve EO-VGS-84 koordináta-transzformáció volt.

1.1 Geoidfelület közelítése

Manapság egyre jobban terjednek a GPS mérések. Ezekből a mérésekből ellipszoid feletti magasságot tudunk meghatározni. A mérnöki gyakorlatban azonban többnyire tengerszint feletti (vagyis geoid feletti) magassággal dolgozunk. A kettő közötti átszámításhoz ismerni kell a geoid-ellipszoid távolságát, vagyis a geoid unduláció értékét.

Rendelkezésünkre áll korábbi kutatások eredményeként egy Magyarország területére meghatározott (HGTUB2000) gravimetriai geoidmegoldás (Tóth, Rózsa 2000). Ez a $45^{\circ}30' \leq \varphi \leq 49^{\circ}$, $16^{\circ} \leq \lambda < 23^{\circ}$ nagyságú területen tartalmazza a geoid-ellipszoid távolságokat $\Delta\varphi=0'30'' \times \Delta\lambda=0'50''$ felbontással.

A geoidunduláció értéke a mérés helyén meghatározható interpolálással az ismert rácsháló pontjait felhasználva. Ez az adatbázis viszont igen sok adatot tartalmaz (211680 pont), és így nem túl egyszerű a használata terepen. Sokkal könnyebb, ha csak egyetlen függvényt kell helyette alkalmazni, ami bármilyen programozási nyelven könnyen leírható és beprogramozható akár egy GPS vevőbe is. Így a feladat tulajdonképpen egy felület közelítése regresszióval. Ez történhet a már ismert módon polinomokkal, vagy akár neurális hálózatokkal is. Ezeknek az alkalmazhatóságát, pontosságát vizsgáltam.

Munkám célja az volt, hogy olyan függvényt állítsak elő, amely megfelelő pontossággal közelíti meg a geoid felületét Magyarország területén. Egy ilyen függvénynek igen jelentős gyakorlati haszna lehet.

Ezt a problémát egy TDK dolgozat keretében kezdtem el vizsgálni 2002-ben, Völgyesi Lajos és Paláncz Béla vizsgálatait folytatva (Paláncz, Völgyesi 2002).

1.2 EOVS-WGS84 koordináta-transzformáció

Magyarországon ma egyszerre több koordináta rendszert is használnak. A két leggyakrabban alkalmazott rendszer a GPS mérések koordináta rendszere, a WGS-84 és az EOVS (Egységes Országos Vetületi) rendszer. Éppen ezért a két koordináta rendszer közötti átszámításokra gyakran van szükség.

Szabatos vetületi átszámítást lehet végezni két vetületi rendszer között, ha azonos az alapfelület és mindkét vetületen ugyanaz az egységes háromszögelési hálózat ill. annak egy-egy része van ábrázolva. Jelen esetben azonban erre nincs lehetőségünk, mivel a GPS mérések alapfelülete a WGS-84 ellipszoid, az EOVS alapfelülete pedig az IUGG/1967 ellipszoidhoz simuló új magyarországi Gauss-gömb.

Eltérő háromszögelési hálózatok vagy az alapfelületek különbözősége esetén az átszámítás azonos pontok (melyek koordinátái mindkét vetületi rendszerben ismertek) alapján történhet, pl. általános sorokkal (ill. egy másik elnevezés szerint hatványpolinomokkal). Jelen esetben 43 azonos pont állt rendelkezésemre. Ezek alapján határoztam meg az 5. fokú polinommal való közelítés együtthatóit, illetve

vizsgáltam meg a transzformációt neurális hálózatokkal, remélve, hogy ezzel az utóbbival jobb transzformációt tudok előállítani.

2 Neurális hálózatok

A következőkben a teljesség igénye nélkül a dolgozatban alkalmazott, felhasznált fogalmakat ismertettem.

2.1 Kialakulás, definíció

A „mesterséges” neurális hálózatok megalkotásához a biológiai ismeretek, illetve az idegsejt működésének a pontosabb megismerése vezetett. Az első mesterséges neuront – a Perceptron-t – Rosenblatt alkotta meg 1958-ban. Eleinte nagyon ígéretesnek tűntek a kutatási eredmények, de egy elismert szaktekintély téves vélekedése miatt a kutatások mintegy húsz évre szüneteltek a témában. A nyolcvanas évek második felétől indultak be újra a vizsgálódások a neurális hálózatokkal kapcsolatban, amikor egy nagyon fontos tanulási algoritmust (a „backpropagation” algoritmust) felfedezték. Azóta a neurális hálózatokkal kapcsolatos tudományterületek igen gyors ütemben fejlődnek. Mivel ez egy viszonylag új terület, még ma is sok kérdés vár válaszra, így a kutatókra még fontos feladatok várnak.

Neurális hálózatnak nevezzük azokat a párhuzamos információfeldolgozásra alkalmas eszközöket, melyek (Horváth 1995):

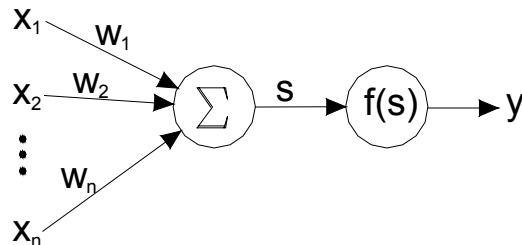
- azonos vagy hasonló műveleti elemekből, ún. neuronokból állnak,
- rendelkeznek tanulási algoritmussal és
- rendelkeznek a megtanult információt előhívó algoritmussal.

A neurális hálózatok működése két szakaszra bontható, az első a tanulás, a második a megtanult információ előhívása, alkalmazása. Az első szakasz egy lassú folyamat, amely többnyire sok iterációs lépésen keresztül történik, esetleg többszöri inicializálással az optimum elérése érdekében. A második szakasz sokkal gyorsabb, ennek köszönhető a hálózatok jó alkalmazhatósága.

2.2 A neuronok felépítése

A neurális hálózatok alapelemei a neuronok, melyeket neveznek még műveleti elemeknek ill. csomópontoknak is. Ezek párhuzamos információfeldolgozásra képes elemek. Egy neuron több bemenettel és egy kimenettel rendelkezik. A neuron meghatározza a bementi komponensek súlyozott összegét és ezen végrehajt valamilyen nem lineáris leképzést. Ez utóbbit nevezik aktivációs, transzfer vagy aktiváló függvénynek. A végeredmény a neuron kimeneti jele. Ez egy általános

neuron működése. Egy másik változat a lineáris összegzést megvalósító neuron, amikor nem történik nemlineáris leképezés. Amit tekinthetünk úgy is, hogy az aktivációs függvény az identitás. Ezt általában csak egy nagyobb hálózat kimeneteinek az előállítására használják.



1. ábra, Neuron felépítése

A fenti ábrán a neuron bemeneteit x_i jelöli, a kimeneti jel pedig y . Először a bemeneti jelek súlyozott összegei kerülnek meghatározásra:

$$s = \sum_{i=0}^N w_i \cdot x_i = w^T \cdot x$$

Lineáris összegzést megvalósító neuron esetében ez már egyben a neuron kimeneti jele is:

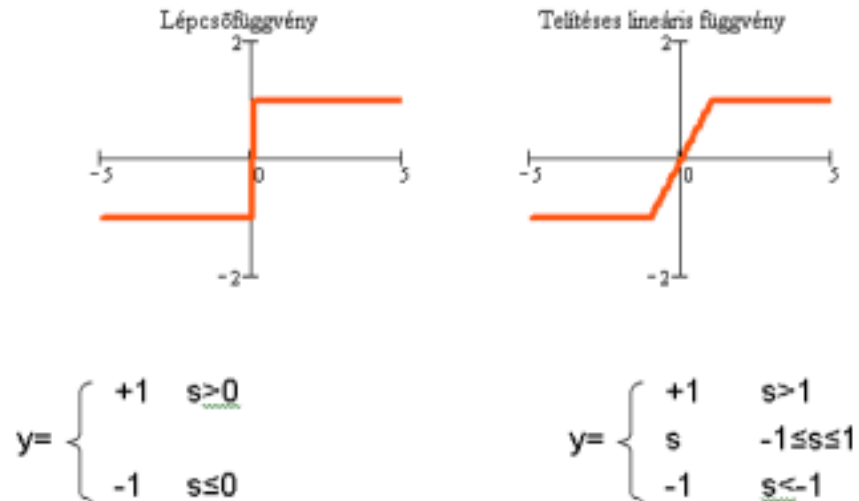
$$y = s = w^T \cdot x$$

Nem lineáris esetben a neuron kimeneti jele a következő:

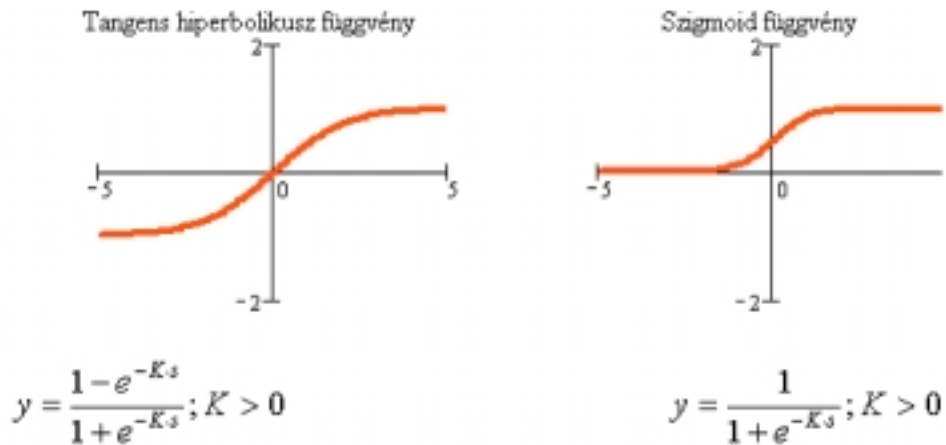
$$y = f(s) = f(w^T \cdot x)$$

ahol $f(s)$ az aktivációs függvény. Erre a célra többféle függvénytípus használata is elterjedt. A négy leggyakrabban alkalmazott függvény a lépcső- vagy szignumfüggvény, a „telítéses lineáris” függvény, a tangens hiperbolikus függvény és a szigmoid függvény.

Neurális hálózatok

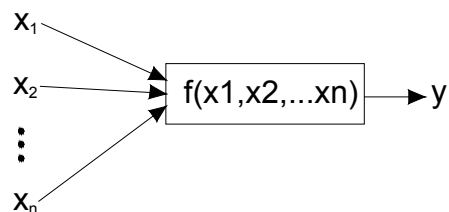


2. ábra Lépcsőfüggvény és telítésszerű lineáris aktivációs függvény



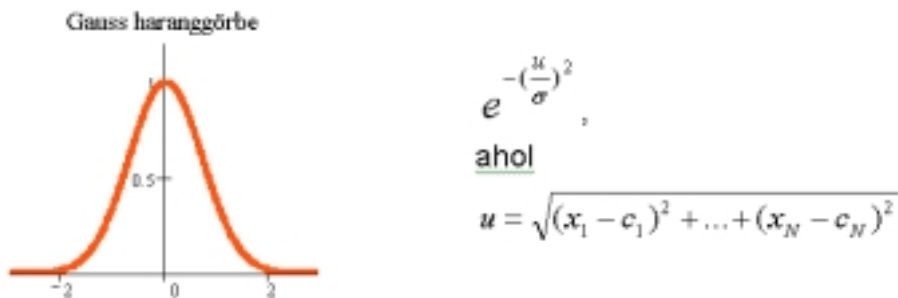
3. ábra, Tangens hiperbolikus és szigmoid aktivációs függvény

Egy másik elterjedt neuron típust használnak az ún. RBF (Radial Basis Function = radiál bázisú függvény) hálózatokban.



4. ábra, RBF neuron felépítése

Ennél a típusnál elmarad a lineáris összegzés, az összes bemenet közvetlenül az aktivációs függvénybe kerül, mely több bemenet esetén többváltozós függvény lesz. Az itt alkalmazott függvénytípus a Gauss-féle haranggörbe.



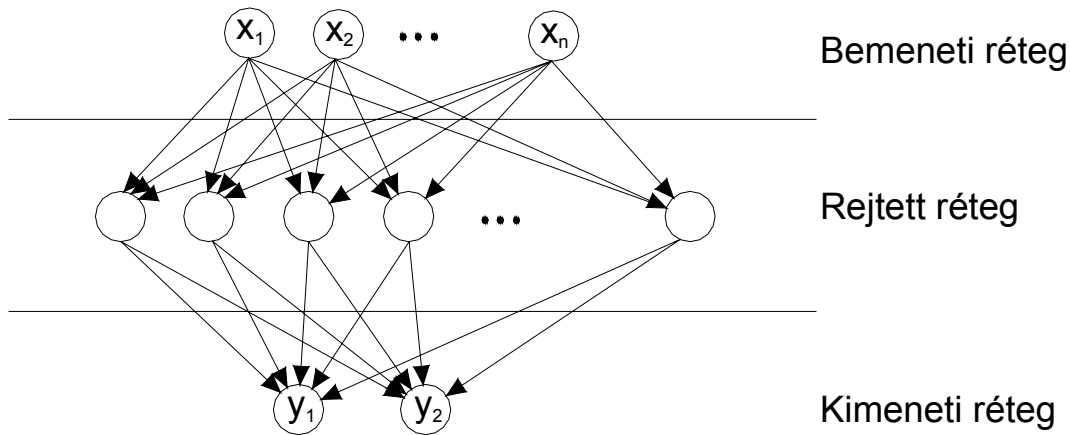
5. ábra RBF aktivációs függvény

A kimenet ebben az esetben egy a neuronra jellemző paraméter, a c vektor (középpont) és a bemeneti vektor távolságának (u) ill. a másik paraméter a görbe lapultságára/szélességére jellemző σ függvénye. Szoktak más távolságfogalmakat is használni a függvényben, nemcsak az euklideszi metrikát.

2.3 A hálózatok felépítése

A neurális hálózatok az előzőekben ismertetett neuronokból épülnek fel. Közöttük egy meghatározott szerkezeti kapcsolat, topológia szerint összeköttetések vannak. Ezek az összeköttetések az ún. feedforward típusú hálózatok (FFN) esetén egyirányú kapcsolatot jelentenek a bemenetek felől a kimenetek felé.

Ezeket a hálózatokat úgy lehet szemléltetni, mint egy több rétegből álló rendszert, ahol a rétegekben a csomópontok, azaz a neuronok helyezkednek el. Ezek a rétegek a következők: bemeneti réteg, egy vagy több rejtett réteg és egy kimeneti réteg. Az azonos rétegekben elhelyezkedő neuronok nincsenek kapcsolatban egymással (ez ad lehetőséget a már korábban említett párhuzamos feldolgozásra). A következő ábra egy általános, egy rejtett réteget tartalmazó hálózatot ábrázol.

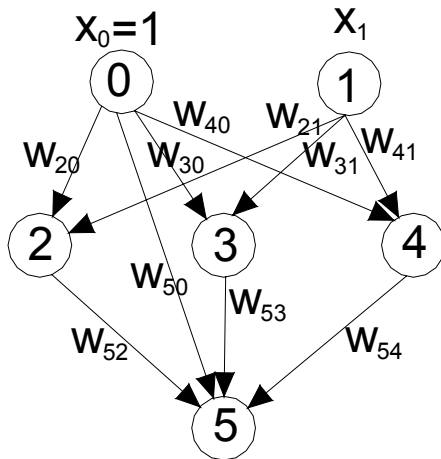


6. ábra Neurális hálózat szerkezete

A bemeneti réteg eltér a többitől, mivel ez nem egy aktív réteg, itt a neuronokban nincs átalakítás, ezek csupán mindenféle változás nélkül bemenetként szolgálnak az első aktív réteg neuronjai számára. Az ábrán az első aktív réteg a rejtett réteg. Itt megtörténik az előzőekben ismertetett jelátalakítás a neuron típusától függően. A kimeneti réteg szintén egy aktív réteg, itt általában csak lineáris összegzés történik, de itt is lehet nemlineáris kimenetet használni, pl. szigmoid aktivációs függvényt vagy telítéssel lineáris függvényt. Rejtett rétegből több is lehet, sőt akár el is maradhat, ami mindenképpen része a hálózatnak, az a bemeneti és a kimeneti réteg. Az RBF hálózat szerkezete kötöttebb ennél, itt ugyanis csak egy rejtett réteget lehet használni.

A bemeneti rétegen szoktak még egy konstans bemenetet (eltolást) biztosító plusz neuront is alkalmazni (x_0), ezt nevezzük „bias”-nak.

A hálózat kapcsolatainak a leírása történhet irányított gráffal, vagy mátrix segítségével. Nézzük meg ezt a kétféle ábrázolást egy példán bemutatva, ahol 1 bemeneti neuron van, 3 rejtett rétegbeli neuron, 1 kimeneti neuron és egy plusz neuron, a konstans bemenetet biztosító „bias”.



7. ábra Ábrázolás gráffal

$$W = \begin{matrix} & w_{20} & w_{21} & 0 & 0 & 0 \\ & w_{30} & w_{31} & 0 & 0 & 0 \\ & w_{40} & w_{41} & 0 & 0 & 0 \\ & w_{50} & 0 & w_{52} & w_{53} & w_{54} \end{matrix}$$

$$W^R = \begin{matrix} & w_{20} & w_{21} \\ & w_{30} & w_{31} \\ & w_{40} & w_{41} \end{matrix}$$

$$W^K = w_{50} \quad w_{52} \quad w_{53} \quad w_{54}$$

A fenti ábrán w_{ij} jelöli az egyes súlyokat. Ez az i -edik neuron bemenetét a j -edik neuron kimenetével összekapcsoló súly. Általános jelölés szerint az első index mindig a követő neuronra utal, a második pedig a megelőző neuron indexe. Ezt lehet egy nagy mátrixban (W) is ábrázolni (ahol 0 elem található a mátrixban, azok a neuronok nincsenek összekapcsolva), de lehet rétegek szerint szétbontva is (W^R a rejtett réteghez tartozó súlyok mátrixa, W^K a kimeneti réteghez tartozó súlyok mátrixa). $x_0=1$ a konstans bemenetet biztosító plusz neuron, a bias. Az így meghatározandó súlyok száma 10 (2*3 súly tartozik a rejtett réteghez, az egy bemenő csomópontot és a bias-t a 3 rejtett rétegbeli neuronnal összekapcsoló súlyok alapján, és 3+1 tartozik a kimeneti réteg neuronjához, 3 súly a rejtett rétegbeli neuronok és 1 a bias alapján).

Nézzük meg, hogy mi lesz a neurális hálózat kimenete, ha a rejtett rétegben szigmoid aktivációs függvényt alkalmazunk, a kimeneti rétegben pedig egy lineáris összegzést megvalósító neuront.

Először a bemeneti jelek s súlyozott összegei kerülnek meghatározásra a rejtett réteg neuronjaiban:

$$s_2 = w_{21}x_1 + w_{20}$$

$$s_3 = w_{31}x_1 + w_{30}$$

$$s_4 = w_{41}x_1 + w_{40}$$

vagy

$$s = w^R \cdot x, \text{ ahol } x = \begin{pmatrix} x_0 \\ x_1 \end{pmatrix} = \begin{pmatrix} 1 \\ x_1 \end{pmatrix}$$

Ezután az aktivációs függvénnyel történik meg a nemlineáris leképezés az előzőekben meghatározott s értékekkel. Jelen esetben az aktivációs függvény f , a szigmoid függvény.

A rejtett réteg kimeneti jelei így a következők lesznek:

$$y_2^R = \frac{1}{1 + e^{-w_{21}x_1 - w_{20}}}$$

$$y_3^R = \frac{1}{1 + e^{-w_{31}x_1 - w_{30}}}$$

$$y_4^R = \frac{1}{1 + e^{-w_{41}x_1 - w_{40}}}$$

vagy $y^R = f(s)$

A kimeneti rétegben csak lineáris összegzés történik, így hálózat kimeneti értéke, a biast is figyelembe véve a következő lesz:

$$y = w_{50} + \frac{w_{52}}{1 + e^{-w_{21}x_1 - w_{20}}} + \frac{w_{53}}{1 + e^{-w_{31}x_1 - w_{30}}} + \frac{w_{54}}{1 + e^{-w_{41}x_1 - w_{40}}}$$

A neurális hálózat tanítási algoritmus a w_{ij} súlyokat állítja elő egy konvergens iteratív eljárással a rendelkezésre álló, összetartozó x, y értékpárok, a tanító adatok alapján, úgy, hogy a hálózat x bemenethez tartozó kimenete minél jobban megközelítse a tanító értékpárok y kimeneti értékét.

2.4 Hálózatok tanítása

A neurális hálózatok fontos tulajdonsága a tanulási vagy adaptációs képesség. Ez azt jelenti, hogy a hálózat a tanulás során képes módosítani a paramétereit, annak érdekében, hogy egy kívánt célt, működést elérjen.

A neurális hálózatok tervezése több lépésben történik.

- 1) Hálózat szerkezetének megtervezése (rétegszám, neuronok száma, aktivációs függvény típusának megválasztása)
- 2) Tanító és tesztpontok kiválasztása
- 3) Hálózat tanítása
- 4) Tesztelés

A hálózat szerkezetének jó kialakítása fontos feladat, mert később a tanulás során a rendszer ezt nem változtatja, csak a neuronokhoz tartozó súlyokat, paramétereiket módosítja. Tehát tulajdonképpen egy általunk meghatározott modellt próbál meg minél jobban illeszteni egy adott feladathoz. Ám arra, hogy mekkora (hány rétegből és rétegenként hány neuronból álló) hálózatot válasszunk, hogyan

vegyük fel a kezdeti súlyokat és válasszuk ki a tanító és tesztpontokat jelenleg nincs egyértelmű szabály. Ezekre leginkább csak tapasztalati válasz adható.

Az RBF hálózatoknál csak egy rejtett réteget lehet alkalmazni, és itt a neuronszámot lehet változtatni. Az adott feladat igényeit legjobban kielégítő hálózati szerkezetet kétféle módon határozhatjuk meg. Ki lehet indulni egy nagyobb hálózatból, amiről bizonyos tapasztalatok alapján tudjuk hogy elegendő lesz, és ezután csökkentjük a neuronszámot amennyire lehet. Vagy megpróbálhatjuk kisebb hálózattal megoldani a feladatot, és ha ez nem megy, akkor növelni a neuronszámot. Szerintem először érdemes kisebb neuronzámmal próbálkozni, mert túl sok esetben nagyon nagy lesz a memória igény és a számítások is lassúak lesznek, sőt fennáll a hálózatok „túltanulásának” a veszélye. Ha a felvett hálózat nem képes megfelelő pontossággal megtanulni az adott problémát, akkor érdemes próbálkozni a neuronok számának a növelésével. Törekedni kell a lehető legnagyobb pontosságra, a lehető legegyszerűbb hálózatszerkezet mellett.

A tanulás lehet:

- ellenőrzött vagy felügyelt tanulás,
- nem ellenőrzött vagy felügyelet nélküli tanulás.

2.4.1 Ellenőrzött tanulás

Ebben az esetben rendelkezésünkre állnak összetartozó be és kimeneti értékpárok, vagy tanító adatok. Ezek alapján próbálunk meg kialakítani egy olyan hálózat szerkezetet, amivel a lehető legjobban tudjuk közelíteni az adatokat.

A tanulás egy konvergens iteratív eljárás alapján történik, amikor a rendszer meghatározza a legkedvezőbb súlyokat, paramétereket a tanító adatok alapján úgy, hogy a hálózat kimeneti értékei minél jobban közelítsék meg a tényleges kimeneti értékeket. Itt tulajdonképpen egy optimum vagy szélsőérték (minimum) meghatározásról van szó, többnyire a legkisebb négyzetek módszere alapján, mivel a tanító adatok száma általában jóval meghaladja az ismeretlen paraméterek számát (regresszió).

A hálózat tanulása nagymértékben függ a súlyok kezdeti értékétől, így érdemes többszöri inicializálással felvenni ezeket és megismételni a tanítási folyamatot.

A tanítás iterációs lépései során az aktuális súlyokat valamilyen korrekcióval módosítják, a megfigyelt hálózati hibák alapján. Ez a folyamat addig tart, amíg a hálózat elér egy hiba minimumot. Ekkor befejeződik a tanítás. Mivel itt egy minimum kereséséről van szó, ezért érdemes a többszöri inicializálás és újra tanítás, mert lehetséges, hogy egy másik kezdeti súlyfelvétel után jobb eredményt ér el a hálózat.

2.4.1.1 Backpropagation hálózatok

A leggyakrabban alkalmazott tanítási algoritmus a „backpropagation” (hiba-visszaterjesztéses) algoritmus és ennek különböző változatai. Ezt lehet általánosan

alkalmazni többrétegű (multilayer perceptron, MLP) hálózatok tanítására. Az alkalmazott aktivációs függvény itt leggyakrabban a szigmoid függvény, de lehet használni nemlinearitásként a tangens hiperbolikus függvényt is.

Az alap algoritmusnak számos változata alakult ki, melyek a konvergencia gyorsítását ill. a lokális hibaminimumokba való beragadás elkerülését szolgálják. Ilyen változatok például a momentum módszer és a Kalman szűrőn alapuló eljárás.

2.4.1.2 Radiál Bázis Függvényű (RBF) hálózatok

A backpropagation (BPN) hálózatok hátránya a viszonylag lassú tanulás. Ez főleg nagy adatmennyiség esetén okoz nehézségeket. A radiális bázis függvényű (RBF) hálózatok előnye, hogy kevesebb iteráció szükséges a tanulás során, így lényegesen gyorsabb a folyamat. Ezeknek a hálózatoknak egyszerű a felépítése, csak két aktív rétegük van: egy rejtett réteg és a kimeneti réteg. A rejtett rétegben alkalmazott aktivációs függvény valamilyen forgásszimmetrikus függvény, melynek a két paramétere a középpont- és a szélesség vagy lapultság paramétere. Leggyakrabban a Gauss-féle haranggörbét alkalmazzák a rejtett rétegben aktivációs függvényként:

A hálózat komplexitása a rejtett rétegbeli neuronok számától függ. A kimeneti réteg neuronjai már csak az RBF függvények lineáris kombinációját valósítják meg.

Az RBF hálózatoknál nagyon fontos a középpontok számának és helyének meghatározása. Az RBF hálózatokkal történő függvényközelítés a hálózatok szélességparaméterére kevésbé érzékeny, így sokszor ezt minden neuronnál azonos értékűnek veszik fel.

Ennek a két paraméternek a meghatározására több módszer is elterjedt. Nagy számú tanítópont esetén a pontokat csoportokba foglalják és minden egyes csoportnak meghatároznak egy-egy reprezentatív középpontot. Ez a csoportosítás v. klaszterizálás történhet nem ellenőrzött tanítással, melyről a következő részben lesz szó, ill. egyéb algoritmussal, pl. az ún. K-átlagképző (K-means) eljárással.

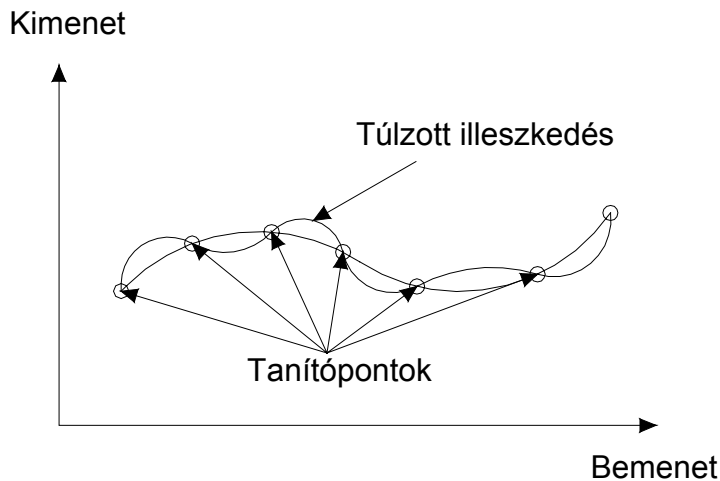
A K-átlagképző eljárás lényege, hogy úgy határoz meg K darab klaszterközéppontot, hogy a tanítópontok távolságának a négyzetösszege a hozzájuk legközelebb eső klaszterközéppontoktól minimális legyen. Először felvesszünk véletlenszerűen K darab középpontot, majd meghatározzuk minden tanítópontra, hogy melyikhez vannak legközelebb, ezek fognak egy csoportot vagy klasztert alkotni. Ezután új klaszterközéppontokat határozunk meg az egy csoportba tartozó tanítópontok átlagaként, az eljárást addig ismétljük, amíg a tanítópontok klaszterbe sorolása már nem változik.

2.4.1.3 Túltanulás

Egy neurális hálózattól azt várjuk el, hogy ne csak a tanítópontokban, hanem a tanítópontok között is jó közelítést adjon. A tanítópontok mellett ezért szükség van tesztpontokra is, hogy minősíteni tudjuk a hálózatokat. A tesztpontok olyan pontok, melyeket nem használtunk fel a tanítás során, de ismertek az összetartozó be és kimeneti értékeik. Megfelelően sok neuron felvétele esetén mind az MLP, mind az RBF hálózatok alkalmasak interpolációra, vagyis tökéletesen meg tudják tanulni a

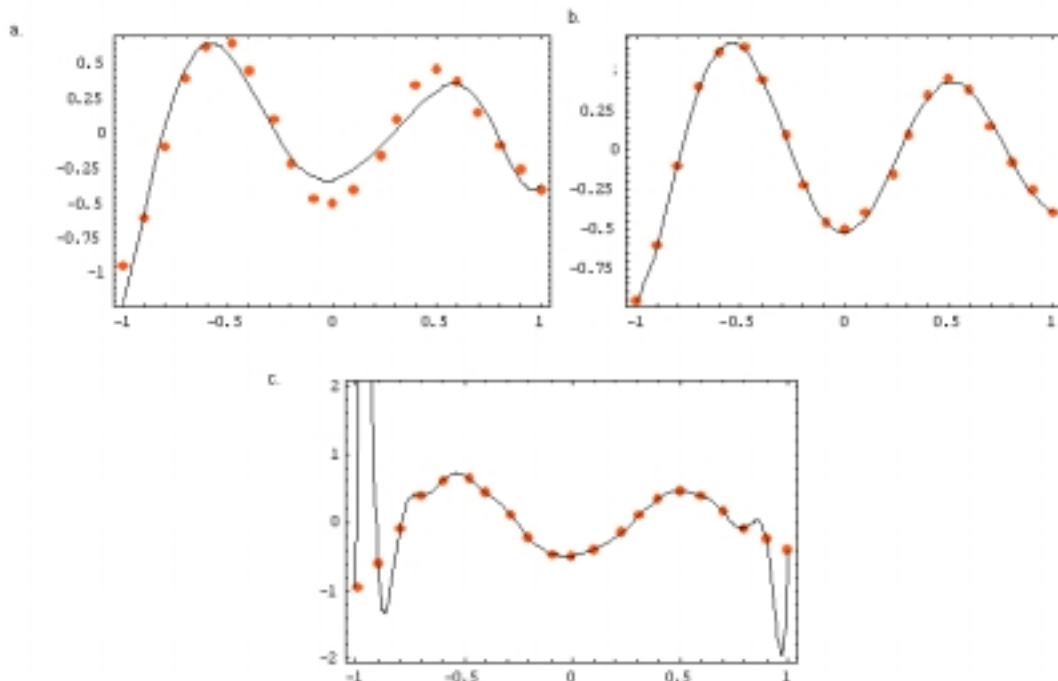
tanítópontok adatait. Ha nincsenek tesztpontjaink az ellenőrzéshez, akkor viszont könnyen túltaníthatjuk a hálózatot.

A túltanítás azt jelenti, hogy míg a tanítópontok hibája egyre csökken, addig a tesztpontok hibája egyre nagyobb lesz, a hálózat túlzottan illeszkedik a tanítópontokra. Ezt úgy lehet elképzelni, mint polinomos regressziónál, ha kevés pontra illesztünk nagy fokszámú polinomot.



8. ábra Túltanulás

Nézzünk meg egy egyszerű egyváltozós példát a hálózatok alul és túltanulására.



9. ábra Példa alultanításra (a), megfelelő tanításra (b) és túltanításra (c)

Ebben a példában 20 tanító adat állt rendelkezésre. Az első képen 2 neuront alkalmaztam a tanításra, a másodikon 3-at a harmadikon pedig 40-et. Látszik, hogy túl kevés neuron esetében még nem tudta megtanulni a hálózat a kapcsolatot

megfelelő pontossággal. 3 neuron esetében kis eltéréseket kaptunk a tanítópontokban és nagyon jó a hálózat általánosító képessége. Túl sok neuron esetében pedig a tesztpontokra tökéletesen illeszkedik a görbe, viszont rossz az általánosító képessége, különösen az intervallum szélein és így nem túl jól alkalmazható a tanítópontokon kívül.

2.4.2 Nem ellenőrzött tanulás

A neurális hálózatok tanításának másik módja a nem ellenőrzött vagy nem felügyelt tanítás. Ellenőrzött vagy felügyelt tanításnál összetartozó értékpárok voltak adottak és ezek alapján határoztuk meg a hálózat súlyait, hogy minél jobban közelítsék meg a kívánt kimeneteket. A nem felügyelt tanításnál nem ismerjük a bemenetekhez tartozó kívánt válaszokat. Nincs ellenőrzési lehetőségünk, hogy jól működik-e a hálózat. A bemenetek alapján kell kideríteni, hogy van-e közöttük valami hasonlóság, ami alapján osztályokba vagy klaszterekbe lehetne sorolni őket.

Hasonlóan az előzőekben ismertetett K-means algoritmushoz, itt is klasztereket és klaszterközéppontokat kell meghatározni. Először meg kell határozni, hogy hány osztályba szeretnénk sorolni az adatokat, majd mindegyik osztályhoz meg kell határozni egy középpontot és a hozzájuk legközelebb eső adatok esnek ebbe a klaszterbe. Ezek után módosulnak a klaszterközéppontok és ismétlődik az eljárás amíg már nem történik változás. Előfordulhat, hogy a szükségesnél több osztályt adtunk meg, és üres osztályok jönnek létre. Az ezekhez tartozó középpontok, a halott csomópontok, vagy kódvektorok. Nem felügyelt tanítású hálózatoknál lehetőség van ezeknek a halott csomópontoknak az eltávolítására is.

Ezt a módszert is lehet alkalmazni az RBF hálózatok középpontjainak a meghatározására.

2.5 Szoftverek a neurális hálózatokhoz

Neurális hálózatok felépítéséhez több szoftver is létezik. Vannak kifejezetten neurális hálózatokra kifejlesztett szoftverek, mint például a Statistica Neural Networks, Neural Works II. És vannak korábbi szoftverekhez készített neurális hálózat kiegészítő modulok. Ilyen például a Braincel For Excel, a Matlab ill. a Mathematica szoftverek kiegészítő modulja.

A geodéziai feladatok neurális hálózatokkal való megoldása során én a Mathematica szoftvert és az ehhez tartozó neurális hálózatok kiegészítő modult használtam. A Mathematica előnye, hogy nem csak numerikus, hanem szimbolikus számításra is képes. A legtöbb rendszer úgy működik, hogy meg kell adni a bemenő számadatokat, és a rendszer kiadja az eredményt szám formában, anélkül, hogy tudnánk közben pontosan mi is történt a hálózatban, mint egy fekete dobozban. Ezzel szemben a Mathematica-ban a bemenő adatok lehetnek változók is (pl. x, y) és ilyenkor eredményként a leképzés függvényét kapjuk meg ($f(x, y)$), amit később máshol is közvetlenül felhasználhatunk.

2.6 Összefoglalás

Az előzőekben igyekeztem ismertetni a neurális hálózatok főbb tulajdonságait, a neuronok felépítését, a hálózatok szerkezetét, főbb fajtáit. Ez persze csak egy rövid ismertető, sok egyéb hálózattípusról nem esett szó így például a dinamikus hálózatokról sem. De mivel dolgozatom elsődleges célja a geodéziai alkalmazások vizsgálata, így legfőképp csak azokra a hálózatokra tértem ki, melyeket én is alkalmaztam a vizsgálataim során. Ezeknek a megértéséhez szerettem volna egy áttekintést nyújtani. A tanítás pontos folyamatát sem ismertetem, de ezekről bőséges információk állnak rendelkezésre a kifejezetten neurális hálózatokról szóló könyvekben. Leginkább persze angol nyelven, de van néhány magyar nyelvű tankönyv is hozzá(Horváth 1995, Paláncz 2003).

3 Geoidfelület közelítése

3.1 Bevezetés

Most térjünk át a neurális hálózatok geodéziai alkalmazásaira. Az első probléma, mellyel foglalkoztam a magyarországi geoidfelület közelítése volt.

Kétféle földalokról beszélünk a geodéziában. Az egyik a Föld fizikai (valóságos) alakja, a tengerek és szárazföldek felülete, a másik a Föld elméleti v. matematikai alakja. Ez utóbbi a szabad folyadékfelszín egyensúlyi alakja, ha arra csak a gravitáció hat. Az ilyen felület a nehézségi erő potenciáljának szintfelülete. Azt a szintfelületet, amely valamely tenger középszintje közelében kijelölt ponton halad át, geoidnak nevezzük, és ezt tekintjük a Föld matematikai (elméleti) alakjának. (Krauter, 1995)

A geoid meghatározása történhet fizikai-geodéziai módszerekkel. Ezeknek a meghatározásoknak közös jellemzője, hogy a meghatározást fizikai feladatként oldják meg, vagyis meghatározzák a földi nehézségi erőter egyik kiválasztott szintfelületét. Ehhez fel lehet használni a nehézségi gyorsulás méréseken alapuló gravimetriai módszereket, vagy a Föld mesterséges holdjainak észleléseit. (Bíró, 1996).

Ilyen geoidmegoldást többet is készítettek Magyarország területére, egyre jobban finomítva, pontosítva a meghatározást. Az általam felhasznált HGTUB2000 gravimetriai geoidmegoldás (Tóth, Rózsa 2000) a $45^{\circ}30' \leq \varphi \leq 49^{\circ}$, $16^{\circ} \leq \lambda < 23^{\circ}$ nagyságú területen tartalmazza a geoid-ellipszoid távolságokat $\Delta\varphi=0'30'' \times \Delta\lambda=0'50''$ felbontással, összesen 211680 pontban.

A hagyományos geodéziai mérésekkel tengerszint feletti magasságokat tudtak meghatározni, ami tulajdonképpen a geoid feletti magasság. A napjainkban egyre

jobban terjedő GPS (műholdas helymeghatározó) mérésekből viszont csak ellipszoid feletti magasságokat tudunk meghatározni. Sokszor szükséges a kettő között átszámítást végezni, ehhez viszont ismerni kell e kettő távolságát, a geoidundulációt.

Mint említettem rendelkezésre áll hazánk területén egy geoidmegoldás, ami több mint 200000 pontban tartalmazza ezeket az értékeket. Ám a méréseket nem pont ezekben a rácsháló pontokban végezzük, így szükség van valamilyen interpolációs vagy regressziós módszerre, hogy az adott mérési helyen is ki tudjuk számítani a geoidunduláció értékét. Végezhetünk hagyományos lineáris interpolációt ezek között a pontok között, ezzel csupán a nagy adatmennyiség a probléma, a tárolás és számítás nehézsége. Csökkenteni lehetne az adatok mennyiségét, ha approximációt végeznénk, és előállítanánk valamilyen a felületet közelítő függvényt. Erre egy hagyományos megoldás, ha polinomokkal közelítjük a felületet, de ugyancsak megoldást jelenthetnek a neurális hálózatok is a már megismert approximációs tulajdonságaik miatt.

3.2 Geoidfelület közelítése polinomokkal

A geodéziában gyakori feladat, hogy valamilyen általános felületet matematikai módszerekkel szeretnénk leírni. Általában a meghatározni kívánt felületről diszkrét pontokban rendelkezünk információval, pl. egy magasságmodell előállításakor sok terepi pontban a megmért magasság értékeivel.

Ha az előállított felület átmegy a megadott pontokon, akkor interpolációról van szó, ha csak közelíti a pontbeli értékeket, akkor approximációról vagy regresszióról.

Regresszió esetén felvesszünk valamilyen függvénytípust, és ezt illesztjük az adatokra, úgy választva a függvények paramétereit, hogy minél kisebb hibával illeszkedjenek a felületre.

A regresszió egyik hagyományos módszere az n-ed fokú polinom illesztése. n=2 esetben a polinom a következő alakban írható fel:

$$z = a_0 + a_1 \cdot x + a_2 \cdot y + a_3 \cdot x^2 + a_4 \cdot x \cdot y + a_5 \cdot y^2$$

Itt hat együttható (a_i) meghatározására van szükség. Ha hat darab pontban áll rendelkezésünkre a magasságérték vagy az egyéb keresett paraméter és ezek lineárisan függetlenek, akkor ez a feladat egyértelműen megoldható. Fel lehet írni hat egyenletet hat ismeretlennel. Ebben az esetben interpolációról beszélhetünk, mert az ismert pontokban visszkapjuk a kívánt függvényértékeket.

Ha hatnál több pont áll rendelkezésünkre, akkor kiegyenlítésre van szükség, ebben az esetben már regresszióról beszélünk, ill. ha lehetőség van rá, akkor magasabb fokszámú polinomot is illeszthetünk az adatokra.

A geoidfelület közelítésénél geoid-ellipszoid távolságok álltak a rendelkezésemre összesen 211680 pontban. Elvileg erre már egy elég magas fokszámú polinomot is lehetne illeszteni, de bizonyos fokszám fölött már rosszul kondicionált lesz az egyenletrendszer együttható mátrixa, és nem egyértelmű a meghatározás.

Geoidfelület közelítése

A regressziót a Mathematica szoftver segítségével végeztem. A közelítésben hatodfokú polinomig tudtam elmenni, utána az egyenletek már nem oldhatóak meg egyértelműen, mert a rendelkezésre álló mátrix rangja kevesebb, mint a meghatározandó ismeretlenek száma.

Az alkalmazott hatodfokú regressziós polinom a következő:

$$\begin{aligned}
 N = & a_0 + a_1 \cdot \varphi + a_2 \cdot \lambda + a_3 \cdot \varphi^2 + a_4 \cdot \varphi \cdot \lambda + a_5 \cdot \lambda^2 + a_6 \cdot \varphi^3 + a_7 \cdot \varphi^2 \cdot \lambda + a_8 \cdot \varphi \cdot \lambda^2 + a_9 \cdot \lambda^3 \\
 & + a_{10} \cdot \varphi^4 + a_{11} \cdot \varphi^3 \cdot \lambda + a_{12} \cdot \varphi^2 \cdot \lambda^2 + a_{13} \cdot \varphi \cdot \lambda^3 + a_{14} \cdot \lambda^4 + a_{15} \cdot \varphi^5 + a_{16} \cdot \varphi^4 \cdot \lambda + a_{17} \cdot \varphi^3 \cdot \lambda^2 \\
 & + a_{18} \cdot \varphi^2 \cdot \lambda^3 + a_{19} \cdot \varphi \cdot \lambda^4 + a_{20} \cdot \lambda^5 + a_{21} \cdot \varphi^6 + a_{22} \cdot \varphi^5 \cdot \lambda + a_{23} \cdot \varphi^4 \cdot \lambda^2 + a_{24} \cdot \varphi^3 \cdot \lambda^3 \\
 & + a_{25} \cdot \varphi^2 \cdot \lambda^4 + a_{26} \cdot \varphi \cdot \lambda^5 + a_{27} \cdot \lambda^6
 \end{aligned}$$

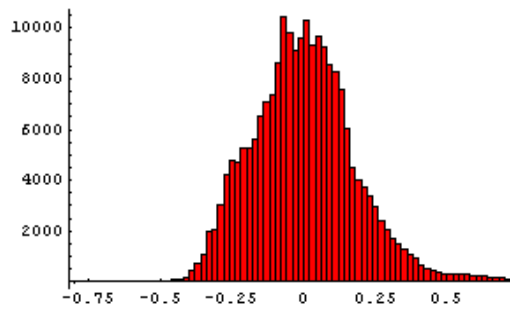
A meghatározott paraméterek pedig a következők:

a ₀ = -794 675 638,612036	a ₁₅ = 18,920421970755942
a ₁ = 100 901 202,16686144	a ₁₆ = 7,035061508113656
a ₂ = 1 224 670,6902528955	a ₁₇ = - 5,8687518268301995
a ₃ = - 5 300 492,862811941	a ₁₈ = - 1,1677857416912019
a ₄ = - 304 504,2887170478	a ₁₉ = - 0,5682650395086407
a ₅ = 205 113,90908560305	a ₂₀ = 0,06719525952545029
a ₆ = 147 524,32529733446	a ₂₁ = - 0,06456880539114203
a ₇ = 19 592,37365185654	a ₂₂ = - 0,03491390325356834
a ₈ = - 15 602,788031860411	a ₂₃ = 0,029367255329547745
a ₉ = - 1 470,1084980565095	a ₂₄ = 0,005738053495589528
a ₁₀ = - 2 294,95887448787	a ₂₅ = 0.0044160290145154706
a ₁₁ = - 544,1607383950874	a ₂₆ = 0,0017053091019810175
a ₁₂ = 448,9958454537547	a ₂₇ = - 0,001300387803579742
a ₁₃ = 77,1508223099868	
a ₁₄ = 10,037735602427968	

Vizsgáljuk meg ennek a közelítésnek a pontosságát két jellemző adat, a szórás (vagy a geodéziában középhiba) és a hibák maximális értékei alapján!

σ (szórás)	Pozitív hiba maximum	Negatív hiba maximum
18 cm	72,2 cm	-81,16 cm

Geoidfelület közelítése



10. ábra Polinomos közelítés hibáinak hisztogramja

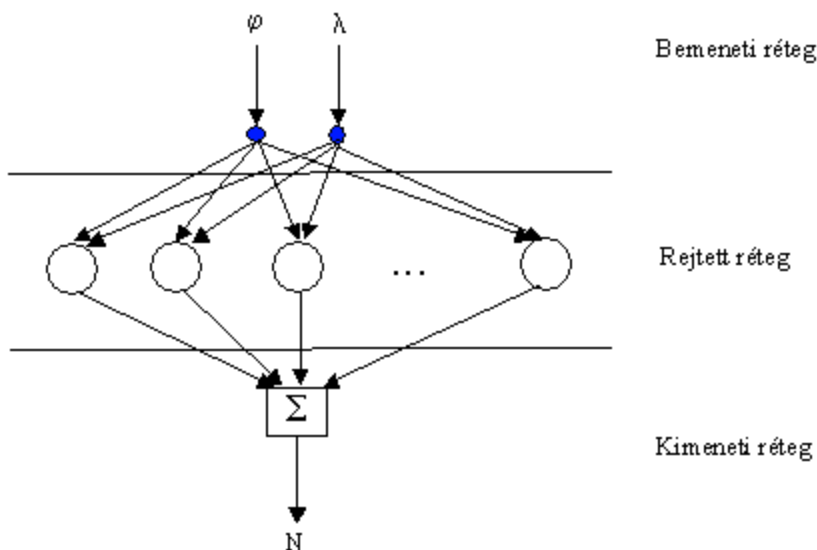
Ezek a hibák elég nagyra adódtak. Célszerű valamilyen más megoldást is kipróbálni, amivel esetleg csökkenteni lehet a hibákat. Egy lehetséges alternatíva lehet ebben az esetben a neurális hálózattal való közelítést.

3.3 Közelítés neurális hálózattal

3.3.1 Az alkalmazott hálózat

A geoidfelület közelítéséhez RBF (Radial Basis Function = radiális bázisú függvény) hálózatot alkalmaztam. Ennek a hálózatnak, mint arról korábban már szó volt, előnye a backpropagation hálózattal szemben a tanulás viszonylagos gyorsasága, a kevesebb iterációs lépés miatt. A gyakorlat azt mutatja, hogy ezekkel a hálózatokkal jól lehet szabálytalan felületek közelíteni, mint amilyen a geoidfelület is.

Az alkalmazott hálózat szerkezete a következő:



11. ábra A geoidfelület közelítésére alkalmazott hálózat

Két bemenő adatunk van: φ, λ , egy rejtett réteg RBF aktivációs függvényével és 35 neuronnal és egy neuron a kimeneti rétegben, ami lineáris összegzést valósít meg. A hálózat kimeneti jele a geoidunduláció értéke (N).

A Mathematica szoftverben az alkalmazott RBF függvény alakja a következő:

$$f(x) = e^{-\lambda^2(x-c)^2},$$

ahol λ a függvény szélességparamétere és c a függvény középpontja.

Mivel jelen esetben egy kétváltozós függvényt szeretnénk közelíteni, így két bemenő adatunk van.

A kétváltozós (alap)függvény alakja a következő:

$$f(x_1, x_2) = e^{-\lambda^2 \cdot [(x_1 - c_1)^2 + (x_2 - c_2)^2]}$$

A hálózat tanításához a ~200000 pontból ~8000 pontot használtam fel egyenletesen elosztva a vizsgált tartományon. Ez a 8000 pont megfelelően reprezentálja a 200000 pontos hálózatot, mint azt a későbbiekben látni fogjuk a tesztelés során. Összesen 35 darab neuront alkalmaztam. Minél több neuront alkalmazunk, annál bonyolultabb felületeket tudunk leírni. Azonban nem érdemes a neuron számot túlságosan megnövelni, mert ez túl bonyolult hálózatot eredményez. Különösen jelen esetben, amikor a nagyon sok tanító adat miatt a tanulási folyamat egy nagyon lassú iterációs eljárás. A tanulás időszükséglete a neuronszám növelésével többszöröse nő, ugyanakkor a rendszer pontossága egy idő után már nem nő tovább arányosan, és a tanítás túl lassúvá, gazdaságtalanná válik. Az ideális neuronszámot próbálgatással lehet meghatározni az egyes hálózatokhoz, figyelembe véve a pontossági követelményeket és a rendelkezésre álló időt, számítógép kapacitást. Próbálkoztam a tanítópontok számának a növelésével is, de ez is nagy mértékben növeli a számítási idő szükségletét, ami így is meglehetősen hosszú volt.

Másik fontos jellegzetessége ennek a közelítő módszernek, hogy nem egy neurális hálózattal közelíti a felületet, hanem neurális hálózatok sorozatával.

Ezt a módszert Paláncz Béla javasolta, és én ezt a modellt teszteltem le és igyekeztem tovább pontosítani. Abban a korábbi kutatásban, amit ő és Völgyesi Lajos végeztek, a már korábban említett 211680 rácsháló pontot tartalmazó adatbázisból a területen egyenletes eloszlásban kiválasztott kb. 8000 ponttal dolgoztak (Paláncz, Völgyesi 2002). Ez azt jelenti, hogy minden huszonötödik pont került bele a tanítópontok közé. Az eredeti adatok a $45^\circ 30' \leq \varphi \leq 49^\circ$, $16^\circ \leq \lambda < 23^\circ$ nagyságú területre voltak megadva $\Delta\varphi=0'30'' \times \Delta\lambda=0'50''$ felbontással, a kiválasztott 8484 pont ugyanezen a területen helyezkedik el $\Delta\varphi=2'30'' \times \Delta\lambda=4'10''$ felbontással.

Ezekkel a pontokkal készült el a neurális hálózat tanítása, majd a szimbolikus kiértékeléssel a leképzési függvényt ki is fejezték. A kész függvény a 35 neuronnak

Geoidfelület közelítése

megfelelően $35 (w \cdot e^{-\lambda^2 \cdot [(x_1 - c_1)^2 + (x_2 - c_2)^2]})$ alakú tagból áll, ehhez jön még egy lineáris tag $(a \cdot x + b \cdot y)$ és egy konstans eltolásérték, a bias.

Mintaként álljon itt az egyik alapfüggvény, ahol a bemenő adatok x_1 és x_2 voltak:

$$f(x_1, x_2) = 5.22702 \cdot 10^6 \cdot e^{-0.753045 \cdot (-47.7342 + x_1)^2 - 0.0753045 \cdot (-21.925 + x_2)^2}$$

Maga a tanítási folyamat ilyen nagyszámú (kb. 8000) pontnál igen hosszadalmas feladat, ami igen sok órát, esetleg több napot is igénybe vehet, a számítógép kiépítettségének (memória, processzor) függvényében. Az iterációk számát megadhatjuk mi is, illetve egy bizonyos minimumhatárt elérve a program magától leáll és befejezi a tanítást. Célszerű megadni egy iterációs számot, mivel nem tudhatjuk előre, hogy a gép magától hány iteráció után állna le. Persze ilyenkor is lehet az iterációt tovább folytatni, ha még nem értük el a hibák minimumát, de legalább nem szükséges egyszerre az egész tanítást végrehajtani. Az általam készített hálózatnál ezt az iterációs határt 200-ra állítottam és ezt a 200 iterációt a gép 3 óra alatt hajtotta végre (a gép adatai: 1 GHz processzor, 256 Mb RAM, ami jelenleg nem számít lassú gépnek). A hivatkozásban szereplő hálózatnál összesen kb. 920 iterációra volt szükség a minimumhatár eléréséhez, az általam továbbfejlesztett hálózatnál pedig kb. 350-re. Ezekből az adatokból is látszik, hogy ez az egész tanulási folyamat mennyire időigényes, viszont, ha már egyszer kész a hálózat azaz a leképzési függvényt előállítottuk, akkor ennek a kiértékelése már rendkívül gyors.

A közelítés javításának persze lehetnek más módjai is, nemcsak a neuronszám vagy a tanítópontok számának a növelése. Példa erre az alkalmazott neurális hálózat sorozat. Ez azt jelenti, hogy miután elkészítettük az első hálózatot a 8000 pont alapján, ki lehet számítani ugyanezekre a pontokra a hibákat, levonva a tényleges értékekből a leképzési függvényel előállított értékeket. Ha következő lépésként megpróbáljuk megtanítani a rendszert ezekre a hibákra, és utána hozzáadjuk az első neurális hálózat hibás értékeihez a hibákat, akkor elvileg a helyes értékeket kapjuk. Persze ez a gyakorlatban nem így van, mivel az új, a hibákat tanuló hálózat sem lesz száz százalékig tökéletes, lesznek ennek is hibái, amikre persze újabb hálózatokat készíthetünk. Ennek a sorozatnak a határértéke a tényleges kimeneti érték. Persze annak is megvan a határa, hogy meddig érdemes elmenni ilyen hálózat sorozat kialakításánál, mivel egyre hosszabb és bonyolultabb függvényeket fogunk kapni, összeadva a sok neurális hálózat leképzési függvényét. Mivel célunk elsősorban az, hogy csökkentjük a felhasznált adatok mennyiségét, hogy ne kelljen mindig a több mint 200000 adatot használnunk, ez megszabja azt is, hogy lehetőleg a függvényünk se legyen ilyen bonyolult, mivel akkor semmivel sem jutottunk előbbre, mintha az interpolációs eljárást választottuk volna.

A korábbi kutatások során egy négy tagból álló neurális hálózat sorozatot alkalmaztak, ahol látszik, hogy a hibák nagysága jelentősen csökkent az első és a negyedik hálózat után. Jellemző adatai e kétféle hálózatnak a szórása és a hiba maximális értéke, mint láthatjuk mind a két adat jelentősen lecsökkent a 4. fokú hálózatnál:

	σ (szórás)	Maximális hiba
1. fok	9.77 cm	58.54 cm

4. fok 6.58 cm 36.23 cm

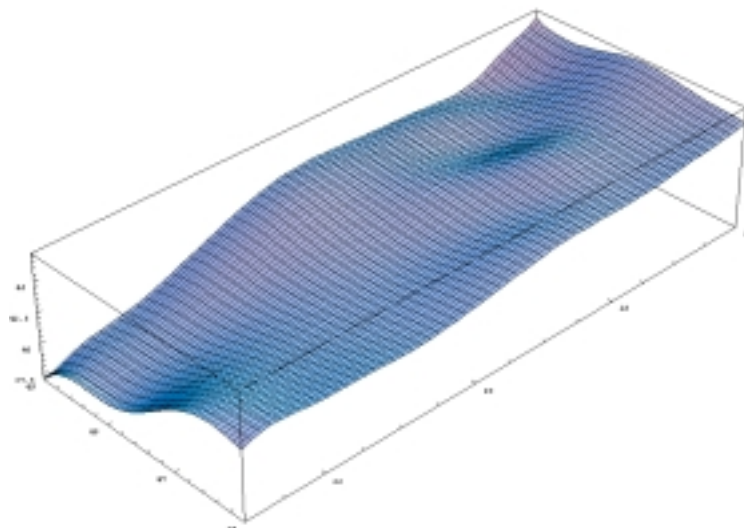
Ha azonban figyelembe vesszük, hogy maga az adatbázis $\pm 3-4$ cm pontosságú, akkor nekünk is ezen a pontosságon belül kellene maradnunk, és szükséges még tovább csökkenteni a középhibákat, a szórást. Ez az egyik megoldandó feladat, amivel én tovább próbálkoztam, illetve teszteltem az elkészült hálózatokat a összes (211680) pontra, és megvizsgáltam, hogy ez a kiválasztott 8000 pont megfelelően reprezentálja-e az eredeti kétszázézes adathalmazt.

3.3.2 A tesztelés eredménye

A hálózat pontosságát mutató adatok közül a legfontosabbak: a maradék hibák szórása, maximális értékei (pozitív, negatív maximumok), és a várható értéke. Ezeknek az alakulását ismertetem a különböző tesztelések során.

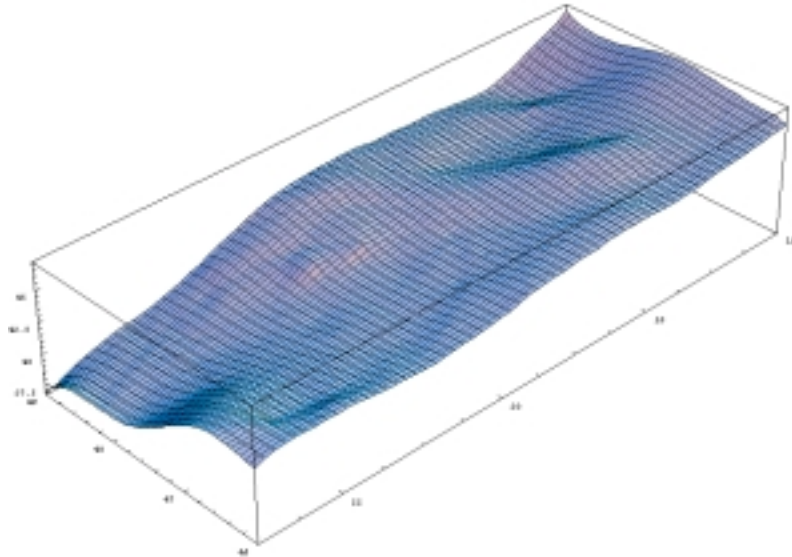
Az eredeti 211680 pontból a tanításhoz összesen 8484 pont került felhasználásra. A következőkben látni fogjuk, hogyan alakultak a fent említett értékek az elsőrendű neurális hálózat tanítása után, ill. a négy tagból álló neurális hálózat sorozat alkalmazása után. Az előbbi eredményeként kapott függvényt f_1 -nek neveztem el, az utóbbit F -nek ($F=f_1+f_2+f_3+f_4$). Továbbá a későbbiekben alkalmazható javításokhoz ábrázolom a fenti függvényeket, illetve a hibákat.

Ha az elsőrendű ill. a negyedrendű neurális hálózatok leképzési függvényét ábrázoljuk a tanítópontokra, akkor a következő felületeket kapjuk:



12. ábra A geoidfelület közelítése 1. rendű neurális hálózattal

Geoidfelület közelítése



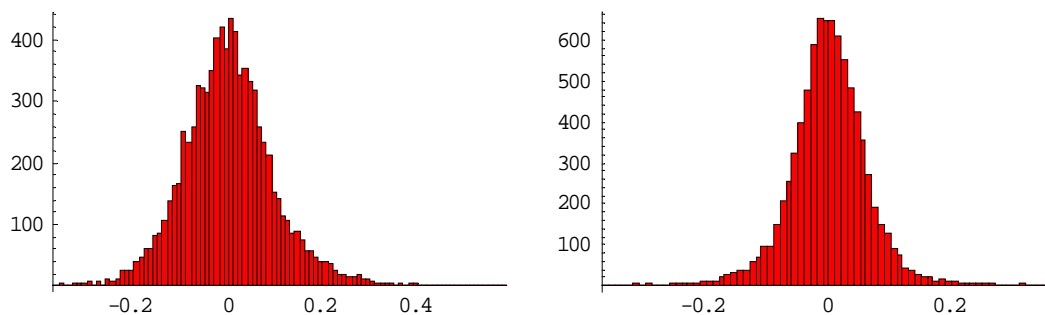
13. ábra A geoidfelület közelítése 4. rendű neurális hálózattal

A 12., 13. ábrából is látszik, hogy a negyedrendű neurális hálózattal egy sokkal bonyolultabb felületet lehet ábrázolni, ami sokkal közelebb áll a tényleges geoidfelülethez.

Ha elvégezzük a **tesztelést** a **8484 tanítópontra**, akkor a következő eredményeket kapjuk:

	σ (szórás)	Pozitív hiba <i>maximum</i>	Negatív hiba <i>maximum</i>	A hiba várható értéke
1. fok	9.77 cm	58.54 cm	-36.66 cm	0.00 cm
4. fok	6.58 cm	36.23 cm	-36.74 cm	0,00 cm

Az eltérések hisztogramjai a következők:



14. ábra 1. és 4. rendű hálózatok hibáinak hisztogramja

A **tesztelést** elvégezve a **211680 pontra** a következő eredmények adódnak:

Geoidfelület közelítése

	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>	<i>A hiba várható értéke</i>
<i>1. fok</i>	9.87 cm	60.03 cm	-41.61 cm	0.01 cm
<i>4. fok</i>	6.76 cm	43.33 cm	-50.62 cm	0,02 cm

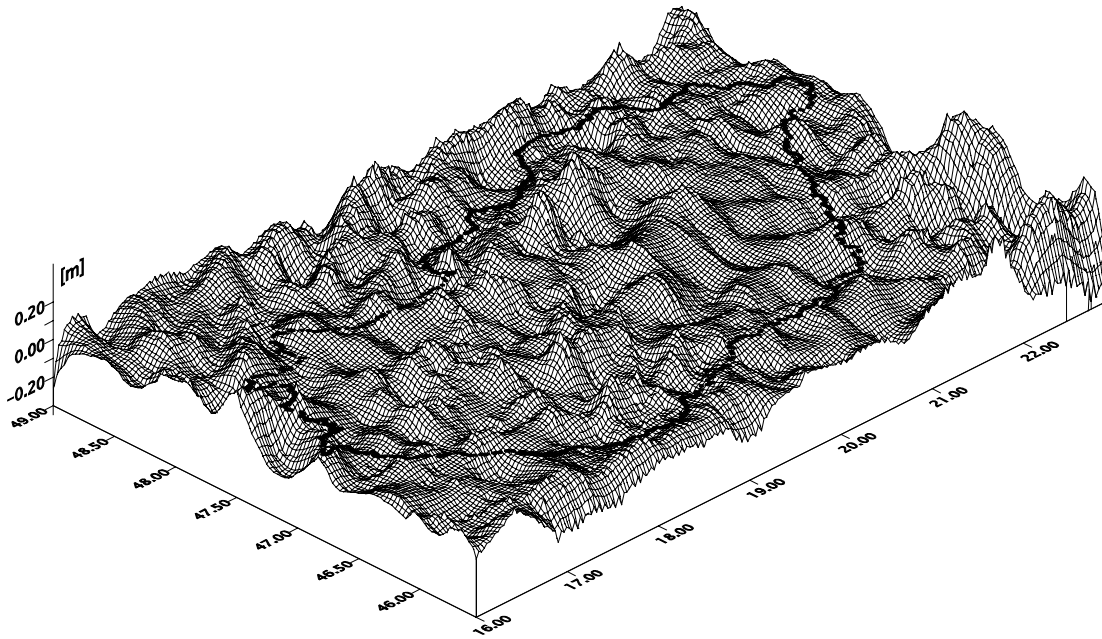
A két adatsor összehasonlításából látszik, hogy a szórás közel változatlan maradt a 8000 és a 200000 pontra végzett teszteléskor is. A várható érték pedig mindkét esetben nullának tekinthető. Tehát úgy tűnik, hogy a kiválasztott 8000 pontos mintaállomány megfelelően reprezentálja az összes adatot. Persze a maximális értékek nagyobbak lettek a teljes állományra végzett teszteléskor, de magának az eloszlásnak, a haranggörbének az alakja változatlan maradt.

Persze ezek az eredmények még igen távol állnak pontosságban a kívánatos 3-4 cm-es értéktől. Ezért ki kellett találni valamit, amivel tovább lehetne csökkenteni a közelítés hibáját.

Először felmerül, hogy el kellene végezni a tanítást egy nagyobb adathalmazra, nem csak 8000 pontra. Így a javítás érdekében először ezzel próbálkoztam. Nem minden 25., hanem minden 4. pontot vettem bele a tanítópontok közé, azaz összesen 52920 pontot. Azonban ez a módszer nem vezetett eredményre a futási idők nagyságrendekkel történő megnövekedése miatt. A 200 iterációt a számítógép a 8000 pontra 3 óra alatt hajtotta végre, ugyanez a 200 iteráció az 50000 pontra már legalább egy hétig tartott volna, és persze ez valószínűleg nem is lett volna elég, hanem jóval több iterációt kellett volna alkalmazni. Így kénytelen voltam ezt a kutatási irányt abbahagyni.

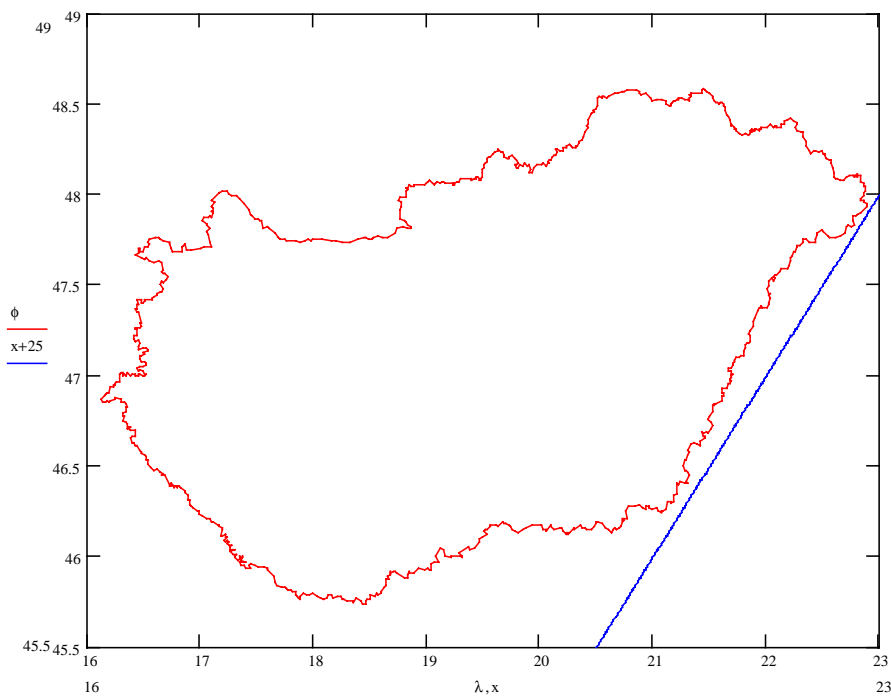
Egy másik javítási lehetőség kínálkozik, ha megfigyeljük a nagy hibák eloszlását.

Geoidfelület közelítése



15. ábra A 4. rendű neurális hálózattal közelített felület eltérései a geoidfelülettől

Az 15. ábrán észrevehetjük, hogy a legnagyobb hibák, nem is Magyarország területére esnek, hanem a romániai részen találhatóak. Ezt a részt egy egyenessel le lehet vágni. Meghatároztam annak az egyenesnek az egyenletét, amely a levágandó adatokra vonatkozó feltételt reprezentálja.



16. ábra A romániai nagy eltérésű területet levágó egyenes

Geoidfelület közelítése

Ennek az egyenesnek az egyenlete igen egyszerű lett: $\varphi = \lambda + 25$. Ezután a Mathematica programmal kiválogattam azokat az adatokat, amelyek e fölött az egyenes felett találhatóak, és a továbbiakban csak ezekkel dolgoztam.

Kiválogatás után a 8484 pontos állományból 7438 pont maradt, a 211680 pontos állományból pedig 184910. Ez után elvégeztem ezekre az állományokra is a tesztelést, hogy megvizsgáljam, mekkora mértékben javulnak ettől az eredmények, illetve, hogy egyáltalán javulnak-e.

Most a tesztelést már csak az F függvénnyel (a negyedrendű hálózattal) végeztem el. Összehasonlításképpen megadtam a korábbi, levágás előtti hálózatok eredményeit is.

Tesztelés 8484 ill. 7438 pontra

		σ (szórás)	Pozitív hiba maximum	Negatív hiba maximum	A hiba várható értéke
8484 pont	4. fok	6.58 cm	36.23 cm	-36.74 cm	0,00 cm
7438 pont	4. fok	5.63 cm	26.80 cm	-25.67 cm	0,01 cm

Tesztelés 211680 ill. 184910 pontra

		σ (szórás)	Pozitív hiba maximum	Negatív hiba maximum	A hiba várható értéke
211680 pont	4. fok	6.76 cm	43.33 cm	-50.62 cm	0,02 cm
184910 pont	4. fok	5.68 cm	29.14 cm	-35.13 cm	0,00 cm

Ha a fenti eredményeket összevetjük a korábbi 4. fokú eredményekkel, akkor láthatjuk, hogy tényleg igen jelentős mértékben javultak. A hibák maximális értékei például 10-15 cm-rel lettek kisebbek, és a sok nagyon nagy hibájú adat levágásának eredményeképpen a szórás is kb. 1 cm-rel kevesebb lett.

3.3.3 Új neurális hálózat

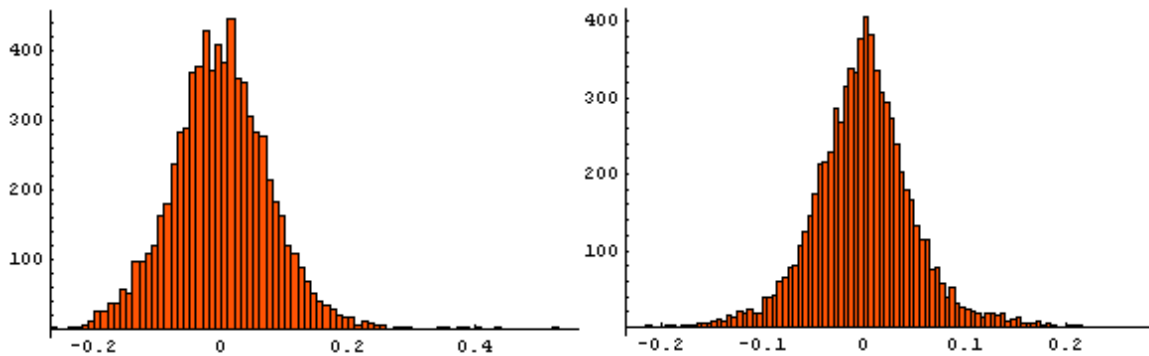
A korábbi vizsgálatokból megállapítható tehát, hogy a legnagyobb hibák nem is Magyarország területére esnek, hanem Romániában találhatóak. Érdemes lenne készíteni egy új neurális hálózatot, annak az adatállománynak a felhasználásával, amely nem tartalmazza az ezeken a területeken levő adatokat, és így valószínűleg jobb eredményeket kaphatnánk.

Hogy ezt a feltevést kipróbáljam újra elvégeztem a 4. rendű neurális hálózat tanítását, ezekkel a leválogatott adatokkal.

Az új f1, f2, f3, f4 függvények megtalálhatók a mellékletekben (M1,M2,M3,M4).

Geoidfelület közelítése

Az új neurális hálózatok hisztogramjai a következők:



17. ábra Az új 1. és 4. rendű hálózatok hibáinak hisztogramja

Az **új hálózat** eredményei a következők lettek, a 7438 tanítópontra vonatkoztatva:

	σ (szórás)	Pozitív hiba maximum	Negatív hiba maximum	A hiba várható értéke
1. fok	7.96 cm	55.12 cm	-26.21 cm	0.00 cm
4. fok	4.97 cm	28.37 cm	-23.36 cm	0,00 cm

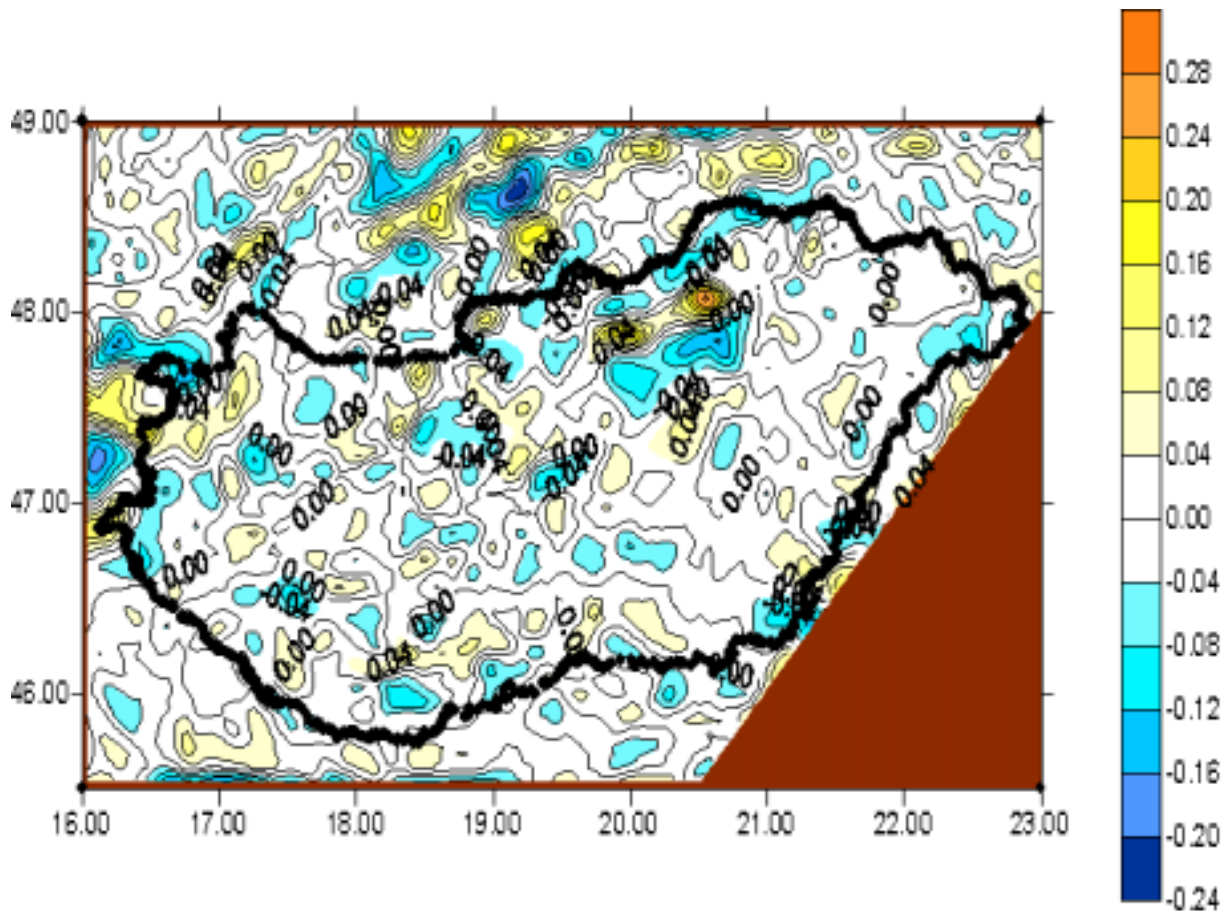
A könnyebb összehasonlításhoz összefoglalom a **korábbi** 8484 tanítópontra készített hálózat 7438 pontra tesztelt eredményeit.

	σ (szórás)	Pozitív hiba maximum	Negatív hiba maximum	A hiba várható értéke
4. fok	5.63 cm	26.80 cm	-25.67 cm	0,01 cm

Itt is némi javulás figyelhető meg a szórásban a korábbi hálózathoz képest (5.63 cm-ről lecsökkentek 4.97 cm-re), a hibák maximális értékei viszont maradtak körülbelül ugyanazok. Ez a jelenség a hálózat robusztusságára utal, azaz néhány „rossz” adat a tanulóhalmazban csak csekély mértékben befolyásolja a tanulás minőségét.

3.4 Eredmények, további lehetőségek

Surfer program segítségével ábrázoltam az eltéréseket, az eredeti (HGTUB2000) geoidmodell adataiból levonva az új neurális hálózattal kapott eredményeket.



18. ábra Eltérések az eredeti (HGTUB2000) geoid magasságok és a neurális hálózattal közelített magasságok között

A 18. ábrán már csak a leválogatott adatokat ábrázoltam, kihagyva az általam levágott romániai részt. Ha megnézzük ezt az ábrát, látható, hogy a nagy hibák egy jó része ismét csak az országhatáron kívülre esik, illetve az is feltűnő, hogy nagyobb hibákat elsősorban a hegyvidékeken találunk, Magyarországon a Mátra, Bükk és az Alpokalja területén.

Az elérni kívánt pontosság 3-4 cm volt. Az ábrán fehérrel jelöltem a 4 cm alatti eltéréseket. Ugyan ezt nem sikerült mindenütt elérni, de az ország jelentős részén már igen. Ebből az a következtetés vonható le, hogy érdemes még ezzel a témával a későbbiekben is foglalkozni, és tovább pontosítani a közelítést, mivel az eddigi eredmények biztatóak.

A fentiek alapján igen valószínű, hogy a magyarországi hegyvidéki területeken a tanítópontok sűrítése javíthat az eredményeken. Érdemes lenne kipróbálni ezt egy új neurális hálózat elkészítésével.

3.5 A polinomos közelítés összehasonlítása a neurális hálózatokkal

Hasonlítsuk most össze a kétféle módszerrel történő közelítést.

Először a hagyományos polinomos regressziót végeztem el. Itt szembesülnöm kellett azzal a ténnyel, hogy hiába a nagy mennyiségű adat, a polinommal való közelítésnek (akárcsak persze a neurális hálózatoknak) korlátai vannak.

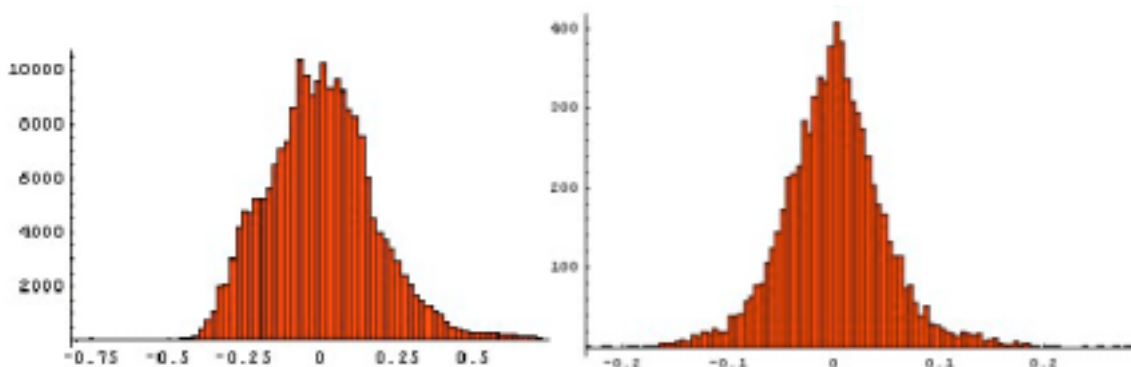
A polinom esetében a fokszám növekedésével a rosszul kondicionáltság miatt a regressziós feladathoz tartozó pszeudinverz előállítására nehézségekbe ütközik.

Több mint 200 000 adat állt rendelkezésemre a felület közelítéséhez, de csak hatodfokú polinomot tudtam rájuk illeszteni. Ennek az illesztésnek a hibái a következők lettek:

<i>Polinomos regresszió</i>		
σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
18 cm	72,2 cm	-81,16 cm

Ugyanezek az adatok a neurális hálózat sorozattal való közelítésnél a következők lettek:

<i>Regresszió neurális hálózatokkal</i>		
σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
4,97 cm	28,37 cm	-23,36 cm



19. ábra Polinomos (a) és neurális hálózatokkal való közelítés (b) hibáinak a hisztogramja

A fenti hisztogramokból és adatokból jól látszik, hogy a neurális hálózatokkal elért pontosság messze meghaladja a hagyományos polinomos közelítés pontosságát. A neurális hálózatok a későbbiekben még fontos szerepet játszhatnak mindenféle digitális domborzatmodellezési feladat megoldása során.

4 EOV – WGS-84 koordináta-transzformáció

4.1 Bevezetés

Magyarországon többféle vetületi rendszert használnak egyidejűleg. Ezért a geodéziában igen fontos feladat a koordináta-transzformáció, melynek sokféle módszere alakult ki az idők folyamán.

Szabatos vetületi átszámítás (zárt matematikai összefüggések felhasználásával) két vetületi rendszer között csak akkor végezhető, ha a két vetület alapfelülete közös és ugyanannak a háromszögelési hálózatnak a pontjai vannak ábrázolva a két vetületen. Ha ugyanis eltérő háromszögelési hálózathoz tartozó pontokat szeretnénk átszámítani, akkor ezek a pontok nem fognak illeszkedni a másik vetületi síkon ábrázolt háromszögelési hálózat pontjai közé. Ennek oka a két háromszögelési hálózat különböző elhelyezésében, tájékozásában, külön alapvonal rendszerükben, az egymástól független szögmérésekből adódó különbségekben és a különböző kiegyenlítésekben keresendő.

Ebben az esetben, ha a két vetületen nem ugyanannak az alaphálózatnak a pontjai vannak ábrázolva ill., ha eltérők az alapfelületek, akkor az átszámítás azonos pontok segítségével történhet, melyeknek mindkét rendszerben ismertek a koordinátáit.

Az ilyen koordináta-transzformációknak több típusa van. Lehetnek kétdimenziós síkbeli vagy háromdimenziós térbeli transzformációk. Típus szerint lehet hasonlósági vagy Helmert transzformáció, ezenkívül affin és kétdimenziós polinomos transzformáció is. A szerint is szokták csoportosítani őket, hogy hány meghatározandó paraméter szerepel bennük, így a térbeli Helmert transzformációt hétparaméteres transzformációnak is nevezik.

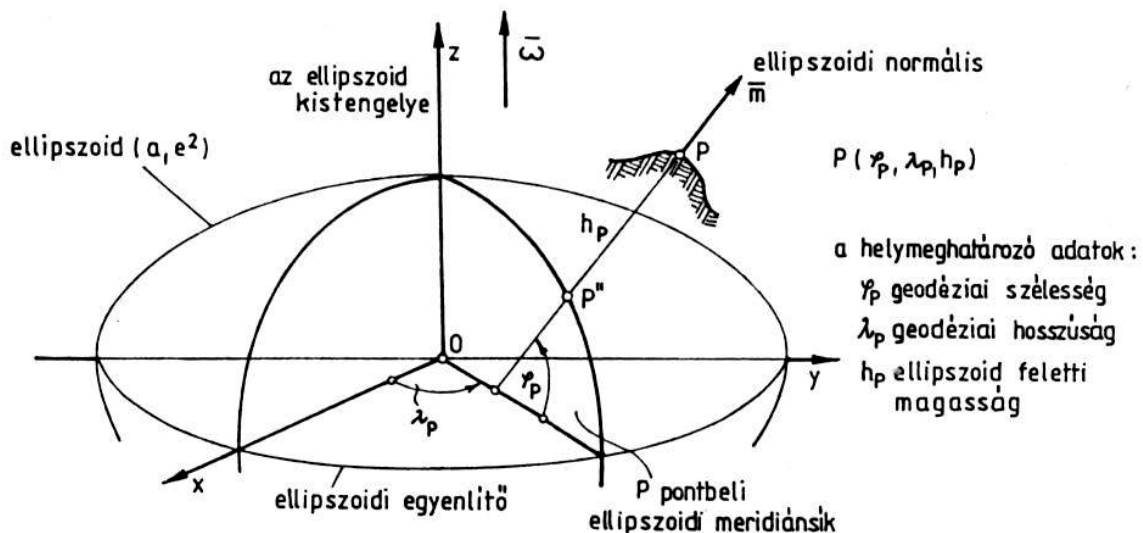
4.2 A feladat ismertetése

Ma Magyarországon a két leggyakrabban használt koordináta rendszer a GPS mérések koordináta rendszere, a WGS-84 ellipszoidi koordináta rendszer, és az Egységes Országos Vetületi rendszer (EOV), melynek alapfelülete az IUGG/1967 ellipszoidhoz simuló új magyarországi Gauss-gömb.

Ezen két rendszer közötti koordináta transzformációs egyenletek meghatározásának lehetőségét vizsgáltam általános sorokkal (polinomokkal) és

neurális hálózatokkal a rendelkezésemre álló 43 azonos pont alapján, mivel a fenti okok miatt a két rendszer között nem végezhető szabatos vetületi átszámítás.

A WGS-84 (World Geodetic System 1984) egy nemzetközileg használt globális vonatkozási rendszer. A rendszert meghatározza az alapfelületként választott a és b féltengely-hosszúságú ellipszoid. A koordináta rendszer z tengelye az ellipszoid kistengelye, egybeesik a Föld közepes forgástengelyével, x és y tengely pedig az egyenlítő síkjában helyezkedik el, x a greenwichi kezdő meridián síkjában, y pedig merőleges rá.



20. ábra Ellipszoidi földrajzi koordináta rendszer (Krauter, 1995)

Ellipszoidi koordináta rendszerben (így a WGS-84 ellipszoidi rendszerben is) egy pont helymeghatározó adatai a következők (a vetülettanban szokásos jelölésekkel):

- Φ_P ellipszoidi földrajzi (geodéziai) szélesség,
- Λ_P ellipszoidi földrajzi (geodéziai) hosszúság és
- h_P ellipszoid feletti magasság.

Jelen feladatnál csak a vízszintes helymeghatározó adatokkal számoltam, így a két felhasznált helymeghatározó adat Φ és Λ volt.

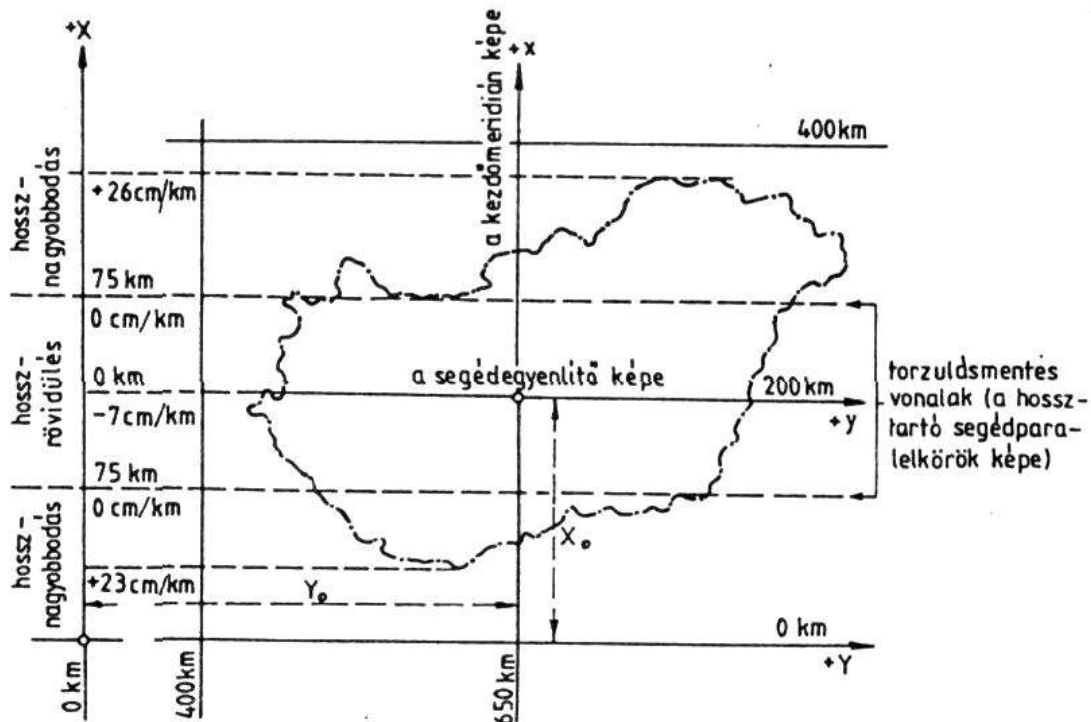
1975-ben vezették be Magyarországon az Egységes Országos Vetületi Rendszert (EOV), mely az új magyarországi Gauss-gömb ferdetengelyű süllyesztett (redukált) hengervetülete. A redukált vetítésre azért volt szükség, hogy az ország egész területe ábrázolható legyen egyetlen vetületi síkon, anélkül, hogy a hossztorzulás túllépne egy megengedett értéket. A vetítés két lépésben történik, először az ellipszoidról vetítenek gömbre, majd gömbről síkra.

A síkkoordináta rendszer x tengelye a Gellérthegyi kezdőmeridián képe, y tengelye pedig a segédegyenlítő képe. A koordináta rendszer északkeleti tájolású. Azért, hogy kiküszöböljék a negatív koordinátákat és meg tudják különböztetni ránézésre az x és y koordinátákat, eltolták a koordináta-rendszer kezdőpontját mindkét irányban. Így az x koordináták mindig kisebbek 400 km-nél, az y koordináták pedig nagyobbak 400 km-nél.

Az új koordinátaértékek a következőképp számíthatóak:

$$X_{EOV} = x + 200000.000 \text{ m,}$$

$$Y_{EOV} = y + 650000.000 \text{ m.}$$



21. ábra Egységes Országos Vetület (EOV) (Krauter, 1995)

A két átszámítandó rendszerben eltérő adatokat használunk, az Egységes Országos Vetületi Rendszerben x,y síkkoordinátákkal dolgozunk, a WGS-84 rendszerben pedig ellipszoidi koordinátákkal. Ezek között kell koordináta-transzformációs kapcsolatot felállítani.

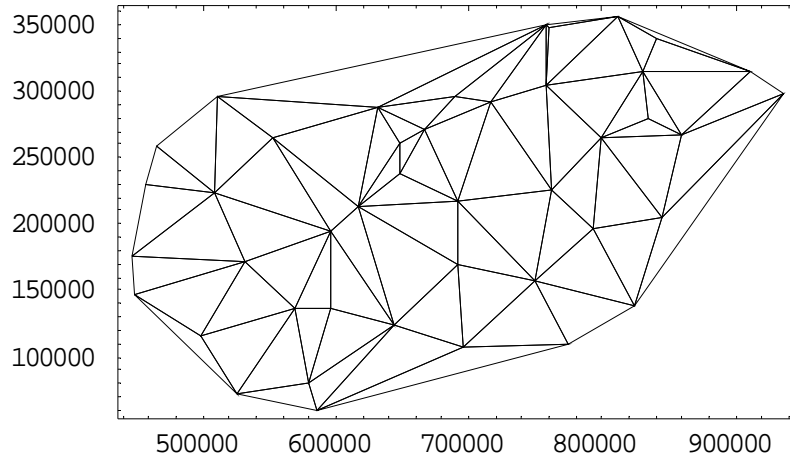
Adottak mindkét rendszerben az összetartozó pontpárok koordinátái (összesen 43 adatpár): (y,x) és (Φ,Λ). Feladatunk meghatározni az F és F⁻¹ függvényeket, ahol:

$$\begin{pmatrix} \Phi \\ \Lambda \end{pmatrix} = F(y, x) = \begin{pmatrix} F1(y, x) \\ F2(y, x) \end{pmatrix}$$

illetve

$$\begin{pmatrix} y \\ x \end{pmatrix} = F^{-1}(\varphi, \lambda) = \begin{pmatrix} FI(\Phi, \Lambda) \\ FI(\Phi, \Lambda) \end{pmatrix}$$

Az azonos pontok elhelyezkedése a következő:



22. ábra Azonos pontok az EOV rendszerben

4.3 A közelítés minőségének kérdése

A különböző módszerek összehasonlításánál felmerül a minőség kérdése, az hogy hogyan tudjuk eldönteni, hogy egy módszer jónak tekinthető e.

Azt szeretnénk elérni, hogy az előállított függvény ne csak a felhasznált azonos pontokban adjon jó közelítést, hanem az adott területen belül minden pontban. Ezt valahogyan ellenőrizni kell. Ennek egyik lehetséges módszere az, hogy a rendelkezésre álló adatok egy részét használjuk fel a függvények előállítására és a további adatokat pedig ellenőrzésre (mint a geoidfelület közelítésénél).

Jelen esetben erre nincsen lehetőségünk, mivel nagyon kevés, összesen 43 azonos pontunk van Magyarország egész területén. Ha jó közelítést akarunk elérni, akkor mindet bele kell vonnunk a kiegyenlítésbe.

A transzformációs függvények megfelelőségének eldöntésére két módszer áll rendelkezésünkre.

Az egyik természetesen az, hogy a függvény a felhasznált azonos pontok koordinátáit milyen pontossággal állítja elő az egyik rendszerből a másikba történő transzformáció során. Ez persze arra még nem jelent biztosítékot arra, hogy a transzformáció az azonos pontok között is jól működik.

Tesztpontok híján egy lehetőségünk van ennek a vizsgálatára. Ha mind a két irányban jól működik a transzformáció, akkor az oda-vissza transzformációnak is jól

kell közelítenie a kiindulási adatokat. Ezt úgy próbálhatjuk ki, hogy tetszőlegesen felvesszünk pontokat az egyik koordináta rendszerben, a transzformációs egyenletekkel átszámítjuk a másik koordináta rendszerbe, majd elvégezzük ezekkel a pontokkal a visszafelé transzformációt is. Ezek után összehasonlítjuk a kiindulási adatokkal, amelyeket megfelelő pontossággal kell közelíteniük. Ez persze csak a transzformáció megfelelőségének a szükséges, de nem elégséges feltétele, hiszen lehet hogy mindkét irányban hibás a transzformáció, de oda-vissza esetben a hibák kiejtik egymást. Ha viszont az oda-vissza transzformáció nagyon nagy hibákat eredményezett, akkor biztos, hogy nem működik megfelelően egyik irányban sem az átszámítás.

Én ezen két ellenőrzési lehetőséget fogom a következőkben elvégezni minden módszer vizsgálatánál.

4.4 Általános sorokkal történő transzformáció

A Vetületi Szabályzat szerint a transzformáció történhet általános sorokkal (így nevezi a szabályzat a hatványpolinomokat), de ezek a sorok legfeljebb ötödfokúak lehetnek.

Egy 5. fokú polinom a következő:

$$x' = A_0 + A_1 \cdot x + A_2 \cdot y + A_3 \cdot x^2 + A_4 \cdot x \cdot y + A_5 \cdot y^2 + A_6 \cdot x^3 + A_7 \cdot x^2 \cdot y + A_8 \cdot x \cdot y^2 + A_9 \cdot y^3 + A_{10} \cdot x^4 + A_{11} \cdot x^3 \cdot y + A_{12} \cdot x^2 \cdot y^2 + A_{13} \cdot x \cdot y^3 + A_{14} \cdot y^4 + A_{15} \cdot x^5 + A_{16} \cdot x^4 \cdot y + A_{17} \cdot x^3 \cdot y^2 + A_{18} \cdot x^2 \cdot y^3 + A_{19} \cdot x \cdot y^4 + A_{20} \cdot \lambda^{y^5}$$

$$y' = B_0 + B_1 \cdot x + B_2 \cdot y + B_3 \cdot x^2 + B_4 \cdot x \cdot y + B_5 \cdot y^2 + B_6 \cdot x^3 + B_7 \cdot x^2 \cdot y + B_8 \cdot x \cdot y^2 + B_9 \cdot y^3 + B_{10} \cdot x^4 + B_{11} \cdot x^3 \cdot y + B_{12} \cdot x^2 \cdot y^2 + B_{13} \cdot x \cdot y^3 + B_{14} \cdot y^4 + B_{15} \cdot x^5 + B_{16} \cdot x^4 \cdot y + B_{17} \cdot x^3 \cdot y^2 + B_{18} \cdot x^2 \cdot y^3 + B_{19} \cdot x \cdot y^4 + B_{20} \cdot \lambda^{y^5}$$

ahol x, y az egyik rendszerbeli koordinátákat jelöli és x', y' pedig a másik rendszerbeli koordinátákat.

A szükséges azonos pontok száma a kétváltozós polinom fokszámától függ:

$$p = \frac{(r+2) \cdot (r+1)}{2},$$

ahol r a polinom fokszáma, p a szükséges azonos pontok száma.

Ötödfokú polinom esetében például 21 azonos pont szükséges. Ha ennél több pont áll rendelkezésünkre, akkor lehetőség van kiegyenlítésre.

Kiegyenlítés esetén a transzformáció eredményeképpen nem pontosan a másik rendszerbeli koordinátákat kapjuk vissza, hanem lesznek ún. maradék

ellentmondások. Ezek az eltérések nem haladhatják meg az érvényes utasításban előírtakat.

4.4.1 Általános sorokkal történő transzformáció segédvetületi rendszer (SVR) közbeiktatásával

A BME Általános és Felsőgeodézia Tanszéken készült egy programcsomag vetületi átszámításokhoz (Völgyesi, Tóth, Varga 2001) VETULET+ néven. Ez több programot tartalmaz. Külön programmal lehet meghatározni a transzformációs polinom együtthatóit az azonos pontok alapján és egy másik program végzi az átszámítást a meghatározott paraméterek alapján.

Ebben a programcsomagban az Egységes Országos Vetületi Rendszer és a WGS-84 ellipszoidi koordináta rendszer közötti átszámítás több lépésben történik. A WGS-84 ellipszoidi Φ, Λ koordinátákat először az új Gauss-gömbre számítják át, majd az új Gauss-gömből egy segédvetületi síkra vetítik (SVR). Végül ezeket a segédvetületi síkkoordinátákat transzformálják át ötödfokú hatványpolinomokkal EOV koordinátákká. EOV-ból WGS-84-be a transzformáció fordított sorrendben történik. Az oda és vissza transzformációhoz összesen 4 polinom szükséges.

4.4.1.1 Átszámítás WGS-84-ből EOV-ba (transzformációs polinom együtthatóinak a meghatározása)

Ha a WGS-84-es Φ, Λ értékek fok-perc-masodpercben adottak, akkor először tizedfokba kell ezeket átszámítani, majd radiánba.

Ezután lehet az ellipszoidi Φ, Λ koordinátákból Gauss-gömbi φ, λ koordinátákat számítani a következő képletekkel:

$$\varphi = 2 \cdot \left\{ \operatorname{arctg} \left[k \cdot \operatorname{tg}^n \left(\frac{\pi}{4} + \frac{\Phi}{2} \right) \cdot \left(\frac{1 - e \cdot \sin \Phi}{1 + e \cdot \sin \Phi} \right)^{\frac{n \cdot e}{2}} \right] - \frac{\pi}{4} \right\}$$

$$\lambda = n \cdot (\Lambda - \Lambda_0),$$

ahol:

$$\Lambda_0 = 0.33246029531$$

$$e = 0.0818205679407$$

$$k = 1.003110007693$$

$$n = 1.000719704936$$

A Gauss-gömbi koordinátákból a segédvetületi rendszer síkkordinátáira való vetítés felírása gömbi segédkoordináták segítségével történhet.

A gömbi koordinátákból gömbi segédkoordináták számítása:

$$\varphi' = \arcsin(\cos \varphi_0 \cdot \sin \varphi - \sin \varphi_0 \cdot \cos \varphi \cdot \cos \lambda)$$

$$\lambda' = \arcsin\left(\frac{\cos \varphi \cdot \sin \varphi}{\cos \varphi'}\right),$$

ahol

$$\varphi_0 = 0.822050077$$

A gömbi segédkoordinátákból a segédvetületi síkkoordináták (x_t, y_t) számítása:

$$x_t = R \cdot m_0 \cdot \ln \operatorname{tg}\left(\frac{\pi}{4} + \frac{\varphi'}{2}\right) + 200000$$

$$y_t = R \cdot m_0 \cdot \lambda' + 650000,$$

ahol

$$R = 6379743,001$$

$$m_0 = 0,99993$$

A program ezek után kiszámítja a segédvetületi síkkoordináták és az EOV rendszer koordinátái közötti 5. fokú polinom transzformációs együtthatóit (lásd 4.4 fejezet első része)

Így ez egy olyan eljárás, melynek egyéb számítások mellett része az általános hatványpolinommal történő transzformáció.

4.4.1.2 EOV-ból WGS-84-be történő átszámítás (transzformációs polinom együtthatóinak a meghatározása)

Ez az előző számítással ellentétesen történik. Először EOV koordinátákat transzformálják át polinomok segítségével segédvetületi síkkoordinátákká, ehhez meghatározzák az inverz polinom együtthatóit.

x' és y' SVR koordináták átszámítása gömbi segédkoordinátákká:

$$x = x' - 200000; \quad y = y' - 650000$$

$$\varphi' = 2 \cdot \operatorname{arctg}\left(e^{\frac{x}{R \cdot m_0}}\right) - \frac{\pi}{2}$$

$$\lambda' = \frac{y}{R \cdot m_0},$$

majd ezek segítségével a gömbi koordináták számítása:

$$\varphi = \arcsin(\sin \varphi' \cdot \cos \varphi_0 + \sin \varphi_0 \cdot \cos \varphi' \cdot \cos \lambda')$$

$$\lambda = \arcsin\left(\frac{\cos \varphi' \cdot \sin \lambda'}{\cos \varphi}\right),$$

Az utolsó feladat az új Gauss-gömből áttérni a WGS-84 ellipszoidi Φ, Λ koordinátákra.

Az ellipszoidi hosszúság, Λ meghatározása egyszerű:

$$\Lambda = \Lambda_0 + \frac{\lambda}{n}$$

Az ellipszoidi hosszúság, Φ meghatározása viszont csak iterációval történhet. Az első iterációs lépés meghatározása:

$$\Phi_1 = 2 \cdot \arctg \left[\frac{\operatorname{tg} \left(\frac{\pi}{4} + \frac{\varphi}{2} \right)}{k} \right]^{\frac{1}{n}} - \frac{\pi}{2}$$

majd ennek segítségével:

$$\Phi_{i+1} = 2 \cdot \arctg \left[\frac{\operatorname{tg} \left(\frac{\pi}{4} + \frac{\Phi_i}{2} \right)}{k \cdot \left(\frac{1 - e \cdot \sin \Phi_i}{1 + e \cdot \sin \Phi_i} \right)^{\frac{n \cdot e}{2}}} \right]^{\frac{1}{n}} - \frac{\pi}{2}$$

Az iterációt addig kell folytatni, amíg a Φ_i és Φ_{i+1} közötti különbség a kívánt pontosságú lesz.

Az így kapott Φ, Λ értékeket át kell váltani radiánból fok-perc-másodperc értékre.

4.4.1.3 Eredmények értékelése

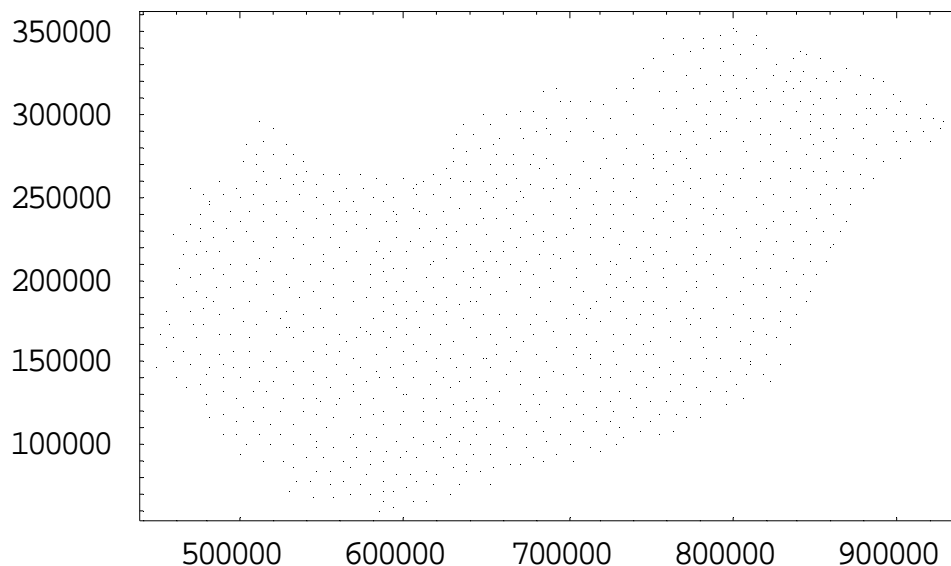
A 4.3 fejezetben ismertetett kétféle minősítő vizsgálatot végeztem el itt is. Először nézzük meg a transzformációk után kapott maradék eltérések mértékét, a legnagyobb, legkisebb értéket és a középhibát vagy szórást, az EOV koordinátáknál m-ben, a WGS-84-nél pedig másodpercben.

<i>Polinomos transzformáció segéd vetületi rendszer alkalmazásával</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0.036 m	0,058 m	-0.089 m
x	0.036 m	0,076 m	-0.086 m
Φ	0.0012"	0,0030"	-0.0022"
Λ	0,0017"	0,0040"	-0,0031"

Ezek az eltérések alacsonyok (3-4 cm a középhiba), a közelítés elég jó.

A másik ellenőrzési lehetőség az adott területen belül tetszőlegesen kiválasztott pontok oda-vissza transzformációja közötti eltérések vizsgálata.

Ehhez felvettem véletlenszerűen több mint 1000 EOV pontot és ezt transzformáltam át WGS-84-be, majd vissza a korábban meghatározott transzformációs képletekkel.



23. ábra A tesztpontok elhelyezkedése az oda-vissza transzformációhoz

Ennek az eltérései a következők lettek:

<i>Oda-vissza transzformáció eltérései SVR-rel</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0,007 m	0,012 m	-0.016 m
x	0,009 m	0,021 m	-0,019 m

Az eltérések itt még kisebbek, tehát valószínűleg ez a közelítés elfogadható, nem csak az azonos pontokban, hanem a pontok között is.

4.4.2 Általános sorokkal történő transzformáció segédvetületi rendszer nélkül

Az előző számítás meglehetősen nehézkes. Először a segédvetületi rendszerre kell átszámolni a koordinátákat bonyolult képletekkel, és csak utána határozzák meg a transzformációs polinom együtthatóit. Az EOV rendszerből WGS-84 rendszerbe történő átszámítást még körülményesebbé teszi, hogy itt iterációs lépéseket is kell alkalmazni.

Kíváncsi voltam milyen eredményeket lehet elérni, ha kihagyjuk a segédvetületi rendszert a számításból, és közvetlenül a WGS-84 ellipszoidi koordinátái és az EOV síkkoordinátái között határozzuk meg a transzformációs polinom együtthatóit. Arra is szerettem volna választ kapni, hogy lehet-e ötödfokúnál magasabb polinomot illeszteni a pontokra.

A meghatározandó egyenletek a következők:

$$\begin{pmatrix} \Phi \\ \Lambda \end{pmatrix} = F(y, x), \quad \begin{pmatrix} y \\ x \end{pmatrix} = F^{-1}(\varphi, \lambda)$$

A megoldás során kétféleképpen is eljárhatunk. Először kiegyenlítéssel meghatározunk 2 közelítő polinomot az egyik irányú transzformációra, és utána meghatározzuk az inverz függvény értékét, ezt egy nemlineáris egyenletrendszer numerikus megoldásával kapjuk. Vagy a másik irányú transzformációra is előállítunk 2 közelítő polinomot, függetlenül az elsőtől. Én ez utóbbi megoldást választottam.

A polinom illesztésének nehézségei vannak, az egyenletek rosszul kondicionáltsága miatt. Az illesztést Mathematica programmal végeztem el. Először egy hagyományos illesztési módszerrel próbálkoztam, a beépített Fit függvénnyel, amely általános lineáris regresszióra használható. Azt az eredményt kaptam, hogy másodfokú polinommal való közelítés után hirtelen nagy mértékben növekedtek a maradék elentmondások. Kénytelen voltam egy másik módszert a Regress függvényt alkalmazni. Ez jobb numerikus stabilitású, különösen magasabb fokszám esetén.

Így is csak ötödfokú polinomig tudtam elmenni az illesztésben az egyenletek rosszul kondicionáltsága miatt, holott a 43 azonos pontra elvileg 7. fokú polinom is illeszthető. Hetedfokú polinom illesztéséhez szükséges azonos pontok száma ugyanis:

$$p = \frac{(r+2) \cdot (r+1)}{2} = \frac{(7+2) \cdot (7+1)}{2} = 36$$

, szemben a rendelkezésre álló 43 ponttal.

Az alkalmazott ötödfokú polinomok képletei a mellékletekben megtalálhatók (M5). Nézzük meg a kapott eredményeket és hasonlítsuk össze a segédvetületi rendszer alkalmazása során kapottal.

<i>Polinomos transzformáció segéd vetületi rendszer nélkül</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0.035 m	0,058 m	-0.088 m
x	0.036 m	0,074 m	-0.082 m
Φ	0.0011"	0,0026"	-0.0024"
Λ	0,0018"	0,0044"	-0,0029"

Ha ezt összehasonlítjuk a segédvetületi rendszerrel történt transzformációval, akkor látszik, hogy ugyanolyan mértékű eltérések adódtak, a kettő közötti különbség minimális. Ha egyenként vizsgáljuk meg a 43 azonos pontra az eltéréseket, akkor a két módszer eredményei között maximum 1-2 mm eltérést találunk.

Most nézzük meg az oda-vissza transzformációt ugyanarra az 1000 pontra, mint korábban.

<i>Oda-vissza transzformáció eltérései SVR nélkül</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0,003 m	0,012 m	-0.011 m
x	0,002 m	0,012 m	-0,007 m

Az oda-vissza transzformáció eltérései még kisebbek lettek, mint az előző esetben. Az y koordináta esetében a középhiba lecsökkent 7 mm-ről 3-ra, x esetében pedig 9 mm-ről 2-re. Ez azt a feltételezést látszik alátámasztani, hogy a segédvetületi rendszer nyugodtan elhagyható, és így jelentős mértékben egyszerűsítenyi lehet a számításokat.

Kérdéses, hogy ennél pontosabb transzformáció lehetséges-e. Mivel a geoidfelület közelítésénél a neurális hálózatok sokkal jobb eredményt produkáltak, mint a hagyományos polinomos regresszió, úgy gondoltam érdemes lenne kipróbálni ezt a módszert koordináta transzformáció esetén is.

4.5 Transzformáció neurális hálózatokkal

4.5.1 Alkalmazott hálózatok

Először az alkalmazott hálózat felépítését kell meghatározni.

A feladatot megoldhatjuk 2 olyan hálózattal, ahol két bemenet és két kimenet van, illetve készíthetünk 4 hálózatot külön-külön mind a 4 koordinátára (y, x, Φ, Λ) két bemenettel és egy kimenettel.

Én ez utóbbit alkalmaztam. Egy kimenettel sokkal egyszerűbb a hálózatok felépítése, kevesebb neuron szükséges ugyanakkora pontossághoz. A hálózat egyszerűsége most különösen fontos, hiszen nagyon kevés adat áll rendelkezésünkre a tanításhoz. Egy másik oka, hogy ezt a szerkezetet választottam, az, hogy két bemenetet két kimenettel, akkor célszerű alkalmazni, amikor keresztkapcsolatok is vannak az adatok között, pl. ha y függne x -től, de jelen esetben ez nem áll fenn. Az alkalmazott hálózat egy bemeneti rétegből, egy rejtett rétegből és egy kimeneti rétegből állt.

Fontos kérdés az alkalmazott aktivációs függvény típusa és a neuronszám. Az előző feladatnál RBF hálózatot alkalmaztam. Ennek elsősorban az volt az oka, hogy ez a hálózat viszonylag gyorsan tanul, szemben a backpropagation hálózattal. 8000 tanulópontnál ez lényeges szempont, hiszen így is nagyon lassú a tanítás. A rendelkezésre álló 43 tanítópontnál viszont mind a két típusú hálózat tanítása viszonylag gyorsan elvégezhető, ezért mindkét hálózat alkalmazási lehetőségét megvizsgáltam.

Az alkalmazott neuronok számát itt is szisztematikus próbálgatással lehet meghatározni. A geoidfelületnél a neuronszám felső korlátját az szabta meg, hogy egy idő után már nem tudott tovább tanulni a hálózat, nem növekedett tovább a pontosság, csak a futási idő, a számítógép véges számábrázolásának következtében. Az EOV-WGS-84 koordináta-transzformációnál a rejtett rétegbeli csomópontok számának felső korlátját a túl kevés tanítópontra szabja meg.

Kétféle hálózatot is kipróbáltam. Az első egy RBF hálózat volt, a másik egy backpropagation hálózat szigmoid aktivációs függvényvel. Mindkét hálózatnál a kimeneti rétegben lineáris összegzést megvalósító neuront alkalmaztam.

A meghatározandó paraméterek száma (p) a rejtett rétegbeli neuronok számától függ. RBF hálózatnál ez a következőképpen alakul n neuronszám esetén:

$$p=4n+3.$$

Azaz neurononként 4 paraméter plusz 3 (egy az eltolásérték vagy bias, kettő pedig a kimenetben alkalmazott lineáris tagból $/ax+by/$).

A backpropagation hálózatnál ugyanez:

$$p=4n+1.$$

Itt ugyanis nincs lineáris kimeneti tag, csak a neurononkénti 4 paraméter plusz a bias értéke.

Ez azt jelenti, hogy 43 tanító adatnál az elméletileg alkalmazható neuronszám 10. Tíz neuron esetében ugyanis 43 ill. 41 a meghatározandó paraméterek száma. A neuronszámot lehet tovább is növelni, lehet több becsült paraméter, mint adat, de ebben az esetben a megoldás bizonytalan lesz és felmerülhet a túltanítás problémája, melyről a 2.4.1.3 fejezetben már volt szó.

A túltanítás most különösen veszélyes lehet, hiszen nincsenek adataink a teszteléshez. Csak a már korábban is alkalmazott oda-vissza transzformáció jelezheti a túltanulást és a transzformáció pontatlanságát a közbenső pontokban.

Nézzük most meg a tanítások eredményeit.

4.5.2 Transzformáció RBF hálózattal

Fokozatosan növeltem a neuronszámot 3-mal kezdve és vizsgáltam, hogy a hálózat mennyire tudja megtanulni a tanuló pontok koordinátáit. 10 neuron után a program visszajelzett, hogy több ismeretlen paramétert szeretnék meghatározni, mint amennyi adatom van, és az eredmény bizonytalan lesz. Ennek ellenére megpróbáltam annyi neuronig elmenni, amíg a háló meg nem tanulta pontosan a tanító pontokat. Megvizsgáltam, hogy milyen eredményeket kapok, ha leálllok 10 neuronnal, vagy, ha elmegyek a tanulás határáig.

Az eredményeket a geodéziában alkalmazott pontossággal vettem figyelembe. Így az EOV koordinátáknál a mm alatti eltérések esetén már úgy tekintettem, hogy a hálózat tökéletesen megtanulta az adatokat. A WGS-84 koordináták esetében pedig $0^{\circ} 0' 0,0001''$ hiba érték alatt vettem tökéletesnek a tanulást (ilyen pontossággal szokták megadni az adatokat, és ha ezt átszámítjuk hossz mértékegységre, akkor ez szintén kb. mm pontosságnak felel meg). Az $x=f_1(\Phi, \Lambda)$ kapcsolatot 13 neuronnal már tökéletesen megtanulta a hálózat, az $y=f_2(\Phi, \Lambda)$ kapcsolatot pedig 14 neuronnal. Az ellenkező irányú koordináta transzformáció tanítása nem volt ilyen sikeres, itt azért kellett 15-16 neuronnal leállnom, mert hiába növeltem a csomópontok számát a rejtett rétegben, a hálózat nem igazán tanult tovább. Ez látszik, ha összehasonlítjuk a 10 neuronnal való tanítás hibáival. Nincs túl nagy javulás.

Nézzük meg a 10 neuronnal tanított négy hálózat hibáit :

EOV – WGS-84 koordináta-transzformáció

<i>Transzformáció RBF hálóval 10 neuronnal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0.021 m	0,053 m	-0.059 m
x	0.018 m	0,042 m	-0.036 m
Φ	0.0013"	0,0024"	-0.0040"
Λ	0,0028"	0,0063"	-0,0059"

Hasonlítsuk össze ezeket az adatokat az általam készített polinomos transzformációval! Az EOV koordináták közelítése pontosabb lett, a középhibák mintegy a felére csökkentek, viszont a WGS-84 koordináták közelítése romlott. Tehát nem egyértelműen jobb az eredmények.

Ha tovább megyünk a taníthatóság határáig, akkor a következő eredményeket kapjuk:

<i>Transzformáció RBF hálóval 13-16 neuronnal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0.000 m	0,000 m	-0.000 m
x	0.000 m	0,000 m	-0.000 m
Φ	0.0010"	0,0023"	-0.0031"
Λ	0,0014"	0,0041"	-0,0026"

A WGS-84-ből EOV-ba történő transzformáció az azonos pontok hibája alapján tökéletesnek mondható. Fordítva már nem így van, de ezek az eredmények is valamivel jobbák a polinomos közelítés hibájánál.

Nézzük meg a másik ellenőrzési lehetőséget, a Magyarország területén tetszőlegesen felvett 1000 pont oda-vissza transzformáció hibáját.

<i>Oda-vissza transzformáció RBF hálóval 10 neuronnal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0,122 m	1,652 m	-0.420 m
x	0,094 m	0,502 m	-0,401 m

<i>Oda-vissza transzformáció RBF hálóval 13-16 neuronnal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0,328 m	6,801 m	-1,841 m
x	0,459 m	2,616 m	-5,416 m

Ezek az eredmények meglepőek. Ha nem léptem túl a 10 neuronos határt, amennyit az adatmennyiség elméletileg megengedett, akkor is nagyon rosszak lettek az oda-vissza hibák, akár méter fölöttiek is. Több neuron esetében pedig már több méteres hibák is adódtak, sőt a szórás is 30-50 cm között mozog, szemben a polinomos közelítés cm alatti oda-vissza hibájával.

4.5.3 Transzformáció Backpropagation (BPN) hálózattal

Nézzük meg a tanítást a 4.5.1 fejezetben említett backpropagation (a továbbiakban BPN) hálózattal. (Az egy rejtett rétegben szigmoid aktivációs függvényrel, a kimeneti rétegben lineárisan összegző neuronnal).

Ugyanúgy bemutatom 10 neuron esetében az eredményeket, és elmegyek a tanítás határáig.

Tíz neuron esetében a következő eredményeket kapjuk:

<i>Transzformáció BPN hálóval 10 neuronnal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0.016 m	0,055 m	-0.061 m
x	0.014 m	0,035 m	-0.046 m
Φ	0.0003"	0,0014"	-0.0013"
Λ	0,0060"	0,0128"	-0,0113"

EOV – WGS-84 koordináta-transzformáció

Ezek az eredmények az RBF hálóval összehasonlítva elég vegyes képet mutatnak. Az EOV koordináták hibái nagyjából hasonlóak, ill. némileg jobbak lettek. A WGS-84 koordináták közül a Φ értékek lényegesen jobbak lettek, a Λ értékek viszont sokkal rosszabbak. A polinomos transzformációval összehasonlítva, Λ kivételével jobb eredményeket kaptunk.

Érdekes még, hogy ezek a hálózatok a további tanulás tekintetében jól működtek mindkét irányban, ugyanis meg tudták tanulni a megadott hiba alatt a tanuló adatokat. Az EOV koordinátákat 14 neuron, a WGS-84 koordinátákat már 13 neuron esetében is tökéletesen megtanulták. A maximális hibák is jóval mm alattiak voltak.

Az oda-vissza hibák viszont annál meglepőbbek lettek:

<i>Oda-vissza transzformáció BPN hálóval 10 neuronnal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0,786 m	5,363 m	-2,174 m
x	0,267 m	0,846 m	-3,556 m

<i>Oda-vissza transzformáció BPN hálóval 13-14 neuronnal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	431,955 m	1997,18 m	-5607,28 m
x	1036,85 m	3982,01 m	-4880,83 m

A teszthibák már 10 neuron alkalmazása esetén is túlságosan nagyok, néha több métereseek voltak, de 13-14 neuron esetében, amikor a tanítópontok szinte hibátlanok voltak, a tesztpontok oda-vissza transzformálása hatalmas (több ezer) méteres hibákat is okozott.

Ez csak egyféleképpen magyarázható, a korábban már ismertett túltanulással. Ez az a jelenség, amikor már túlzottan illeszkedik a felület a tanuló adatokra, viszont a tanító pontok közötti tesztpontokban már túl nagy mértékűek a hibák. Erre nagyon oda kell figyelni, ezért szükséges mindig tesztpontokkal ellenőrizni a hálózatot. Sajnos nekünk a túl kevés adat miatt csak az oda-vissza transzformáció maradt ellenőrzési lehetőségként, tetszőlegesen felvett pontokban.

A fentiek alapján egyértelmű, hogy nem szabad addig elmenni a neuron szám növelésében, amikor már több a meghatározandó paraméterek száma, mint a rendelkezésre álló adatmennyiség.

Ugyancsak a túltanítás elkerülése végett nem alkalmaztam a geoidfelület közelítésénél használt neurális hálózat sorozatot az eredmények pontosítására. A geoidfelületnél elegendő adat állt rendelkezésünkre 4 tagból álló sorozat tanítására is. Ott azért lehetett maximum 35 neuront alkalmazni, mert utána már nem

növekedett kellően a pontosság a számítógép véges számábrázolásának köszönhetően, így a neuronszám felső határát nem a rendelkezésre álló adatok kevés volta szabta meg. A koordináta-transzformációnál ugyanarra arra eredményre vezetne a neurális hálózat sorozat, mintha túltanítanánk a hálózatot a sok neuronnal. Ebben az esetben is egy olyan hálózatszerkezettel próbálnánk meg közelíteni a transzformáció leképzési függvényeit, amelyben sokkal több a becsült paraméter, mint a felhasznált adatok száma.

4.5.4 Skálázás

Ha a fenti tanítási eredményeket megnézzük (nem az oda-vissza eltéréseket), akkor feltűnő, hogy mennyire másként viselkedett a tanítás a különböző koordinátákra. Bizonyos esetekben sokkal pontosabb volt a tanítás, mint a polinomos transzformációnál, más esetekben viszont rosszabb. Ez az eltérő viselkedés a be- és kimenő adatok eltérő szerkezetével magyarázható.

Ezekben a problémákban talán segíthet, ha az adatokat skálázzuk, hogy ne legyenek túl nagyok, hanem pl. 0 és 1 között vagy -1 és 1 között legyen az összes.

Elvileg a neurális hálózat bemenő adatai bármilyen valós számok lehetnek, de számítástechnikai szempontból jó ötlet a bemenő adatok skálázása, mert így a súlyok nem lesznek túl kicsik, szemben azzal, ha nem skáláztuk a bemenő adatokat és azok nagyon nagy értékek voltak.

A skálázás szükségessége nincs bizonyítva, elvileg, ha a kimenő rétegben lineárisan összegző neuront alkalmazunk (ahogy ez történt jelen esetben is), akkor ez is megoldhatja a skálázás problémáját.

Sokféle skálázási vagy normalizálási eljárás terjedt el a gyakorlatban. Normalizálásról akkor beszélhetünk, ha a bemenő adatokat, mint egy vektort tekintjük, és ennek a vektornak a hosszát módosítjuk, hogy a hossza maximum 1 legyen. Skálázás esetében külön az összes bemenetet/kimenetet módosítjuk.

A skálázás lehet lineáris, ahol megszabhatjuk, hogy az adataink pl. 0 és 1 közé vagy -1 és 1 közé essenek (ez a kettő a legelterjedtebb skálázási módszer.)

Ennek a képlete a következő:

$$x = x_{\min} + \frac{X - X_{\min}}{X_{\max} - X_{\min}} \cdot (x_{\max} - x_{\min}),$$

ahol X az eredeti érték, x a skálázott érték, X_{\min}, X_{\max} az eredeti adatok minimális/maximális értékei és x_{\min}, x_{\max} pedig az általunk meghatározott minimális/maximális értékek (pl. $0, 1$).

Egy másik lineáris skálázási eljárás, ha a skálázandó értékből kivonjuk az eredeti adatok középértékét, és ezt elosztjuk az adatok szórásával.

$$x = \frac{X - \bar{X}}{\sigma},$$

ahol X az eredeti érték, x a skálázott érték, \bar{X} az eredeti adatok átlagértéke, σ pedig a szórása.

A lineáris skálázáson kívül, használnak még többféle logaritmikus skálázást is. Az egyik alkalmazott logaritmikus skálázás képlete a következő:

$$x = \ln(X - X_{\min})$$

Arra, hogy ezek közül melyik skálázást mikor érdemes használni, egyelőre nincsenek egyértelmű szabályok.

A koordináta-transzformációnál az elsőként ismertetett lineáris skálázás (-1 és 1 ill. 0 és 1 közötti értékekkel) és a logaritmikus skálázás nem vezetett pontosabb eredményekre, mint a skálázás nélküli megoldás 10 neuron alkalmazása esetén.

A második típusú lineáris skálázás (az átlagérték kivonása után a szórással való osztás) már jobb eredményre vezetett. Ilyen típusú skálázással előállított be és kimeneti értékek esetén nem volt olyan nagy különbség a különböző koordináták tanítása során. Mindegyiket elég jól sikerült megtanítani, különösen a backpropagation hálózat esetében. Nézzük meg ezeket az eredményeket!

(A hálózatok leképzési függvényei megtalálhatóak a mellékletekben: M6,M7)

Transzformáció RBF hálóval 10 neuronnal skálázott be és kimeneti adatokkal			
	σ (szórás)	Pozitív hiba maximum	Negatív hiba maximum
y	0.020 m	0,044 m	-0.072 m
x	0.012 m	0,037 m	-0.040 m
Φ	0.0005"	0,0015"	-0.0014"
Λ	0,0005"	0,0012"	-0,0010"

Transzformáció BPN hálóval 10 neuronnal skálázott be és kimeneti adatokkal			
	σ (szórás)	Pozitív hiba maximum	Negatív hiba maximum
y	0.003 m	0,010 m	-0.010 m
x	0.003 m	0,007 m	-0.011 m
Φ	0.0002"	0,0006"	-0.0005"
Λ	0,0003"	0,0009"	-0,0008"

Ez utóbbi hálózat tanítása sikerült a legjobban, EOV koordináták esetében a középhiba 3 mm, WGS-84 esetében pedig 0,0003”. Ez nagyon jó eredmény, főleg, hogy nem léptük túl a 10 neuronos határt, amennyit az adatok mennyisége megengedett.

Nézzük meg, hogy e hálózatok általánosító képessége ugyanilyen jó-e, mint a tanulópontokra kapott eredmények. Vizsgáljuk meg, hogyan működnek a tesztpontokban oda-vissza transzformáláskor.

<i>Oda-vissza transzformáció RBF hálóval 10 neuronnal</i>			
<i>skálázott be és kimeneti adatokkal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0,107 m	0,618 m	-0,547 m
x	0,051 m	0,169 m	-0,234 m

<i>Oda-vissza transzformáció BPN hálóval 10 neuronnal</i>			
<i>skálázott be és kimeneti adatokkal</i>			
	σ (szórás)	<i>Pozitív hiba maximum</i>	<i>Negatív hiba maximum</i>
y	0,098 m	0,286 m	-0,382 m
x	0,456 m	5,208 m	-2,212 m

Úgy tűnik, hogy hiába az adatok skálázása és a jó eredmények a tanulópontoknál, a tesztpontokban ezek a hálózatok is elég megbízhatatlanul, pontatlanul működnek, szemben a polinomos transzformációval. Ismét csak szembesülnünk kellett a 2.4.1.3 fejezetben ismertetett túltanítás problémájával, a kevés adatnak köszönhetően.

Ez nem azt jelenti, hogy neurális hálózatokat nem lehet alkalmazni koordináta transzformációra és, hogy ne tudnák elérni ugyanazt a pontosságot, mint a polinomos közelítés esetében, csak azt, hogy több tanító adatra van szükségünk.

4.5.5 Polinomok közelítése neurális hálózattal

Ha úgy tekintjük, hogy a polinomos transzformáció a jó megoldás, akkor bebizonyítható, hogy neurális hálózatokkal tökéletesen lehet közelíteni a polinomokkal előállított függvényt.

A neurális hálózatok polinomközelítési képességével kapcsolatban ugyanis bebizonyították a következő tételt egy rejtett rétegű és egy kimenettel rendelkező hálózatokra vonatkozóan (F. Scarselli-Ah Chung Tsoi, 1998):

Ha $p(x)$ egy d változós algebrai polinom, amelynek fokszáma nem nagyobb, mint r , továbbá legyen $\sigma: R \rightarrow R$ függvény $r+1$ -szer folytonosan deriválható a b pont nyílt környezetében, valamint tegyük fel, hogy $\sigma^{(i)}(b) \neq 0$ igaz minden $0 \leq i \leq r$ esetre, akkor minden nemnegatív valós ε -hoz létezik egy fent definiált típusú neurális hálózat σ aktivációs függvényrel és

$$r \cdot \binom{r+d-1}{d-1}$$

számú rejtett rétegbeli csomóponttal és egy b küszöbértékkel (bias) úgy, hogy

$$\|p - f\| < \varepsilon,$$

ahol f reprezentálja a neurális hálózat által megvalósított függvényt.

Ez azt jelenti, hogy a fent definiált függvényrel megkaphatjuk, hogy hány rejtett rétegbeli csomópontot kell alkalmaznunk, ahhoz, hogy egy r -ed fokú polinomot tökéletesen közelítsünk.

5. fokú és kétváltozós polinom közelítése esetében összesen:

$$5 \cdot \binom{5+2-1}{2-1} = 5 \cdot \binom{6}{1} = 30 \text{ neuront kell alkalmazni.}$$

Ebből viszont kiszámítható a hálózat meghatározandó paramétereinek a száma is.

Egy rejtett rétegű hálózatnál (egy bias értéket is figyelembe véve) a paraméterek száma a következő:

$$p = 4n + 1, \text{ ahol } n \text{ a rejtett rétegbeli neuronok száma.}$$

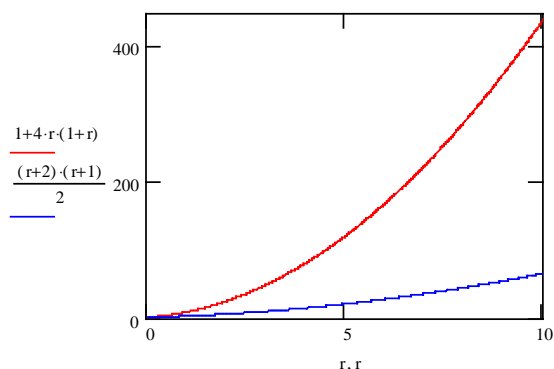
Kétváltozós függvény közelítése estében ez:

$$p = 1 + 4 \cdot r \cdot \binom{r+1}{1} = 1 + 4 \cdot r \cdot (r+1)$$

Meghatározhatjuk az r -ed fokú polinom paramétereinek a számát is. Ez a következő:

$$p = \frac{(r+2) \cdot (r+1)}{2} = 1 + r \cdot (r+3)$$

Összefoglalás



24. ábra Polinom és a közelítéséhez használt neurális hálózat meghatározandó paramétereinek a száma

Látszik, hogy azonos közelítési pontosság esetén az algebrai polinom kevesebb ismeretlen paraméterrel rendelkezik, mint a neki megfelelő neurális hálózat, tehát ebből kifolyólag kevesebb adatra is van szüksége az ismeretlenek meghatározásához.

5. fokú, kétváltozós polinom ismeretlen együtthatóinak a száma 21, míg ugyanezen polinom neurális hálózattal való közelítéséhez 121 súlyt kell meghatározni. Ha tehát a polinommal egyező pontosságot szeretnénk neurális hálózattal előállítani, akkor ehhez minimum 121 adatra lenne szükségünk, vagy többre.

Ezt ellenőriztem is a koordináta-transzformációra. Úgy tekintetem, mintha a polinomos megoldás hibátlan lenne. Magyarország területén felvettem tetszőlegesen 150 tanulópontot és 150 tesztpontot, és ezekre kiszámoltam a koordináta-transzformáció értékét a polinomos közelítéssel. Utána egy 30 rejtett rétegbeli neuront tartalmazó hálózatot próbáltam megtanítani a tanító adatokra, majd a tesztpontokban is ellenőriztem a transzformáció pontosságát. Ezzel a hálózattal nem volt semmi gond, tökéletesen megtanulta a tanítópontok adatait mindkét irányú koordináta-transzformáció esetében. A hibák a tesztpontokban is jóval mm alatt maradtak, és nagyságrendileg azonosak voltak a tanítópontok hibáival.

A kapott eredmények alapján azt mondhatjuk, hogy a koordináta-transzformációnál a rendelkezésre álló 43 azonos pont csekély számát figyelembe véve pontosabb a hagyományos polinomos transzformáció, mint a neurális hálózattal készülő. Ugyanakkor, ha több azonos pont állna rendelkezésre a neurális hálózatokat is megfelelően lehetne alkalmazni.

5 Összefoglalás

Mint minden területen, a geodéziában is fontos figyelemmel kísérenünk a technikai újításokat. Éppen ezért a diplomamunkám célja egy új informatikai eszköznek, a neurális hálózatok geodéziai alkalmazhatóságának a bemutatása volt.

Összefoglalás

Mivel ez viszonylag új tudományterület és a geodézián belül csak kevesen foglalkoznak vele, ezért először általános áttekintést adtam ezeknek a hálózatoknak a felépítéséről, fajtáiról és működéséről. Majd két gyakorlati példán keresztül mutattam be a használatukat.

Az első feladat a geoidfelület közelítése volt, ami tulajdonképpen egy felületmodellezési problémának felel meg. Itt kiindulási adatként több mint 200000 pontban voltak ismertek a geoidunduláció értékei. Ezekre illesztettem felületet a hagyományos polinomos regresszióval és neurális hálózatokkal. A véges számábrázolás következtében 6. fokúnál magasabb polinomot nem tudtam illeszteni a felületre, a rosszul kondicionáltság növekvő mértéke miatt. A neurális hálózatokkal való közelítést egy 4 tagból álló neurális hálózat sorozattal végeztem. Ez utóbbi közelítés pontossága messze meghaladta a polinomos regresszió eredményeit. Ezek az eredmények azt tükrözik, hogy érdemes ezt az új módszert alkalmazni a geodéziában gyakori felületközelítési, domborzatmodellezési feladatoknál.

A második feladat egy EO-VGS-84 koordináta-transzformáció volt. Itt is összehasonlítottam egy hagyományos, általános sorokkal (polinomokkal) történő transzformációt a neurális hálózatokkal. A megoldáshoz összesen 43 azonos pont állt rendelkezésemre. Ennél a feladatnál a hagyományos módszer bizonyult jobbnak, megbízhatóbbnak, ami a rendelkezésre álló adatok kevés számával magyarázható. Bizonyítható azonban, hogy jóval több adat esetén neurális hálózattal is elérhető a polinomos közelítés pontossága, tehát nyugodtan alkalmazható koordináta-transzformációra is ez az eszköz, abban az esetben, ha megfelelő mennyiségű azonos pont áll rendelkezésünkre.

Végezetül összefoglalom a dolgozatom eredményei alapján levonható következtetéseket a neurális hálózatok geodéziai alkalmazhatóságára vonatkozóan.

A dolgozatomban függvény és felületközelítési feladatokat vizsgáltam a geodézián belül. A közelítési probléma megoldási módszerének kiválasztását a rendelkezésre álló adatok mennyisége szabja meg. A 24. ábrából leolvasható, hogy azonos közelítési pontosság esetén a hatványpolinomnak jóval kevesebb meghatározandó paramétere van, mint a neurális hálózatnak. Először tehát érdemes mindenképpen a hatványpolinomokat megvizsgálni. A közelítés javítása érdekében lehet a polinom fokszámát növelni egy bizonyos határig. Ezt a határt az egyenletrendszer együttthatómátrixának a kondicionáltsága szabja meg, ugyanis bizonyos fokszám fölött ez a mátrix rosszul kondicionált lesz és a megoldás bizonytalanná válik.

A polinomos közelítés határát elérve neurális hálózatokkal lehet tovább pontosítani az eredményeket. Ehhez szükséges, hogy elegendő adat álljon rendelkezésre, ugyanis a neurális hálózatoknak sokkal több az ismeretlen paramétere (ld. 24. ábra). Kevés adat esetén könnyen előfordulhat a hálózat túltanításának a problémája. A közelítés pontosságát a neuronszám növelésével lehet fokozni. Ennek is határt szab azonban a számítógép véges számábrázolásából eredő hibája, egy bizonyos neuronszám fölött ugyanis már ez a közelítési módszer sem javul. Ekkor az eredményeken még javíthat a geoidfelület közelítésénél is alkalmazott neurális hálózat sorozat.

A feladat megoldása során tehát meg kell határozni a kívánt pontosságot és vizsgálni a rendelkezésre álló adatok mennyiségét, és ezek alapján lehet kiválasztani

Összefoglalás

a legjobb megoldási módszert a polinomos, a neurális hálózattal és a neurális hálózat sorozattal való közelítés közül.

A neurális hálózatok legfőbb előnye tehát abban rejlik, hogy igen nagyszámú adat esetén jól használhatóak függvény approximációra, amikor a polinomokkal való közelítés már nem működik a rosszul kondicionáltság miatt. Ezért úgy gondolom, hogy ezt az eszközt a közeljövőben egyre több feladatnál fogják alkalmazni a geodéziában, térinformatikában. Érdeemes foglalkozni vele.

Hivatkozások

1. Barsi Árpád (1999): Koordináta-transzformáció megoldása neurális hálózatokkal, Geodézia és Kartográfia, Budapest, LI, No. 10. 12-18.
2. Barsi Árpád (1997): Landsat-felvétel tematikus osztályozása neurális hálózattal. Geodézia és Kartográfia, 1997/4, pp. 21-28.
3. Bíró Péter (1996): Felsőgeodézia, Műegyetemi Kiadó, Budapest
4. Detrekői Ákos (1991): Kiegyenlítő számítások, Tankönyvkiadó, Budapest
5. Horváth Gábor (1995): Neurális hálózatok és műszaki alkalmazásai, Műegyetemi Kiadó, Budapest
6. Krauter András (1995): Geodézia, Műegyetemi Kiadó, Budapest
7. Mérő László (1989): Észjárások, A racionális gondolkodás korlátai és a mesterséges intelligencia, Akadémiai Kiadó, Optimum Kiadó, Budapest
8. Paláncz, B.-Völgyesi, L (2002): High accuracy data representation via sequence of neural networks, megjelenés alatt
9. Paláncz Béla (2003): Neurális hálózatok a Mathematica felhasználásával, 'www.fmt.bme.hu/~palancz/education.html'
10. Sárközy Ferenc (1998): Mesterséges neurális hálózatok mint GIS függvények 'http://bme-geod.agt.bme.hu/public_h/neurgis/neurmint.html'
11. Scarselli, Franco-Ah Chung Tsoi (1998): Universal Approximation Using Feedforward Neural Networks: A Survey of Some Existing Methods, and some New Results, Neural Networks, Vol. 11, No. 1, pp. 15-37, Elsevier Science Ltd., Great Britain
12. Tóth Gy- Rózsa Sz (2000): New Datasets and Techniques – an Improvement in the Hungarian Geoid Solution. Paper presented at Gravity, Geoid and Geodynamics Conference, Banff, Alberta, Canada July 31-Aug 4, 2000.
13. Tóth Gy - Rózsa Sz - Andritsanos V D - Ádám J - Tziavos I N (2000): Towards a cm-Geoid for Hungary: Recent Efforts and Results. Phys. Chem. Earth (A), Vol. 25. No. 1. pp. 47-52.
14. Varga József (1997): Vetülettan, Műegyetemi Kiadó, Budapest
15. Veres Gábor (2002): RBF neurális hálózat alkalmazása magasság meghatározására, Geodézia és Kartográfia, Budapest LIV. No. 7. 25-30.
16. Vetületi szabályzat az egységes országos vetületi rendszer alkalmazására (1975), Budapest, MÉM Országos Földügyi és Térképészeti Hivatal
17. Völgyesi Lajos-Tóth Gyula-Varga József (2001): Magyarországi vetületi rendszerek közötti átszámítások, a VETULET+ program leírása, A leírás elektronikus verziója megtalálható a 'http://sci.fgt.bme.hu/~volgyesi/vetulet/gkvetful.pdf ' internet címen

18. Wickham-Jones, Tom (1994): *Mathematica Graphics*, Springer-Verlag, New York

Mellékletek

(M1-M8)

M8 Polinom közelítésének tanuló- és tesztpontjai

Az ötödfokú polinom neurális hálózatokkal való közelítéséhez felvett 150-150 tanuló és tesztpont elhelyezkedése:

