



TDK DOLGOZAT

Molekuláris elektronikai mérések elemzése neurális hálózattal

Balogh Nóra

Témavezetők: **Dr. Halbritter András**

egyetemi tanár

BME Fizika Tanszék

Magyarkuti András

BME Fizika Tanszék

Budapesti Műszaki és Gazdaságtudományi Egyetem

2018

Összefoglaló

A molekuláris elektronikai kutatások egyik főbb kérdésköre, hogy hogyan lehetne egyetlen atomból vagy molekulából álló áramköri elemeket létrehozni. Egyedi molekulák vezetőképességének vizsgálatára napjainkban az ún. MCBJ (mechanikusan szabályozható törőkontaktus) technikával kiemelkedő mechanikai stabilitás mellett nyílik lehetőség.

Az ilyen módon elvégzett mérésekből igen nagy számú vezetőképesség-görbe keletkezik, melyek között megtalálhatók molekulák jelenlétét mutató és nem mutató görbék is. Az egyik alapvető feladat, hogy a molekulák jelenlétét mutató görbéket kiszűrjük a mért adatokból. A mért görbék nagy száma miatt erre a célra mindenképpen egy automatizált algoritmusra van szükség. Az adatok osztályozására, gyors feldolgozására használt korábban elterjedt módszerek olyan algoritmusok, melyek több, sokszor nagyon bonyolult feltételek alapján tudják szűrni a molekuláris jelleget mutató görbéket.

Kutatási munkám keretén belül olyan neurális háló alapú osztályozás megvalósítását tűztem ki célul, mely korábbi mérési adatokon betanulva maga határozza meg a különböző szűrési feltételeket. Munkám során szempont volt, hogy egyszerű, jól átlátható modellt valósítsak meg a vezetőképesség-görbék osztályozására, közvetlenül a molekuláris elektronikai mérési adatok feldolgozását végző IGOR szoftveren alapuló kiértékelő programban. Munkám során kitérek a modell paramétereinek megfelelő megválasztására, és a modell általánosíthatóságára is.

Az alkalmazott neurális háló egyszerűségét kihasználva a neurális háló által betanult információkat is szeretném szemléletessé tenni. A neurális háló súlyainak elemzésével abba is bepillantást szeretnék nyerni, hogy milyen szempontokat használ a háló a döntéshozatalnál.

A dolgozat az Emberi Erőforrások Minisztériuma ÚNKP-18-2 kódszámú Új



Nemzeti Kiválóság programjának támogatásával készült.

Tartalomjegyzék

1. Motiváció	vi
2. MCBJ-technika	2
2.1. MCBJ mérés technika	2
2.2. Atomi kontaktusok vizsgálata	3
2.3. Egyedi molekulák vezetőképességének vizsgálata	4
2.4. Problémafelvetés	5
2.5. Módszerek a vezetőképesség-görbék osztályozására	6
3. Neurális háló	9
3.1. Bevezetés	9
3.2. Neurális hálózatok felépítése	10
3.2.1. Perceptron, aktivációs függvény	10
3.2.2. Hálózat topológiája, MLP	12
3.3. Neurális hálózat tanulása adatkból	13
3.3.1. Az alkalmazott tanítási stratégia	14
3.3.2. A hálózat általánosító képessége, validáció	14
3.3.3. Keresztvalidáció	15
3.4. Hálózat konstrukciójának általános kérdései	15
4. Vezetőképesség-görbék osztályozására alkalmas neurális háló fejlesztése	20
4.1. Neurális háló az Igor Pro programcsomagban	20
4.2. Adatelőkészítés	21
4.3. Keresztvalidáció megvalósítása	24
4.4. Optimális paraméterek keresése	25
4.4.1. Iterációk számának hatása	25
4.4.2. Bemeneti adatpontok számának hatása	26
4.4.3. Vezetőképesség-tartományok megválasztása	27
4.4.4. A legegyszerűbb, 12 bines modell	29

4.5. Neurális hálós modellek tesztelése	30
4.5.1. Modell tesztelése alacsony hőmérsékletű adatokon	31
4.5.2. Modell tesztelése magas hőmérsékletű adatokon	33
5. Mit tanult meg a hálózat?	35
5.1. Hálózat súlyainak értelmezése	35
5.2. Két bemenő adaton tanuló hálózat működésének szemléltetése	37
6. Összefoglalás, kitekintés	41
Köszönetnyilvánítás	43
Irodalomjegyzék	44

1. Fejezet

Motiváció

Kutatási munkám a nanofizika területén, molekuláris elektronikai mérési adatok osztályozásának neurális hálókkal történő megvalósítását foglalja magába. A következőkben szeretném áttekinteni témaválasztásom két fő motivációját.

Szakedolgozatom során a nanofizika területén belül a memrisztorokkal foglalkoztam, melyek olyan passzív áramköri elemek, melyeknek ellenállása függ a korábban rajtuk átfolyó áramtól. Memrisztorok többek között ezüst-szulfid, vagy ezüst-jodid vékonyrétegben is kialakíthatók. Ekkor egy ezüst elektróda és egy hegyes, inert tű (például platina-iridium) közötti ezüst-szulfid vagy ezüst-jodid rétegben hozunk létre vékony vezető szálat. Az ezüst elektródára kapcsolt nagy pozitív, illetve negatív feszültséggel a memrisztort két különböző ellenállású állapota között kapcsolgathatjuk, kisebb feszültséggel pedig kiolvashatjuk, hogy melyik állapotban van. A memrisztor tehát memóriaként működik. A memrisztorok gyors, kis fogyasztású háttértárolóként való alkalmazására már számos számítástechnikai cég (mint például Intel, Samsung, IBM, SanDisk, HP) folytat előrehaladott fejlesztéseket.

A memrisztorok távolabbi, de talán még izgalmasabb alkalmazási területe lehet a mesterséges neurális hálózatok fejlesztése. A Nature Materials folyóiratban egy korábbi publikáció megmutatta, hogy nanométeres skálájú ezüst-szulfid memrisztorokkal jól szimulálható az idegrendszer hosszú- és rövidtávú memóriafunkciója [1]. A szakdolgozatomban ehhez hasonló kísérleti elrendezéssel és mérésekkel kívántam demonstrálni a memrisztort, mint egy (egyelőre egyetlen egységből álló) mesterséges szinapszist.

Számos publikáció ír arról, hogy a memrisztorok a napjainkban népszerű mesterséges neurális hálózatok hardver szintű megvalósításában is megoldáshoz vezethetnek [2][3]. Kutatási munkám távolabbi célja ennek köszönhetően az, hogy a memrisztorokból alko-

tott hálózatokat modellezem, mellyel a mesterséges neurális hálózatok hardveres megépítésének megalapozásához szeretnék hozzájárulni. Ehhez első lépésként a szoftver alapú neurális hálók és a gépi tanulási módszerek működésének megismerése szükséges. Ezen ismereteim elmélyítésének célja volt a TDK munkám egyik fő motivációja.

A témaválasztást indokló másik fontos tényező a molekuláris elektronikai mérési adatok hatékony osztályozása iránti igény. A 2.5. fejezetben bemutatom a korábban elterjedt módszereket, melyek a mérési adatok osztályozására gyakran nagyon bonyolult feltételekkel képesek. A különböző statisztikai módszereken, a vezetőképesség-görbék egy-egy tulajdonságán alapuló osztályozáson túl többek között akár a görbék korrelációjának vizsgálatán alapuló főkomponens analízissel is lehet molekuláris elektronikai mérési adatokat válogatni. A közelmúltban a kutatócsoportunk a Koppenhágai Egyetemmel együttműködve elsőként használt gépi tanulási algoritmusokat ezen a területen az adatok elemzésére [4]. A kutatásuk eredményeképpen megalkotott rekurrens neurális hálón alapuló modell eredményesen osztályozza ugyan a mérési adatokat, azonban nagyon komplex és számos paramétert használ, melyek közül sokra érzékeny. A rekurrens neurális hálót alkalmazó osztályozás másik nagy hátránya, hogy nincs mód arra következtetni, hogy a hálózat a bemeneti adatokból milyen szempontok alapján osztályoz. Munkám során ezért igyekszem egy olyan egyszerűségű, de az adatokat jól osztályozó neurális háló alapú osztályozó modellt kialakítani, mely lehetőséget teremt a háló által betanult információk szemléltetésére is.

2. Fejezet

MCBJ-technika

A számítástechnikai eszközök méretének csökkentése iránti igény a molekuláris elektronika fejlődésének egyik hajtóereje. A területen zajló kutatások azt vizsgálják, hogy elektronikus eszközökben hogyan lehetne egy-egy atommal vagy molekulával helyettesíteni a mai áramköri elemeket. Az atomi méretskálás eszközök fejlesztésének egyik fontos része a néhány atomból álló kontaktusok vezetési jelenségeinek vizsgálata.

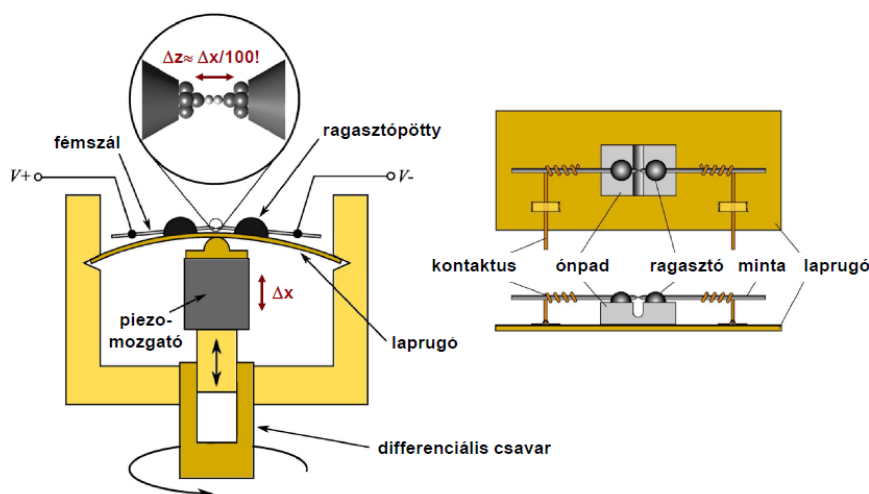
2.1. MCBJ mérés technika

Az atomi kontaktusok vizsgálatára alkalmas mérőberendezések egyike az úgynevezett MCBJ (Mechanically Controllable Break Junction), azaz mechanikusan szabályozható törőkontaktus technika.

A 2.1.ábrán látható MCBJ-elrendezésben egy, a közepén elvékonyított fémszál két ragasztópötty segítségével egy két végénél befogatott rugalmas lapka egyik oldalára van rögzítve. A lapka másik oldalához piezo mozgató ér, melyre feszültséget kapcsolva a piezo megnyúlása a lapka meghajlását és ezzel együtt a fémszál két rögzített pontjának eltávolodását okozza.

A piezo mozgatása nm alatti pontossággal végezhető, az MCBJ elrendezés geometriájának köszönhetően pedig a szétszakítandó vezeték végeinek elmozdulása a piezo mozgásánál két nagyságrenddel kisebb. Ebből fakadóan a fémszál elszakadása előtt közvetlenül kialakuló akár egyetlen vagy pár atomból álló atomi kontaktusok hosszú időn keresztül, kiemelkedő mechanikai stabilitás mellett vizsgálhatók.

Egy minta teljes elszakadása után a fémszál végeinek közelítésével a kontaktus újra



2.1. ábra. Az MCBJ-elrendezés vázlata [5].

felépül, így egy mintával akár több ezer szakítási görbe is felvehető.

2.2. Atomi kontaktusok vizsgálata

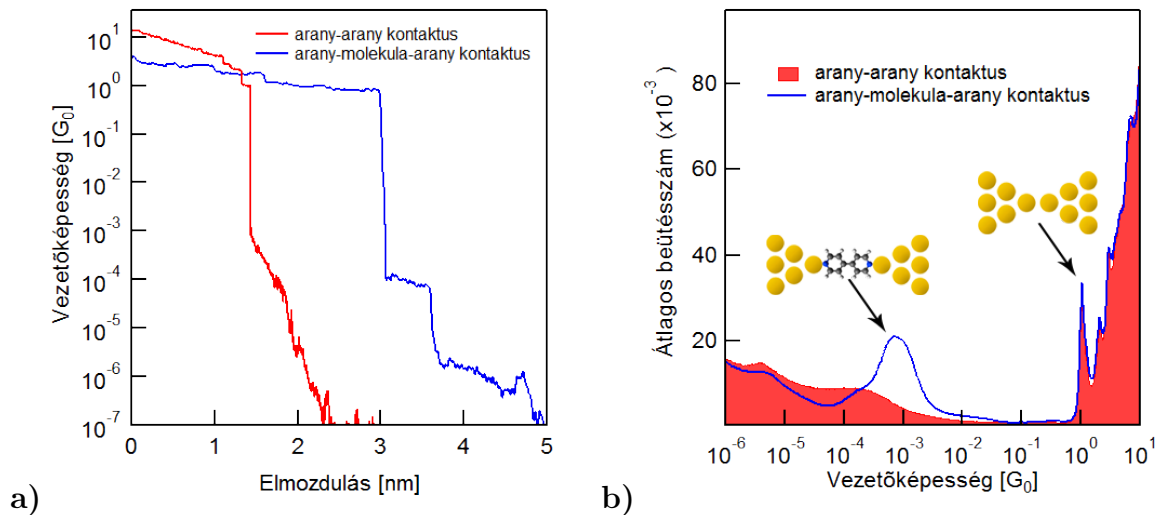
Egy makroszkopikus méretű fémszál elszakításakor kezdetben a vezetőképesség folytonosan csökken a fémszál keresztmetszetével arányosan. Amikor viszont elérjük, hogy a vezeték keresztmetszetében már csak néhány atom található, azt tapasztaljuk, hogy a vezetőképesség lépcsőzetesen változik.

A vezetőképesség ezen viselkedése elsősorban az anyag atomi felépítésével magyarázható, de a folyamatban az elektronok kvantummechanikai viselkedése is fontos szerepet játszik [6]. A vezetőképesség ugrásszerű változása a nanokontaktus nyújtása közben ugrásszerűen jelentkező atomi átrendeződéseket tükrözi [7]. A fémszál teljes szétszakadása előtti utolsó plató a legtöbb esetben egy olyan kontaktushoz tartozik, melyben a fémszál két végét, mint két elektródát már csak egyetlen atom köti össze. Ekkor a vezetőképesség érték arany vezeték esetén 12900Ω -nak felel meg, mely a $G_0 = \frac{2e^2}{h}$ vezetőképesség-kvantum értéke, ahol e az elektron töltése, h pedig a Planck-állandó [6]. A nanovezeték további széthúzásával előfordulhat az is, hogy több atomot "húzzunk ki" a kontaktusból, létrehozva egy egyetlen atom vastagságú nanovezetéket, miközben a kontaktus vezetőképessége változatlanul $1 G_0$ marad.

2.3. Egyedi molekulák vezetőképességének vizsgálata

Az MCBJ-technikával a vezeték folyamatos széthúzása majd összenyomása közben molekulákat is juttathatunk a kontaktushoz, melyek képesek beépülni a két elektróda közé. A molekuláris elektronika ezen területén lehetőség nyílik egyedi, szerves molekulák vezetőképességének vizsgálatára, molekuláris elektronikai eszközként áramköri szerepük feltérképezésére.

Egy kontaktusba beépülő molekula újabb geometriai elrendeződést von maga után. Az adott molekula beépülésére, geometriai beállítására a vezetőképességet mérve, majd logaritmikusan ábrázolva a 2.2/a). ábrán megjelenő újabb vezetőképesség-platók utalnak. A 2.2/a) ábrán a molekula nélkül mért vezetőképesség-görbén megfigyelhető az a jellegzetes tulajdonság, hogy a szétszakítás közben az utolsó vezetőképesség-plató utáni szakaszban a vezetőképesség a távolság függvényében exponenciálisan cseng le, azaz logaritmikus skálán egy egyenest látunk. A jelenség háttérében az áll, hogy a nanovezetéken a kontaktusvégek távolságának függvényében exponenciálisan változó alagútáram folyik keresztül. Az ilyen jelleget mutató görbékre alagútgörbe vagy alagút-típusú vezetőképesség görbe néven fogok hivatkozni.



2.2. ábra. a) Logaritmikus skálán ábrázolt vezetőképesség az arany elektródák elmozdulásának függvényében molekula jelenlétében és anélkül [8]. b) Vezetőképesség-görbékéből készített egydimenziós hisztogram, melyen molekulák beépülése esetén újabb csúcs jelenik meg [8].

A molekulák bekötődése esetén az újabb vezetőképesség-platók jellemző értékét

vezetőképesség-hisztogramokat készítve szemléltethetjük. Egy hisztogram elkészítésekor a vezetőképesség-értékeket intervallumokra (úgynevezett binekre) osztjuk, majd megszámloljuk, hogy egy-egy intervallumban hány mérési pont található, ezt $N_i(r)$ -el jelöljük, ahol i a bin-eket, r pedig a mérési görbéket indexeli. Az így elkészülő egydimenziós hisztogramokat több vezetőképesség-görbére átlagolva az $\langle N_i(r) \rangle$ -t kapjuk, amin az adott molekulára jellemző karakterisztikus vezetőképesség értékeknél csúcs jelenik meg (2.2/b). ábra).

A dolgozatomban feltüntetett molekulák beépülését mutató vezetőképesség-görbék és mérési adatok mindegyike a 4,4'-bipyridine molekula beépülését vizsgálja arany nanovezeték szétszakításakor MCBJ-technikával.

2.4. Problémafelvetés

Az MCBJ-technikával elvégzett méréssorozatok eredményeképp igen nagy számú vezetőképesség-görbe keletkezik. A folyamat során az adott nanovezeték sorozatos elszakítása közben molekulák nem minden esetben épülnek be a kontaktusba, illetve egy-egy beépüléskor mért vezetőképesség-görbe függhet a bekötődés geometriájától, a molekula bekötődési szögétől, stb. Ezekből következik, hogy MCBJ-adatok feldolgozása egy big data probléma, hiszen az adatokra teljesülnek annak legfőbb jellemzői (angolul "the four V's of big data") [4]:

1. Nagyon nagy adatmennyiség (Volume). Egy-egy mérés gigabyte-os adatmennyiséget is generálhat.
2. Változatos adatok (Variety). A vezeték nagyon sokféleképpen szakadhat szét, a molekulák bekötődése is számos módon megvalósulhat.
3. Gyors adatfeldolgozás (Velocity). A mintavételezéskor több százezer adatot vehetünk fel másodpercenként.
4. Adatok megbízhatósága (Veracity). Mi az adatból a jel és mi a zaj?

A mérési adatok egyik legkézenfekvőbb osztályozása az alagútgörbére és molekula beépülésére jellemző görbékre történő válogatás. A mért görbék nagy száma miatt erre a célra mindenképpen egy automatizált algoritmusra van szükség.

A molekulák jelenlétét mutató görbék mérési adatokból való kiszűrésére, az adatok gyors feldolgozására korábban elterjedt módszerek olyan algoritmusok, melyek több, sokszor nagyon bonyolult feltételek alapján tudják szűrni a molekuláris jelleget mutató görbéket.

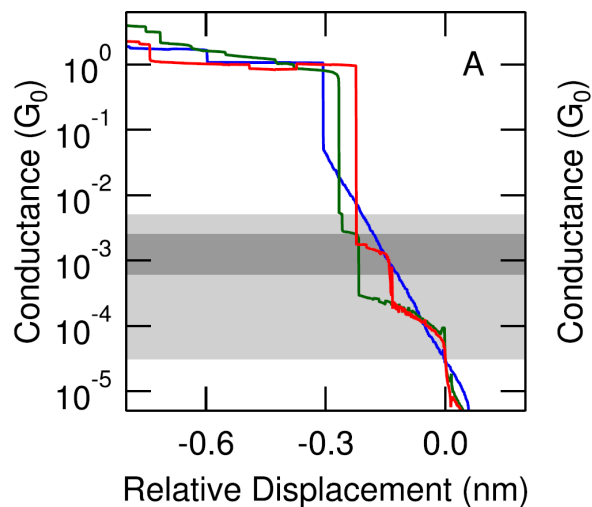
2.5. Módszerek a vezetőképesség-görbék osztályozására

A megoldandó osztályozási problémára természetesen számos megoldást, különböző szempontokból történő megközelítést javasoltak már a szakirodalomban, melyeket a teljesség igénye nélkül szeretnék röviden bemutatni.

Az első és legegyszerűbb, az adott vezetőképesség tartományban lévő beütésszám meghatározásán alapuló osztályozási módszert a 2.3. ábra alapján szeretném bemutatni. Az ábrán a pirossal és zölddel jelölt vezetőképesség-görbéken nagyjából a 10^{-3} - 10^{-4} G_0 vezetőképesség-tartományon plató jelenik meg, ami egy-egy molekula kontaktusba való beépülésére utal. A kézzel jelölt görbén jól látható az alagútgörbék jellemző exponenciális távolságfüggése, mely ez esetben jóval a molekulás vezetőképesség-tartomány fölött elkezdődik. Az osztályozási módszer alapötlete, hogy a molekula beépüléséhez tartozó karakterisztikus vezetőképesség-tartományba eső mérési adatpontok számát a molekuláris plató a hosszával arányosan jelentősen megnöveli, míg az alagútgörbék esetén nincs ilyen növekedés a mérési adatpontok számában. Ezen módszert beütésszám- vagy platóhosszúság-alapú osztályozásnak is nevezik.

A beütésszám-szerinti osztályozás hiányosságait azonban jól szemlélteti a 2.3. ábra. Az ábra egy példa is arra, hogy a különböző molekuláris konfigurációk változatos vezetőképesség-értékeknél előfordulhatnak, a hisztogramok molekuláris csúcsának szélességét növelve. Ha a vizsgált vezetőképesség-tartományt túl szélesre választjuk (az ábrán világos szürke sáv), akkor az ábrán kézzel jelölt alagútgörbét is molekulásnak osztályoznánk a tartományba eső beütésszámok, azaz mérési adatpontok száma alapján. Az osztályozási hiba csökkentésének érdekében az adódna, hogy szűkítsük le jobban a vezetőképesség-tartományt, például a 2.3. ábrán sötétszürke sávval jelölt régióra. Ekkor az ábrán pirossal jelölt görbét már molekulás görbének osztályoznánk a plató miatti nagyobb beütésszámnak köszönhetően, a kézzel jelölt görbét pedig alagút-típusúnak sorolnánk be. Lenne azonban számos görbe, mint az ábrán zölddel jelölt, mely molekula jellegű, azonban a

molekuláris plató kívül esik a vizsgált szűkebb vezetőképesség-tartományon, így rosszul osztályozzuk.



2.3. ábra. Három vezetőképesség görbe. A szürke terület azon széles vezetőképesség-tartomány, melyen a molekulák hatása várható [4].

Kitekintésképp megemlíteném, hogy a vezetőképesség-görbék osztályozására a görbék korrelációjának vizsgálatán alapuló ún. főkomponens-analízis is használható [9][10]. A módszer részleteiben a megjelölt forrásokban olvasható, lényege, hogy az egyes vezetőképesség-görbéket egy nagy adatmátrixban tárolják el binékre osztva, majd ezen adatmátrixokra számolják ki az ún. korrelációs mátrixot. Az adatokat a korrelációs mátrix diagonalizálásával kapott sajátvektorokra vetítve osztályozzák.

Fontos azt is megemlítenem, hogy az általam bemutatott neurális hálós megközelítés nem újdonság, K. Lauritzen és csapata, témavezetőimmel együttműködve rekurrens neurális hálókkal már osztályozott hasonló vezetőképesség görbéket [4]. Céljuk többek között az volt, hogy elméleti szimulációk alapján készített vezetőképesség-görbékkel betanított rekurrens neurális hálózattal tudjanak mérési görbéket osztályozni, ezzel pedig jól ellenőrizhetővé tenni az adott atomi konfigurációk hatását a vezetőképesség-görbékben. A módszer hátránya talán az volt, hogy a rekurrens neurális hálózatban sok szabad paraméter állítható volt, melyekre igen érzékeny volt az osztályozás eredménye. A modell komplexitása miatt pedig nem volt követhető, hogy a hálózat mi alapján osztályozza a görbéket. További nehézséget okozott, hogy különböző osztályozási feladatokra más és más paraméterekkel megadott neurális hálózatok voltak optimálisak és nem találtak általánosan jól alkalmazható paramétereket.

Munkám során ezen korábbi megközelítések tükrében szeretnék visszanyúlni a lehető legegyszerűbb és legjobban nyomonkövethető neurális hálókhoz. Egy jól osztályozó modellt szeretnék kialakítani, melynek egyszerűségét kihasználva a döntési mechanizmusát is szeretném szemléletessé tenni.

3. Fejezet

Neurális hálók

A mesterséges neurális hálózatok a természetes biológiai neurális hálózatok által inspirált számító rendszerek. A hagyományos algoritmusokhoz képest az előnyük, hogy a mintákból, példákból nyert információk felhasználásával tanulás útján alakítják ki a feladatmegoldó képességüket. Gyakorlati szempontból jelentőségük igen nagy, hiszen egy ilyen algoritmust olyan feladatok megoldására is használhatunk, ahol az adott folyamat nem teljeskörűen ismert, vagy nem áll rendelkezésre a problémához illeszkedő speciális modell vagy algoritmus.

A fejezetben áttekintem a neurális hálókkal kapcsolatos legfontosabb fogalmakat, a hálózatok felépítését és a hozzájuk kapcsolódó legismertebb tanító eljárásokat, továbbá bemutatom a kutatómunkám során alkalmazott, az Igor Pro szoftverbe épített modell jellegzetességeit is. Mivel a neurális hálózatok témakörében nem alakult még ki egységes magyar nyelvű terminológia, ezért dolgozatomban számos esetben az ismertetett fogalmak angol megfelelőjét is feltüntettem.

3.1. Bevezetés

A neurális hálózatok számítási feladatok megoldására létrejött párhuzamos feldolgozást végző, adaptív eszközök [11]. Az élő szervezetekben lévő bonyolult számítási rendszerek megismerése inspirálta azt a gondolatot, hogy próbáljunk meg sok azonos vagy hasonló elemből álló, nagymértékben összeköttetésben lévő hálózatot kialakítani, mely a párhuzamos felépítés mellett tanulási képességekkel is rendelkezik. A bemeneteik és kimeneteik között nemlineáris leképezést megvalósító neurális hálózatok viselkedésüket a

környezetből nyert ismeretek, tapasztalatok segítségével tudják javítani. Bebizonyítható, hogy megfelelő kialakítás esetén a neurális hálózatok szinte tetszőleges be-kimeneti kapcsolatot, leképezést meg tudnak valósítani [11], ezért az irodalomban gyakran univerzális approximátorokként is emlegetik őket.

A neurális hálózatok működésénél tipikusan megkülönböztetünk két fázist, a tanulási és előhívási fázist [11]. A tanulási fázis során a hálózat kialakítása történik. Ez magába foglalja azt, hogy a rendelkezésre álló adatok alapján a későbbiekben bemutatott tanulási algoritmusok valamelyikével a hálózat megtanulja az információ feldolgozás módját, azaz kialakítsa az átvitelét. Az előhívási fázisban a kapott információ feldolgozó rendszer használatára kerül sor. A 3.3. fejezetben bemutatom a háló tanításának menetét, a hálózatok alkalmazására, előhívására pedig a kutatási eredményeimnél térek majd ki.

3.2. Neurális hálózatok felépítése

3.2.1. Perceptron, aktivációs függvény

A neurális hálózatok alapegysége a neuron vagy processzáló elem (*processing element*), mely az idegsejt bizonyos tulajdonságait megragadó modelljének is tekinthető. A 3.1. ábrán látható processzáló elem működése azon alapul, hogy a bemeneteire érkező jelek súlyozott összegét képezi, majd az eredményre egy nemlineáris függvényt alkalmazva kapja meg a skalár kimenetét. Ezen egyszerű felépítésű neuront *perceptron* néven Frank Rosenblatt [12], illetve *adaline* néven Bernard Widrow [13] javasolta. Az egyszerű neuron arra képes, hogy két lineárisan szeparálható bemeneti mintahalmazt szétválasszon.

A 3.1. ábrán x_1, \dots, x_N -el jelölt bemeneteket a neuron w_i súlyokkal (*weight*) szorozza meg, majd összeadja őket. Az összeadott értékekhez hozzáad egy a neuronra jellemző eltolás (*bias*) értéket. Az így kapott összegből a neuron kimenetét előállító nemlineáris függvényt aktivációs függvénynek (*activation function*) vagy más néven transzfer-függvénynek hívják. A leggyakrabban használt aktivációs függvények monoton növekvőek. Ilyenek lehetnek többek között a szignum, vagy a szigmoid alakú függvények, mint a tangens hiperbolikus függvény vagy a logisztikus függvény.

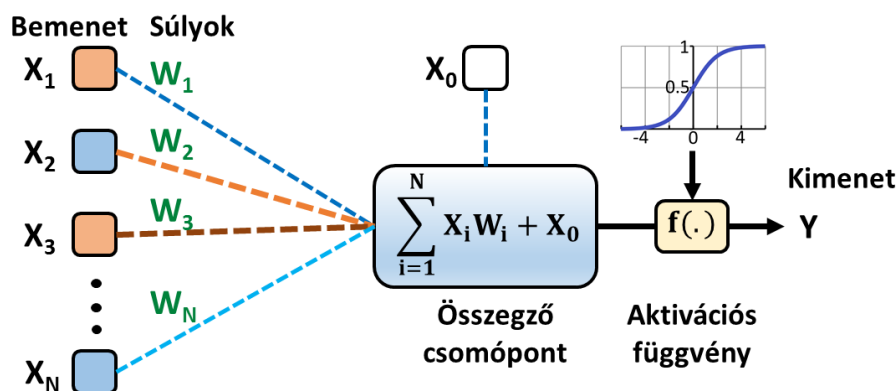
A tangens hiperbolikus aktivációs függvény határértéke a plusz végtelenben $+1$, a mínusz végtelenben -1 , így kimenete ezen intervallumba esik. Munkám során alkalmazott neurális hálók függvénye a logisztikus függvény volt, melyet a következő összefüggés ad

meg:

$$f(x) = \frac{1}{1 + e^{-x}}$$

A függvény határértéke a plusz végtelenben +1, a mínusz végtelenben 0. Ez a függvény tulajdonképpen a szilárdtestfizikában is jól ismert Fermi-függvény.

A 3.1. ábrán az x_0 érték nem bemenő érték, hanem az ún. eltolás (*bias*) értéke, mely azért felelős, hogy például tangens hiperbolikus aktivációs függvény esetén a kimenet előjelváltása, logisztikus függvény esetén pedig a kimenet 0.5 küszöbértékének átlépése ne feltétlenül a nulla bemeneti vektornál, hanem a kívánt leképezésnek megfelelő bemeneteknél történjen.



3.1. ábra. Egy perceptron sematikus ábrája.

Az egyszerű felépítésű perceptron érdekessége abban rejlik, hogy a kívánt működést úgy sajátítja el, hogy az összetartozó be-és kimeneti értékek alapján egy tanító algoritmus segítségével a súlyokat és a bias értéket változtatva kialakítható a megfelelő átvitel. Ezáltal a bemenetére kerülő adatokban a neuron önmaga próbál valamilyen hasonlóságot felfedni [11]. Egy logisztikus függvényt aktivációs függvényként használó perceptron tanulása során a cél a súlyokból kialakított súlyvektor meghatározása úgy, hogy az egyik osztálybeli bemeneti adatokra a lineáris kombináció eredménye +1-hez, míg a másik osztálybeli pontokra 0-hoz közelítsen. Ha a perceptron kimenete átlépi a 0.5 értéket, akkor az egyik, alatta pedig a másik osztályba soroljuk az adatokat. Egy-egy tanító lépés során a súlyvektort olyan módon változtatjuk, hogy a módosított súlyvektorral a perceptron kimenete a helyes értékhez közelítsen [11].

3.2.2. Hálózat topológiája, MLP

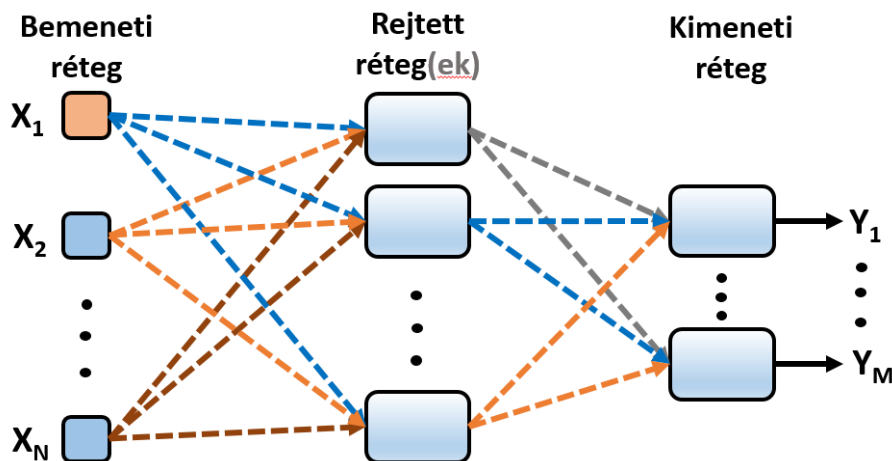
Mivel a perceptron csak lineárisan szeparálható problémák megoldására képes, így a történelem során a neurális háló tudományterületének fejlődése a 60-as évek kezdeti sikerei után megakadt. Az áttörést csaknem 20 évvel később az elemi neuronok hálózatba kapcsolásának első optimalizálási feladatok során tapasztalt sikerei, az Hopfield-háló [14] és a 3.3. fejezetben bemutatott ún. hibavisszaterjesztéses (*backpropagation*) tanulási algoritmus [15] hozta meg.

Az elemi neuronok hálózatba kapcsolásával kapott struktúra a hálózat topológiája. A neuronok összeköttetéseit, a bemenetek és kimenetek helyét általában egy irányított gráffal reprezentáljuk, melyben a csomópontok a neuronok, az összeköttetések pedig a gráf élei, ahogy ez a 3.2. ábrán is látható. A hálózat súlyainak értékét a gráf éleihez társítjuk.

A neurális háló alkalmazási területein számos hálózat topológiát alkalmaznak. A következőkben így csak a munkám során fontos és a gyakorlatban is igen elterjedt többrétegű hálózatokat (*multilayer perceptron, MLP*) mutatom be. Az MLP rétegekbe szervezett neuronokból áll, általában tetszőleges számú réteget használhat. A rétegek között megkülönböztetünk bemeneti réteget (*input layer*), rejtett réteg(ek)et (*hidden layer*) és kimeneti réteget (*output layer*) [11]. A bemeneti réteg buffer jellegű neuronokból áll, információ feldolgozást nem végez, feladata, hogy a háló bemeneteit eljuttassa a következő réteg bemeneteihez. Fontos megjegyezni, hogy az egyes rétegek neuronjaihoz tartozó súlyok egy mátrixba foghatók össze, ahol a mátrix egyes sorai az egyes neuronok súlyaiból képezett vektorok. Az későbbiekben az Igor Pro programcsomagban általam vizsgált neurális hálózatok ezen súlymátrixok eltárolásával lesznek majd alkalmasak az osztályozásra az előhívási fázisban.

A 3.2. ábrán egy speciális általam is vizsgált MLP sematikus ábrája látható, mely egy bemeneti, egy rejtett és egy kimeneti réteggel rendelkezik. Az ábra bal oldalán van a bemeneti réteg, jobb oldalon a kimeneti, közöttük pedig általánosságban egy vagy több rejtett réteg lehetséges. Az információ a hálóban balról jobbra áramlik, azaz egy adott rétegbeli neuron bemenetére a tőle balra lévő rétegbeli neuronok kimenete érkezik. A többrétegű perceptron fontos tulajdonsága, hogy annak ellenére, hogy az összes neuronja használhatja ugyanazt az aktivációs függvényt, az egyes neuronokhoz ugyanazon adatok más-más súlyozással érkezhettek, ezzel módosítva az egyes aktivációs függvények bemenetét. Ezen

kívül az egyes neuronok máshogy reagálhatnak egy esetleges ugyanolyan bemenő értékre is, hiszen minden neuronnak saját bias értéke lehet, amely különböző módon eltolja a bemeneteket.



3.2. ábra. Egy többrétegű perceptron (MLP) sematikus ábrája.

3.3. Neurális hálózat tanulása adatkból

Az élő szervezetek tanulásához hasonlóan beszélhetünk gépi tanulásról is, melynek során egy gép, mint tanuló rendszer, a környezetből nyert ismeretek alapján javítja a teljesítőképességét [11]. A gépi tanulás alapvetően nagyon sokféle lehet. A szakirodalom megkülönböztet többek között ellenőrzött tanulást (*supervised learning*), félig ellenőrzött tanulást (*semi-supervised learning*) és nemellenőrzött tanulást (*unsupervised learning*). Ellenőrzött tanulás esetén rendelkezésre állnak összetartozó be- és kimeneti tanító mintapont párok, melyek alapján a háló megtanulja a bemenet-kimenet leképezést. Félig ellenőrzött és nemellenőrzött tanulás esetén már részben vagy egyáltalán nem állnak rendelkezésre az adott bemenetekhez tartozó kívánt válaszok.

A tanítási folyamat előkészítéseként a hálózat méretének megválasztása után a hálózat súlyainak inicializálása a feladat. Megfelelő súlybeállításokkal jelentősen gyorsíthatjuk a tanulást. Előzetes ismeretek hiányában azonban a súlyok véletlenszerű inicializálása a leginkább célravezető.

3.3.1. Az alkalmazott tanítási stratégia

Munkám során az alkalmazott MLP háló tanítása ellenőrzött módon ún. hiba-visszaterjesztéses (*back-propagation*) algoritmussal történik. A neurális hálózat a valós, modellezni kívánt rendszer alapján előkészített bemenet-kimenet párokból álló tanító halmaz felhasználásával alakítja ki a megfelelő leképezést a tanulási folyamat során. Ezt tekinthetjük egy modell illesztési feladatnak is. A tanítás során a cél, hogy a tanító adatokat végigfuttatva a hálózaton a generált kimenet és az elvárt kimenet közötti különbség minél kisebb legyen. Ezen eltérést az ún. hibafüggvénnyel (*error function*) vagy más néven költségfüggvénnyel (*cost function*) vagy veszteségfüggvénnyel (*loss function*) jellemezhetjük. A számos lehetséges hibafüggvény közül a neurális hálózatok esetén legelterjedtebb a négyzetes hibafüggvény, melyet a neurális hálóim is alkalmaztak.

A hiba-visszaterjesztéses algoritmus egy pillanatnyi gradiensen alapuló iteratív eljárás, melynek részletes bemutatása meghaladja jelen dolgozat kereteit. Röviden azonban elmondható, hogy az algoritmus a tanító adatokon iterálva a kapott kimeneti értékekből származó hiba alapján módosítja a hálózat súlyait úgy, hogy csökkentse a hibát. A hiba-visszaterjesztéses algoritmus a nevét is innen kapta: a kimeneti rétegre számított hibát az egyre felsőbb rétegekre terjeszti vissza úgy, hogy a hiba skálázódik a megfelelő súly értékekkel. Az algoritmus belső paramétere a tanulási ráta (*learning rate*), mely nagy mértékben befolyásolja a tanulás sebességét, stabilitását. Túlságosan kicsi learning rate esetén a súlyok csak kis mértékben változnak, lassul a konvergencia, míg túl nagy ráta esetén az algoritmus instabillá válhat, oszcillálhat [16]. A tanulási ráta meghatározására nincs egyértelműen javasolható egyszerű módszer, értékét általában tapasztalati úton határozzák meg [11].

3.3.2. A hálózat általánosító képessége, validáció

A neurális hálózat tanításánál a tanítópontokat többször, ismételten használjuk fel, és a folyamatot addig folytatjuk, míg az eljárás magától le nem áll vagy megfelelően kis hibát nem érünk el [11]. Fontos megjegyezni, hogy a tanuló eljárásnak nem célja az, hogy a háló a véges számú tanítóponton eredményezzen helyes válaszokat, hanem az a lényeg, hogy a háló megfelelő általánosítóképességgel (*generalization capability*) rendelkezzen.

Ahhoz, hogy a modell általánosító képességét jellemezni tudjuk, szükség van egy ún. validációs halmazra (*validation set*) vagy más néven kiértékelő mintakészletre. Ez olyan

bemeneti és kimeneti mintapárokat tartalmaz, melyeket nem használtunk fel a háló tanítása során. Ha a hálózat általánosító képessége jó, akkor az várható, hogy ezen sosemlátott mintákra is jó eredményt szolgáltat.

Ha a hálózatot csak a tanítóhalmazra adott válaszai alapján értékelnénk, könnyen találkozhatnánk a túltanulás (*overfitting*) esetével. Ekkor a tanító mintakészleten már nagyon kis hibával a helyes kimeneti értéket szolgáltatja a hálózat, a validációs halmazon azonban nagyobb hibával dolgozik. A jelenség magyarázata az, hogy ilyenkor a háló túlságosan rátanul a véges számú tanító adatokra, míg a közbenső pontokra adott válasza jelentősen eltérhet.

3.3.3. Keresztvalidáció

A hálózatok teljesítőképességének jellemzésére általánosan elterjedt módszer az ún. k -szoros keresztvalidáció (*k-fold cross validation*). Ennek során a rendelkezésre álló mintapontokat véletlenszerűen felosztjuk k diszjunkt részhalmazra, majd $k - 1$ részhalmaz pontjait tanításra, a maradék részhalmaz pontjait pedig kiértékelésre használjuk [11]. A tanítópontok közül mindig más-más részhalmazt kihagyva az eljárást megismételjük az összes részhalmazra. A hálózat eredő minősítése az egyes kiértékelések átlagaként kapható meg.

Munkám során az egyes hálózatok jellemzésére négyszeres keresztvalidációt (*4-fold cross validation*) használtam. Az alkalmazott validációs technika előnye, hogy a rendelkezésre álló ki- és bemeneti adatpont párok jelentős részét felhasználhatom a tanításra, miközben a hálózat eredő értékelését az összes pont alapján végezhetem. A módszer hátránya a nagyobb számításigény, hiszen így nem egy, hanem minden egyes vizsgálatkor négy darab hálózatot kell betanítani.

3.4. Hálózat konstrukciójának általános kérdései

A neurális hálózatok képességeit és tulajdonságait többek között a használt műveleti elemek tulajdonságai, a hálózat felépítése valamint a hálózat súlyainak értéke határozzák meg. Az előzőekben bemutatott MLP és a hozzá kapcsolódó architektúra kialakítása során számos szabad paraméter módosítható az alkalmazás céljainak megfelelően. A következőkben ezen paraméterek rövid áttekintése következik, különös tekintettel azokra,

melyeket a vezetőképesség-görbék osztályozására alkalmas neurális háló fejlesztése során megvizsgáltam.

Fontos hangsúlyozni, hogy a neurális hálózatok alkalmazása a mai napig fejlődő tudományág. Bizonyított tétel, hogy neurális hálózatok esetén egy aktív (bemeneteit súlyokkal összegző, a kimenetét aktivációs függvénnyel előállító) réteg is elegendő ahhoz, hogy egy folytonos függvényt tetszőleges pontossággal közelíteni tudjunk [11]. Az adott megoldandó problémára azonban gyakran nem állnak rendelkezésre ilyen tételek, melyek megszabnák például a neurális hálózat nagyságát, a tanulási ráta értékét, vagy a tanítás idejét. Az egyes problémák megoldásakor éppenezért nagy szerepe van a tervezők tapasztalatainak. A bonyolultabb képfelismerési problémák esetén például gyakran találkozunk olyan esetekkel, mint a jól ismert és gyakran alkalmazott AlexNet [17] hálózat, mely működésének hatékonysága nincs matematikailag alátámasztva, a gyakorlatban viszont nagy sikerrel számos területen használják.

Egy-egy neurális hálózat hatékony működésének és osztályozó képességének kulcsa lehet a megfelelő, a feladathoz illeszkedő bemeneti adatok megválasztása. Ezen problémát a 3.3. ábra neurális hálói alapján szeretném szemléltetni. Mindkét hálózatnak az a feladata, hogy az ábrák jobb oldalán lévő grafikonon a kétdimenziós sík X_1 és X_2 koordinátáival jellemezhető kékekkel és narancssárgával jelzett adatpontokat kék és narancssárga osztályokba sorolja. Az ábrán a neuronok kis négyzetek, melyekben a neuron által szolgáltatott kimenet látható kicsiben az egész X_1 - X_2 síkra. Az ábrán a háló neuronjai tangens hiperbolikus aktivációs függvényt használnak, a hálózat összeköttetéseinek vastagsága a zöld számokkal jelölt súlyok értékével arányos. Az egyes neuronok alsó sarkánál az adott neuronhoz tartozó bias értéket rózsaszínnel tüntettem fel. Az ábrák jobb oldalán ábrázolt grafikonon a hálózat által vizsgált teszt adatsor pontjai jelennek meg, melyek eredeti osztálycímkéje kék és narancssárga. A grafikon háttérszínei megadják, hogy a neurális hálózat a kétdimenziós sík egyes X_1 és X_2 koordinátáival jellemezhető adatpontjait minek osztályozná.

A 3.3 a) ábra bemenetére a vizsgálandó adatpontok X_1 és X_2 koordinátája kerül. Ezen adatok különféle súlyokkal a rejtett réteg neuronjainak bemeneteire kerülnek. A rejtett réteg neuronjai a súlyozott bemeneteket és a bias értékeket felhasználva tangens hiperbolikus aktivációs függvény alkalmazásával állítják elő a kimenetüket, melyek ismét különböző súlyozással a kimeneti réteg neuronjára kerülnek. A hálózat kimenete -1 és $+1$

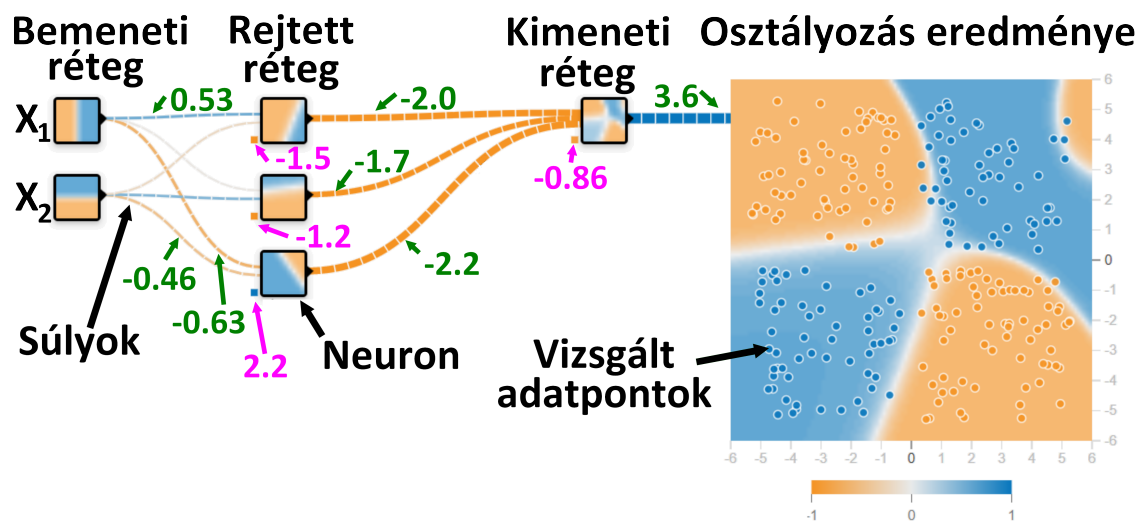
között lesz a tangens hiperbolikus aktivációs függvénynek köszönhetően. Ha a hálózat kimenete pozitív, akkor az adott adatpontot kék, ha negatív, akkor narancssárga osztályba sorolja.

Az adatpontok síkon való elhelyezkedéséből jól látszik, hogy csupán az X_1 és X_2 koordináták alapján a pontokat nem lehetne helyesen osztályozni. Ezt jól mutatja a 3.3 a) ábra grafikonjának háttérszíne, mely a hálózat kimenetét mutatja kétdimenziós sík minden egyes pontjára. Jól látható, hogy jó néhány helyen eltér az a vizsgált adatpontok és az alattuk lévő háttér színe, amely azt mutatja, hogy ezen adatpontokat a hálózat rosszul osztályozná.

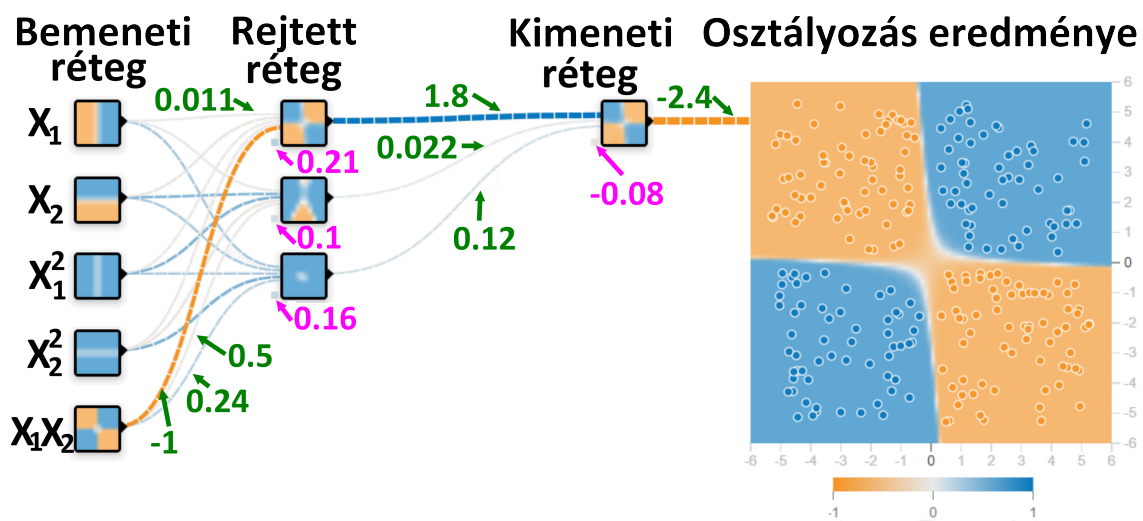
A 3.3 b) ábrán egy olyan neurális hálózat látható, mely a "nyers" adatokon, azaz a koordinátákon kívül "processzált" adatokat, azaz a koordináták négyzetét és szorzatát is megkapja bemenetként. Látható, hogy a hálózat szinte azonnal rátanul a triviális tényre, hogy az osztályozási feladatot az X_1X_2 koordináta-szorzatból azonnal pontosan meg lehet oldani. A processzált adatok alapján történő sikeresebb osztályozás azt sugallja, hogy az MCBJ-mérésekből származó vezetőképesség-görbék válogatása esetén is törekednem kell a nyers adatok helyett a vezetőképesség-görbékből olyan bemeneti adatokat kialakítani, melyek hatékonyabbá tehetik az osztályozást.

Arról, hogy mik a releváns bemeneti adatok, a hálózat súlyai is árulkodóak. A 3.3 b) ábrán X_1X_2 adat megy a legnagyobb súllyal a bemeneti és rejtett réteg között a legfelső rejtett rétegbeli neuronhoz, és szintén ebből a neuronból megy tovább a legnagyobb súllyal az információ a kimeneti réteg felé. A hálózat tanulása szempontjából irreleváns adatok, mint az X_1^2 vagy X_2^2 egészen nagy súlyokkal mennek a rejtett réteg alsó két neuronjához, azonban ezen neuronok súlya a kimeneti réteg felé már nagyon kicsi lesz. Az előzőek tükrében, ha veszünk egy bemeneti adatot, és az egyes rejtett rétegbeli neuronokhoz tartozó súlyait megszorozzuk a megfelelő rejtett rétegbeli neuron kimenetének súlyával a kimeneti réteg felé, majd ezen súly-szorzatokat összeadjuk, akkor egy olyan mennyiséget kaphatunk, mely a releváns adatpontokra nagyobb lesz, mint az irrelevánsokra. A súlyok szorzatának vizsgálatával így könnyen összehasonlíthatóak az egyes bemeneti adatok a hálózat működésében betöltött szerepük szempontjából. A későbbiekben a saját neurális hálóim döntési mechanizmusának szemléltetésénél fogom kihasználni az egyszerű, három rétegű hálózatok ezen tulajdonságát.

A neurális hálózat méretének, rétegei számának és a rejtett rétegben lévő neuronok



a)



b)

3.3. ábra. Két különböző neurális háló a Tensorflow Playground alkalmazásból ugyanazon osztályozási probléma megoldására (<https://playground.tensorflow.org/>).

számának meghatározása szintén fontos feladat, hiszen ha találunk az adott problémára egy jól működő, ám nagyon komplex neurális háló helyett egy kevesebb műveleti elemből állót, azzal számítási kapacitást és időt nyerhetünk.

Általában a neurális hálózat kialakításakor a bemeneti és kimeneti neuronok száma adott. Egy három bemenetű és két kimenetű függvény közelítésekor három bemeneti neuron és két kimeneti szükséges. Esetemben a bemeneti neuronok számát az adott vezetőképesség-görbékből előállított adatpontok száma adja meg. A későbbi 4.2. fejezetben megmutatom, hogy ezen adatpontok száma a vezetőképesség-binek száma lesz. Neurális hálóm kimenetén elég egy kimeneti neuront alkalmazni, mely megadja az adott osztályba

tartozás valószínűségét.

A hálózat rejtett rétegeiben lévő neuronok számának meghatározása már egy nehezebb feladat. Ha túl kevés rejtett rétegbeli neuront használunk, a hálózat nem fog megfelelően tanulni, túlzottan sok használatával pedig több időre és számítási kapacitásra van szükség a háló működéséhez. Általánosságban a bonyolultabb, nagyobb bemeneti adatponttal rendelkező problémák több rejtett rétegbeli neuront, esetleg több rejtett réteget igényelnek, azonban ennél több konkrétum általában nem adható meg.

A neurális hálózatok tanításakor felmerül azon fontos kérdés is, hogy meddig tanítsuk a modellt. A tanítás idejét a 3.3.1. fejezetben leírtak miatt a tanulási ráta nagyban befolyásolja. Általánosságban elmondható az, hogy a hálózatot addig tanítjuk, míg el nem érjük a kívánt vagy elérhető maximális osztályozási pontosságot úgy, hogy még elkerüljük az *overfitting* jelenségét.

4. Fejezet

Vezetőképesség-görbék osztályozására alkalmas neurális háló fejlesztése

A molekuláris elektronikai mérésekből származó vezetőképesség-görbék osztályozására közvetlenül a mérések kiértékelését végző Igor Pro szoftveren alapuló programban valósítottam meg a neurális hálókat. A fejezetben áttekintem a választás előnyeit és korlátait, valamint a neurális hálózat működéséhez megírt függvényeket. Részletesen bemutatom az optimális osztályozó hálózat megkeresésének, és az egyes paraméterek változtatásának hatását a neurális hálózat működésére.

4.1. Neurális háló az Igor Pro programcsomagban

Az MCBJ-technikával végzett mérések kiértékelése és az adatok begyűjtése a BME Molekuláris Elektronika laboratóriumában az Igor Pro szoftveren alapuló programmal történik, így a vezetőképesség-görbékre ideális lenne közvetlenül a méréskiértékelő programban megvalósítani az osztályozást. Ennek előnye nemcsak az adatok gyors elérése, hanem az is, hogy egy jól működő neurális hálóval akár a mérések közben is jósolhatjuk a molekulák beépülési arányát, a mérések sikerességét.

Az Igor Pro program két beépített neurális hálózatok kialakításához használható függvénnyel rendelkezik, ezek a *NeuralNetworkTrain* és *NeuralNetworkRun* függvények.

A *NeuralNetworkTrain* egy a 3.2. ábrához hasonlóan három rétegű, a kimeneti rétegben egy neuront tartalmazó neurális hálózat megalkotására képes, melyben minden neuron aktivációs függvénye a logisztikus függvény. A függvény a bemenetén olyan adatmátrixot vár,

melynek minden sora megfelel egy tanítási eseménynek (egy-egy vezetőképesség-görbe) és az oszlopok száma pedig a bemeneti értékek számával egyezik meg. A bemeneti adatmátrix értékeit $[0,1]$ intervallumra kell skálázni, illetve a neurális hálózat kimenete is ezen intervallumba eső érték lesz. A függvény bemeneteként az ellenőrzött tanulás megvalósításához a kérdéses osztálycímkeket tartalmazó adatsort is meg kell adni. A hálózat a 3.3.1. fejezetben bemutatott hibavisszaterjesztéses algoritmust használja a tanításhoz, melynek során négyzetes hibafüggvényt minimalizál. A tanulás eredményeképp a függvény a létrejött neurális hálózat súlyait két súlymátrixban ($M_Weights1$ és $M_Weights2$ néven) tárolja el, az egyik mátrixban a bemeneti és hidden layer közötti súlyok, míg a másikban a rejtett réteg és a kimenet közötti súlyok találhatók.

A *NeuralNetworkRun* függvény a bemenetén ugyanúgy $[0,1]$ közé eső értékekkel rendelkező teszt adatsort vár, majd betölti a betanított neurális hálózathoz tartozó súlymátrixokat. A függvény a teszt adatsorra megadja a kérdéses címkeváltozó értékét.

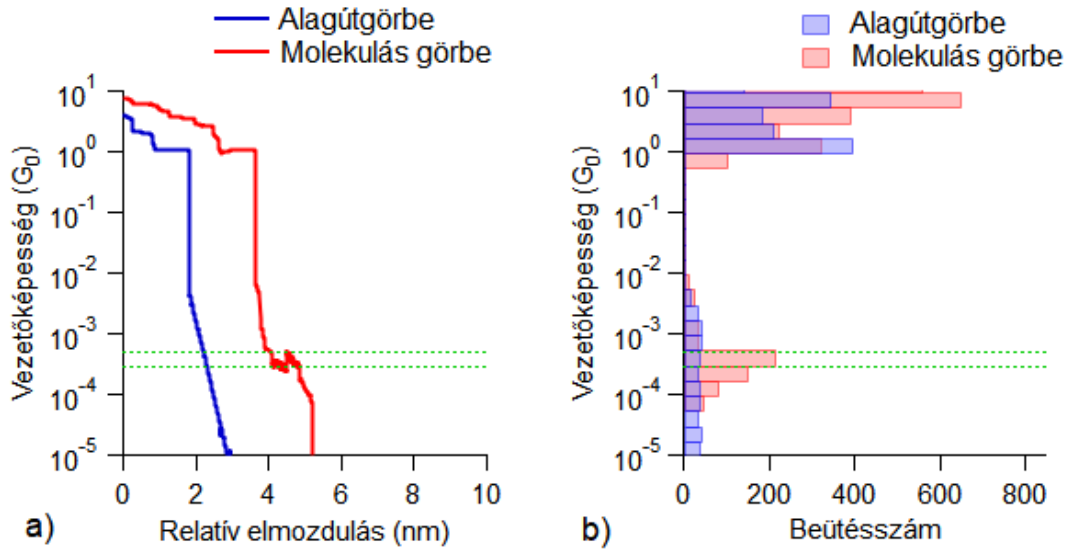
A fentiekből látható, hogy a programcsomag beépített függvényei már sok paramétert előre megszabnak a neurális hálózat felépítésére és működésére vonatkozóan. Ez akkor lehet hátrány, ha az adott probléma megoldásához bonyolultabb, esetleg más tanulási algoritmust használó hálózatot szeretnénk kialakítani. Mint a későbbiekben megmutatom, számomra a vezetőképesség-görbék osztályozására a programcsomag beépített függvényei által nyújtott lehetőségek megfelelőek voltak. Fontos hangsúlyozni, hogy az adott osztályozási probléma feltehetően számos más neurális háló architektúrával is megoldható lenne, melyeket jelen dolgozat keretein belül nem vizsgáltam.

4.2. Adatelőkészítés

Munkám során a neurális hálózatokat tanító, tesztelő függvényeket írtam, melyek fontos része volt a 4.1. fejezetben bemutatott függvények megfelelő bemenetének előállítására, értelmezésére. A 3.3.3. fejezetben bemutatott keresztvalidációt is magvalósítottam, melynek menetét, illetve a hálózat teljesítőképességének értékelési módját is szeretném a következőkben bemutatni.

A mérési adatok a 2.2 a) és 4.1 a) ábrán látható vezetőképesség görbék, melyekből a 3.4. fejezetben leírtak alapján olyan bemeneti adatokat kell megalkotni, melyek jól reprezentálják a megoldandó osztályozási feladatot. Ez önmagában egy nehéz kérdés, hiszen

nem tudjuk előre, hogy az osztályozó hálózat mi alapján fogja szétválogatni a görbéket. Ha túl sok bemeneti adatpontot adunk meg a hálózatnak, például az összes mérési adatpontot, akkor egyrészt a nagyon nagy méretű bemenet nagyon nagy számítási igényt von maga után, illetve könnyen előfordulhat az is, hogy a hálózat a mérési adatok zajossága miatt nem képes jól osztályozni.



4.1. ábra. a) Logaritmikusan ábrázolt alacsony hőmérsékleten mért vezetőképesség-görbék. b) Vezetőképesség-görbékből külön-külön készített egyedi, egydimenziós hisztogramok.

Kézenfekvő megoldásnak a 4.1 b) ábrán bemutatott egy-egy vezetőképesség-görbéből külön-külön elkészített hisztogram adódik. Az ábrán a vezetőképesség tartományt logaritmikusan ekvidisztánsan szakaszokra, ún. binekre osztjuk és megszámloljuk az egy binbe eső mérési adatpontok számát. Szemléltetésképpen a 4.1 a) és b) ábrán szaggatott zöld vonalakkal bejelöltem egy ilyen vezetőképesség-bint. A neurális háló bemeneti adatpontjainak száma meg fog egyezni a vezetőképesség-binek számával. A neurális háló bemenetére jutó adatpontokat jelölhetjük $N_i(r)$ -el is, amely az r -edik görbén található i -edik vezetőképesség-binben lévő pontok száma. Fontos, hogy ez nem egyezik meg az $\langle N_i(r) \rangle$ -el, amely az összes görbére kitérítve a teljes hisztogram, melyet a 2.2 b) ábrán láthatunk. A hisztogram készítésekor elveszítjük ugyan a mérési adatpontok időbeli sorrendjét, azonban a neurális háló bemenetére minden egyes vezetőképesség-görbéből jövő információt jóval egyszerűbb formában tudjuk továbbítani. A vezetőképesség binek számának megválasztása ismét egy szabad paraméter, melyet a következő fejezetben meg is

vizsgálók.

Az előző fejezetben bemutatott beépített függvények bemenetére $[0,1]$ intervallumra skálázást a hisztogramokkal könnyen meg lehet oldani. Lehetne úgy is skálázni, hogy az egyes adatsorokban a hisztogram maximális értékével leosztok, azonban ezzel a különböző adatsorokra elvégzett számolások nem lesznek összehasonlíthatóak. Ezen módszer helyett olyan megoldást választottam, amely minden adatsorra egységes skálázást eredményez. A módszerem azon alapul, hogy az alagútgörbe alapján történő kalibrációval számolva a mintavételezési sebességből és a piezo mozgató mozgatási sebességéből az adott mérési pontokat távolsággá tudjuk váltani. Ha a vizsgálandó adatsorokra van egy maximális platóhosszúság, akkor ki lehet számolni az ehhez tartozó elmozdulás értékét, majd az ezen elmozdulás alatti beütésszámmal leosztani a hisztogramokat. A legnagyobb beütésszámot a vezetőképesség-hisztogramokban tipikusan $1 G_0$ -nál látjuk, ahol atomi láncot is lehet húzni [18]. A tapasztalataink alapján legfeljebb körülbelül 7 atom hosszú láncot tudunk húzni az MCBJ mérések során, amely akár 2 nm-es elmozdulást is eredményezhet. Esetemben ennek a kétszeresét, 4 nm-t választottam referenciának és az ezen elmozdulás alatti mérési adatpontok számával osztom le a számolt hisztogramokat.

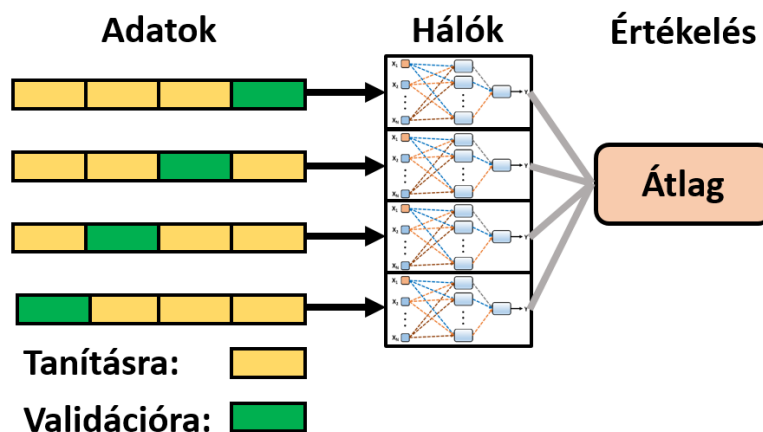
Az adatelőkészítés másik lényeges mozzanata a tanító adathalmaz megválasztása. Ennek véghezvitelénél is számos lehetőség kínálkozik, hiszen korábbi osztályozó algoritmusok eredményeként kapott osztályozott görbéket is felhasználhatnák. Számomra azonban fontos volt, hogy a tanító adatok egyértelműek legyenek, és biztosan a megfelelő osztályba tartozzanak, ezért a tanító adathalmaz létrehozására a görbék kézi válogatását választottam. A következő, 4.4. fejezetben bemutatott hálózatok esetén a tanító adathalmaz alacsony hőmérsékletű MCBJ-mérésekből származó, kézzel válogatott 320 görbe.

A tanító adathalmaz megadásakor rögtön felmerül a kérdés a tanító adathalmaz méretére vonatkozóan. A korábban hivatkozott K. Lauritzen és csapata hozzám hasonlóan molekulákat osztályozó rekurrens neurális hálózathoz azt találta, hogy 60-90 tanító adatsor elég a hálózat megfelelő működéséhez. A tanító görbék számát 90 fölé növelve nem tapasztaltak érdemi javulást, de romlást sem a háló működésével kapcsolatban (lásd: [4] Support Material). A hálózat kialakítása során magam is azt tapasztaltam, hogy az osztályozó ugyanezen görbeszámoknál már jól működik, azonban általában minden esetben az eredetileg kiválogatott nagyobb görbeszámmal dolgoztam.

4.3. Keresztvalidáció megvalósítása

A 3.3.3. fejezetben bemutatott keresztvalidáció megvalósítását a méréskiértékelő programban függvények megírásával automatizáltam. A rendelkezésre álló tanító adatokat a 4.2. ábrán vázolt módon ugyanolyan molekula-alagútgörbe arányú részekre osztom, majd a különböző tanító halmazokkal négy különböző, ám ugyanolyan paraméterekkel rendelkező neurális hálózatot tanítok be. A hálózatokat ez után a validációs halmazon tesztelem.

A hálózatok minősítésére legtöbb esetben a pontosság (*accuracy*) fogalmát használtam. Ezen mérőszám megadja, hogy az összes görbének hány százalékát osztályozta a modell helyesen. Számomra ezen mérőszám jellemezte legjobban a modell teljesítőképességét, azonban természetesen számos más mérőszámot is figyeltem az osztályozó modellek kiértékelése során. Ilyen például a szenzitivitás, vagy más néven érzékenység (*sensitivity*), vagy a precizitás (*precision*), mely mennyiségeket mindkét típusú görbére számoltam. A molekulára való érzékenység például megadja, hogy az összes molekulás görbe közül mennyit osztályoztunk helyesen molekulának, míg a molekulára való precizitás azt határozza meg, hogy a molekulaként osztályozott görbék hányad része valóban molekula.



4.2. ábra. A négyes keresztvalidáció megvalósításának sematikus ábrája.

Az általam alkalmazott négyes keresztvalidáció után a kapott négy neurális hálózatot az átlagos pontosságukkal értékeltem. A módszer előnye, hogy a következő fejezetben bemutatott paraméterek változtatásával kapott pontosság értékek mögött mindig négy különböző, ám ugyanazon paraméterekkel betanított neurális hálózat rejlik.

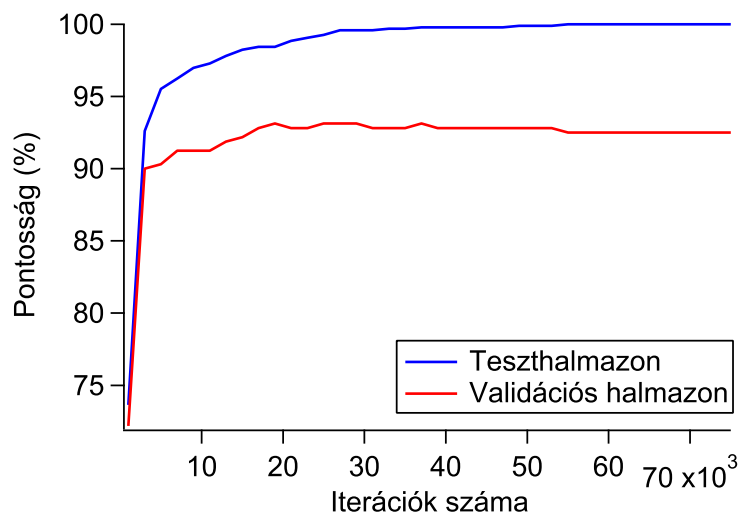
4.4. Optimális paraméterek keresése

Az adott paraméterbeállítások mellett a keresztvalidációt, a neurális hálózat tanítását és tesztelését végző programkódomhoz a neurális hálózat optimális paramétereinek megkeresése érdekében olyan ciklusokat írtam, melyek adott paramétereket egy-egy intervallumban változtatva automatikusan tanítanak be neurális hálózatokat.

4.4.1. Iterációk számának hatása

Elsőként azt vizsgáltam, hogy milyen hatása van a neurális hálózat tanítása során az iterációk számának. Egy-egy iteráció során a tanító halmaz hálózaton történő végigfutása hatására a háló súlyai frissülnek a hibavisszaterjesztéses algoritmusnak megfelelően. Az iterációk száma természetesen nagyban függ a tanulási ráta értékétől. A 3.3.1 fejezetben leírtak miatt a tanulási rátát igyekeztem elég kicsinek választani ahhoz, hogy a hálózat tanítása ne váljon instabillá. A hálózat tanításakor a hibafüggvény értékét az egyes iterációk után figyelve a 0.005 értékű tanulási rátát választottam, mely esetén a hibafüggvény értéke folyamatosan minden iteráció után kissé csökkent, de nem oszcillált.

A tanulási ráta megválasztása után megvizsgáltam, hogy az adott ráta mellett a maximális teljesítőképesség eléréséhez hány iterációra van szükség, ahogy azt a 4.3. ábra mutatja. A vizsgálat során a vezetőképességet a 10^{-5} és $3 G_0$ tartományon, 100 binre osztottam, 150 rejtett rétegbeli neuront használtam.



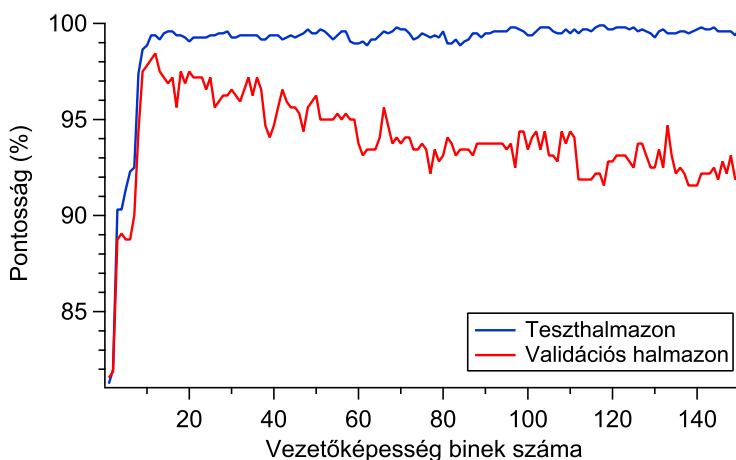
4.3. ábra. Az iterációk számának hatása a neurális hálózat pontosságára.

Az eredmények azt mutatják, hogy 30 000 iteráció elég a hálózat betanításához. Az

iterációk számának további növelésével a tanító adatokat a hálózat már 100 %-osan be-tanulja, a validációs halmazon azonban már nem javul, sőt kis mértékben csökken a tel-jesítőképesség, amely az overfitting jelenségével magyarázható.

4.4.2. Bemeneti adatpontok számának hatása

A következő problémakör a vezetőképesség-binek számának hatása a neurális háló-zat működésére. Előzetesen is azt várjuk, hogy a túl sok bemeneti információval a zajra is rátanulhat a háló, így nem érdemes a vezetőképesség-görbékből készített egydimenziós hisztogramot túlságosan sok binre osztani. A túl kevés binszám pedig a 2.5. fejezetben be-mutatott problémához hasonlóan megnehezítheti a hálózat megfelelő döntéshozatalának kialakítását. Az előzőhöz hasonlóan 10^{-5} és $3 G_0$ vezetőképesség-tartományon vizsgál-va, 150 rejtett rétegbeli neuronnal 0.005-ös tanulási rátával és az előző vizsgálat alapján 30000 iterációval a vezetőképesség-binek számának hatására a 4.4. ábrán látható ered-ményt kaptam. Fontos hangsúlyoznom, hogy a vezetőképesség-binek számának változta-tásakor mindig ugyanazon vezetőképesség-intervallumot osztottam fel, így a binek mérete változott.



4.4. ábra. A bemeneti adatpontok számának hatása a neurális hálózat teljesítő-képességére.

A bemeneti adatpontok számával megegyező vezetőképesség-binek száma tehát nagy-ban befolyásolja a hálózat teljesítőképességét. A tanító és validációs adatoknak egyaránt 62,5 %-a molekula. Ha a hálózat a tanító adathalmaz alapján egyszerűen az összes görbére a többségi címkeváltozót tippelné, akkor is ilyen százalékban találnánk el a molekulákat. Ezen értékre a későbbiekben véletlen találati arányként fogok hivatkozni. A 4.4. ábra

azonban jól mutatja, hogy még 1-2 bin esetén sem csökken le ilyen alacsonyra a pontosan felismert görbék aránya. Ez annak köszönhető, hogy a teljes vezetőképesség-tartományon mért beütésszám néhány vezetőképesség-görbénél elegendő a molekula vagy alagútgörbe osztályokba soroláshoz.

A vizsgálat eredménye, hogy ezen speciális osztályozási feladatnál 12 vezetőképesség-bin esetén lesz a legpontosabb a hálózat predikciója.

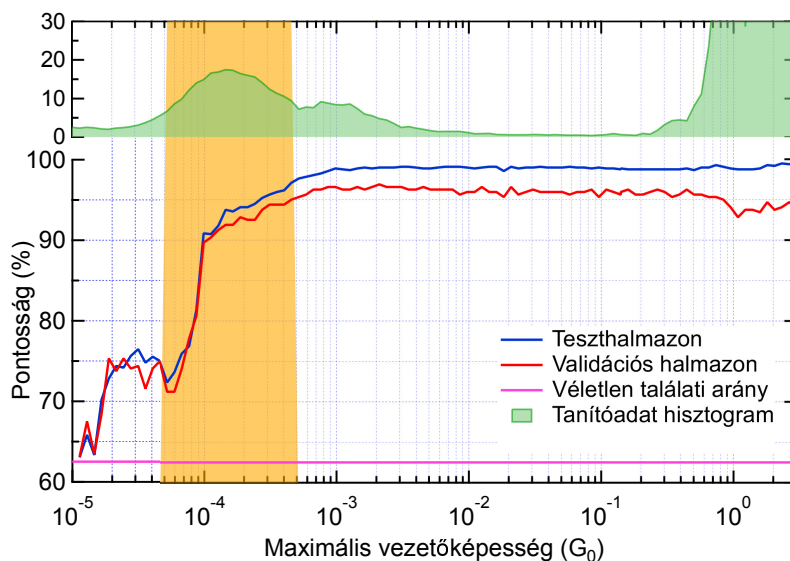
4.4.3. Vezetőképesség-tartományok megválasztása

Az eddigi vizsgálatok során minden esetben a 10^{-5} és $3 G_0$ vezetőképesség-tartományon tanulmányoztam a hálózatok működését. Azért esett erre a szakaszra a kezdeti választás, mert ez a nagy intervallum tartalmazza az egyatomos nanovezetékhez tartozó $1 G_0$ -os platót, minden molekula beépülést jelző platót, valamint alsó határa pedig két nagyságrenddel a mérési zaj értéke felett van. Érdekes kérdés, hogy a háló működése szempontjából például szükségünk van-e az $1 G_0$ -os plató ismeretére. Összefügg-e ez a molekula beépülésével? A vezetőképesség-intervallum alsó határát meddig érdemes növelni? Az alábbiakban bemutatom ezen vezetőképesség-tartományok megválasztásának hatását.

A vezetőképesség maximális értékét a következőkben úgy csökkentettem, hogy a vezetőképesség alsó határát fixen $10^{-5} G_0$ -os értéken hagyva a kezdeti 100 bines vezetőképesség-intervallumból mindig lecsíptem egy-egy bint. Ezzel tehát nemcsak a vezetőképesség-maximum értéke, hanem a vezetőképesség-binek száma is csökkent, viszont a binek nagysága nem változott. A módszer előnyös tulajdonsága így az volt, hogy a különböző hálózatok ugyanazon információ darabkákból kaptak meg mindig egyre kevesebbet. A tanulási rátát ismét 0,005-nek, az iterációk számát pedig 30000-nek választottam.

A 4.5. ábrán látható vezetőképesség-maximum csökkentésével kapott eredmények azt mutatják, hogy az $1 G_0$ -os platónak a hálózat döntése szempontjából nincs kitüntetett szerepe, sőt elhagyásával a hálózat osztályozási pontossága nő. Az ábra felső részében zölddel felvázoltam a tanítóadatok hisztogramját is, sárgával pedig kijelöltem azon tartományt, ahol a legtöbb molekuláris beütésszám várható. A 4.5. ábra jól mutatja, hogy a molekuláris tartomány elhagyásával a hálózat teljesítőképessége drasztikusan leromlik, azonban még sokáig így sem csökken le a véletlen találati arány szintjére. Ennek az lehet a magyarázata, hogy a molekuláris kontaktus elszakadásakor a vezetőképesség ugrásszerűen lecsökken, ezáltal minimális beütésszámot eredményezve ezen vezetőképesség-tartományon.

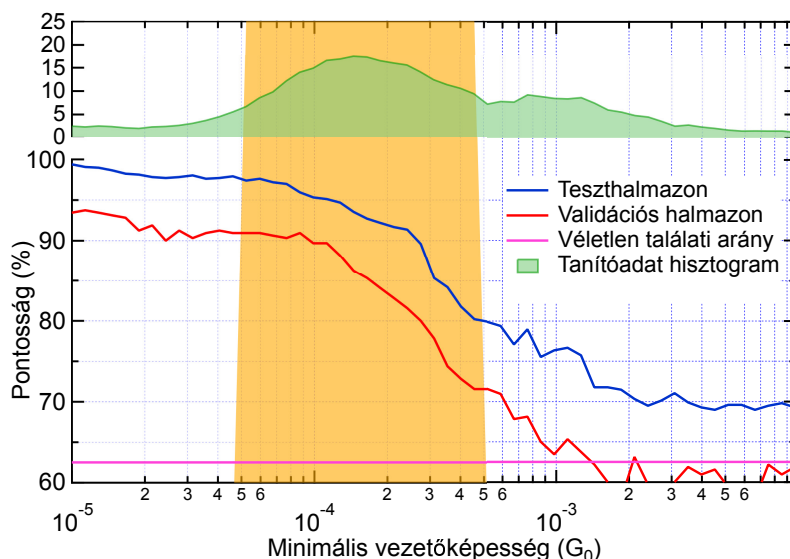
Ezzel szemben az alagút-típusú görbénél az alagútáram hatására a vezetőképesség exponenciális távolságfüggést mutat ezen alacsony vezetőképesség-tartományokban, amely a molekuláris görbénél nagyobb beütésszámot eredményez. Ezen görbe-alakok a maximális vezetőképességet a molekuláris tartomány alá csökkentve is lehetővé tesznek egy egészen pontatlan, de a véletlen találatnál jobb osztályozást.



4.5. ábra. Az osztályozási pontosság változása a maximális vezetőképesség csökkentésével.

A vezetőképesség minimális értékének növelését az előzőekhez hasonlóan vizsgálva a 4.6. ábra szemlélteti. Ha a vezetőképesség maximális értékét fixen $3 G_0$ -on tartva a minimális vezetőképesség-értékét növelem, akkor a hálózat a molekuláris tartományt elhagyva már nem képes a vezetőképesség-görbéket jól osztályozni.

Ezen eredmény ismét megerősít abban, hogy a magasabb vezetőképesség-tartományok nem hordoznak információt a molekula beépülésére vonatkozóan. Ez a vizsgálataim érdekes következménye, hiszen előfordulhatna az is, hogy a magasabb vezetőképesség-régiókban bekövetkező változások nagy hatással vannak arra, hogy a későbbiekben be tud-e épülni a kontaktusba egy molekula vagy sem.



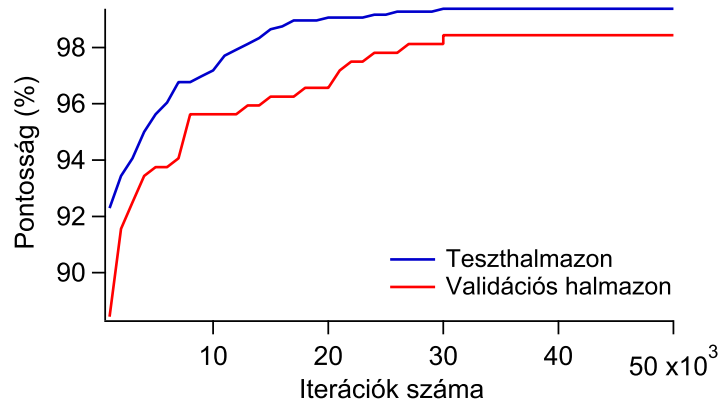
4.6. ábra. Az osztályozási pontosság változása a minimális vezetőképesség-érték növelésével.

4.4.4. A legegyszerűbb, 12 bines modell

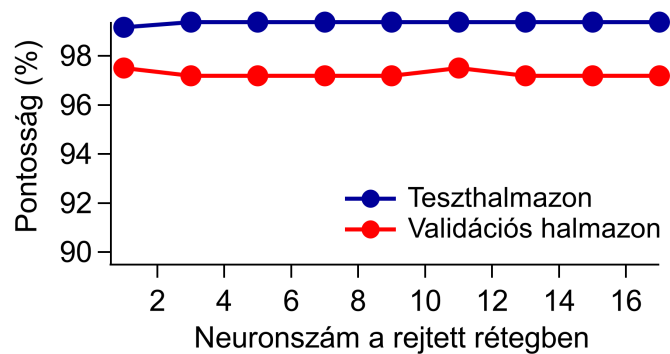
A munkám során egy egyszerű, jó általánosító képességgel rendelkező modellre törekedtem. A korábbi vizsgálatok eredményeképp ezért egy 12 bemeneti értéket használó hálózatot alakítottam ki, 18 neuronnal a rejtett rétegben. A vizsgált vezetőképesség tartományból a 4.5. ábra tanulságai alapján elhagytam az $1 G_0$ tartományát és a hálózat által vizsgált vezetőképesség-intervallumot $10^{-5} G_0$ és $0,5 G_0$ közötti sávra választottam.

A 4.7. ábrán látható módon ezen egyszerű hálózatra is megvizsgáltam az iterációk számának hatását, de az eddigiekhez hasonlóan azt tapasztaltam hogy ezen hálózat tanulásához sem túlságosan sok a 30000 iterációs lépés.

A 4.8. ábrán pedig azt is megfigyeltem, hogy a 12 bemenettel rendelkező hálózat teljesítőképessége nem függ a rejtett rétegben lévő neuronok számától. Ez azért érdekes eredmény, mert azt mutatja, hogy már a vezetőképesség-értékek egyetlen középső rétegbeli neuron segítségével történő súlyozása esetén, igen magas, 97 %-os pontosság körüli osztályozást tudunk megvalósítani. Ezen eredmény arra utal, hogy ha több rejtett rétegbeli neuront alkalmazunk, akkor előfordulhat, hogy ezen neuronok "ugyanazt tanulják meg". A hálózat döntési mechanizmusát a 5. fejezetben vizsgálom meg alaposabban.



4.7. ábra. 12 bemenettel rendelkező neurális háló pontossága az iterációk számának függvényében.



4.8. ábra. 12 bemenettel rendelkező neurális háló pontossága a rejtett rétegben lévő neuronok számának függvényében.

Fontos megjegyezni, hogy az a tény, hogy a hálózat a validációs halmazon ilyen jó eredményt ért el, nem jelenti azt, hogy az összes mérési adaton is jól fog teljesíteni. A tanításhoz, illetve validáláshoz használt előre osztályozott adatsorok olyan görbéket tartalmaznak, melyekre vizuálisan is könnyen megállapítható, hogy melyik csoportba tartoznak. Ugyanakkor az összes mért görbe tartalmaz olyan mérési adatokat is, amelyekre nem ennyire egyértelmű az osztályozás. A következő fejezetben megvizsgálom, hogy hogyan teljesít a neurális hálózatom az összes mérési adaton.

4.5. Neurális hálós modellek tesztelése

A következőkben a neurális hálózatos modelleknek szeretném bemutatni egy másik kiértékelési módját is. Eddig a hálózatok teljesítőképességének kvantitatív vizsgálatát

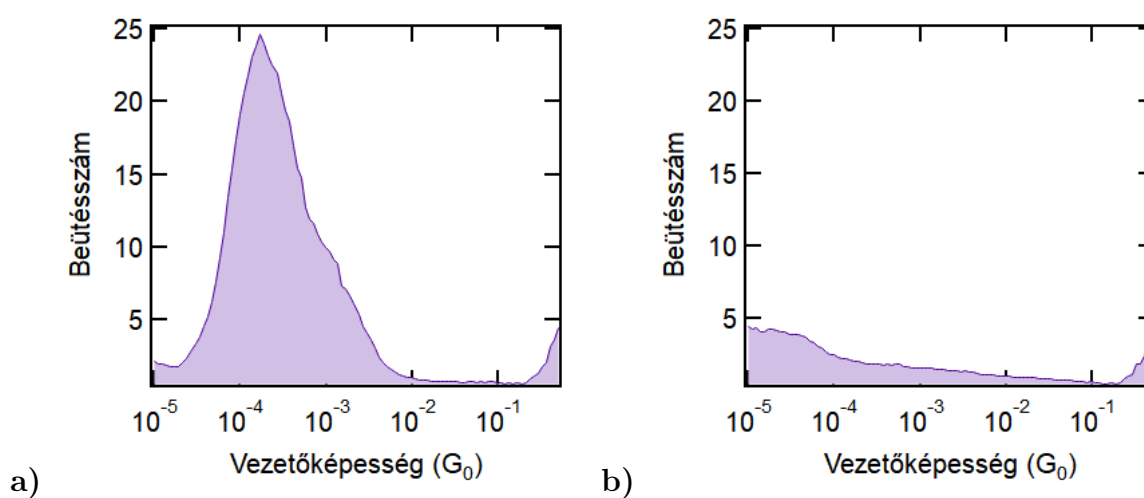
hajtottam végre, melyhez rendelkezésre álltak előre "legyártott" osztálycímkével ellátott adatok, a hálózat osztályozási pontosságát így a helyesen osztályozott görbék arányával számszerűsíteni tudtam.

A vezetőképesség-hisztogramok jellegzetességei és a nagyszámú mérési adat azonban lehetővé teszik, hogy a hálózatot ismeretlen osztálycímkéjű adatokra futtatva, kvalitatíven lehessen minősíteni. Ezen vizsgálatra egy, előző (4.4.4.) fejezetben bemutatott, 12 vezetőképesség-bin, 18 rejtett rétegbeli neuront és 35000 iterációs lépést alkalmazó hálózatot választottam ki és tanítottam be a korábbiakban is használt 320, alacsony hőmérsékleten mért tanító adat keresztvalidációjával.

4.5.1. Modell tesztelése alacsony hőmérsékletű adatokon

A betanított, 12 vezetőképesség-bin alapján osztályozó hálózatot futtattam a tanító adatokhoz hasonlóan alacsony hőmérsékleten mért MCBJ-mérésekből származó 6000 ismeretlen osztályba tartozó görbére (4.10. ábra). A hálózat 3982 darab görbét alagút-, 2018 darabot pedig molekula osztályba sorolt. Pontosságát természetesen nem tudjuk megmondani, viszont teljesítőképességére vonatkozóan árulkodóak a 4.9. és 4.11. ábrák.

A molekula osztályba sorolt görbékből egydimenziós hisztogramot készítve jól látható a molekulákra jellemző csúcs, míg a alagútgörbének osztályozott görbékből készített hisztogramon ez egyáltalán nem jelenik meg (4.9. ábra).

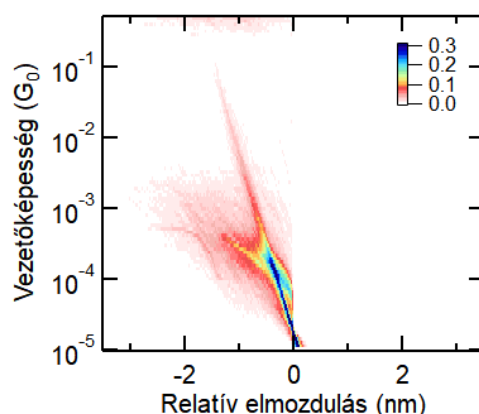


4.9. ábra. a) Molekulának osztályozott vezetőképesség-görbék, b) alagút-típusúnak osztályozott vezetőképesség-görbék egydimenziós hisztogramja.

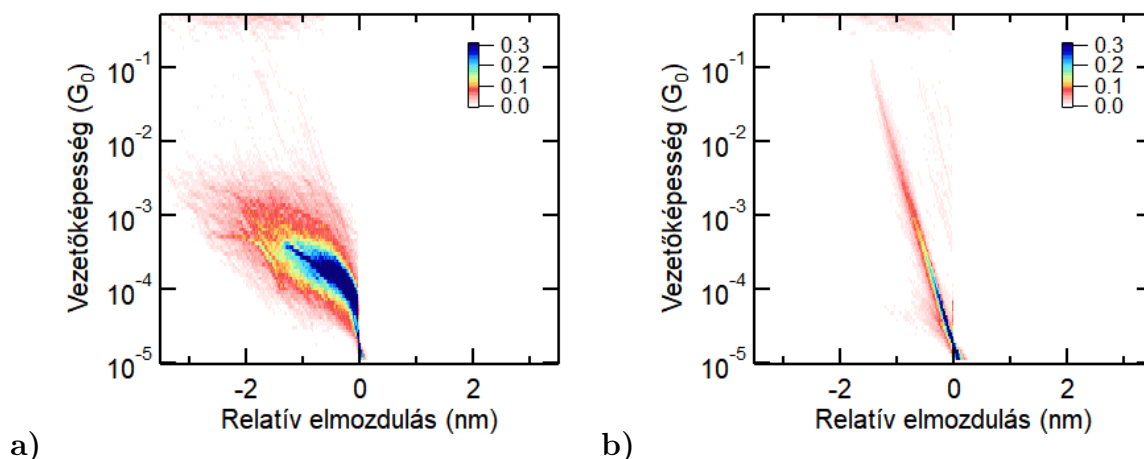
Az osztályozás minőségét talán még jobban szemléltetik a 4.11. ábra összetolt, két-dimenziós hisztogramjai. Ezen ábrák úgy készülnek, hogy a vezetőképesség-görbéket egy adott vezetőképesség-értéknél összetoljuk, majd a vezetőképességet és a relatív elmozdulást is intervallumokra osztva kétdimenziós hisztogramot számolunk. A vezetőképesség értéket, ahol az összetolást végezzük, úgy választottuk meg, hogy jól láthatóvá váljanak a különböző típusú görbék. Esetemben a görbéket $2 \cdot 10^{-5} G_0$ -nál összetolva az alagút-típusú görbék egy egyenesre esnek, ezáltal a kétdimenziós hisztogramon jól megkülönböztethetőek a laposabb platót eredményező molekulás görbéktől. Egy ilyen kétdimenziós hisztogram látható a 4.10. ábrán is, ahol az összes 6000 darab vizsgált görbéből készítettünk hisztogramot.

A 4.11 a) ábrán a molekulának osztályozott görbék esetén a molekulás vezetőképesség-görbékhez hasonlóan a molekuláris tartományban nagyobb beütésszám jelenik meg, míg ez a jelenség a 4.11 b) alagútgörbének osztályozott görbéi esetén a várttal egyezően nem látható. Az alagút-típusba sorolt görbék esetén az alagútáramra jellemző exponenciális távolságfüggés is szépen kivehető.

A 4.11. ábra alapján elmondható, hogy kvalitatívan, 6000 vezetőképesség-görbére megvizsgálva a neurális hálóval egy jól osztályozó modellt sikerült megvalósítanom. A következő fejezetben megvizsgálom ugyanezen neurális hálós modell teljesítőképességét magas hőmérsékleten mért adatokra is.



4.10. ábra. Az osztályozás során vizsgált összes vezetőképesség-görbe (6000 db) összetolt, kétdimenziós hisztogramja.



4.11. ábra. a) Molekulának b) alagútgörbék osztályozott vezetőképesség-görbék összetolt, kétdimenziós hisztogramja.

4.5.2. Modell tesztelése magas hőmérsékletű adatokon

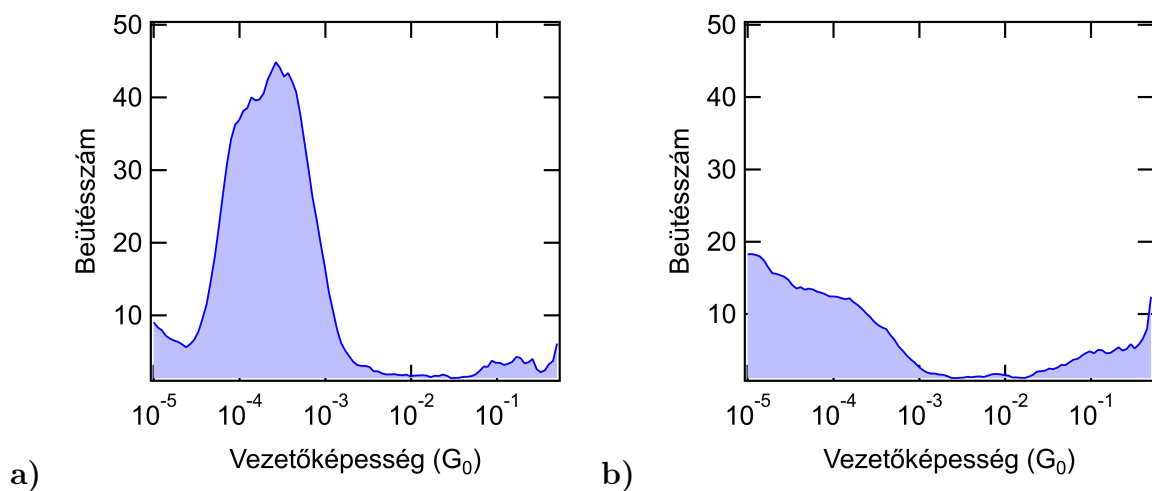
A korábbi, rekurrens neurális háló alapú osztályozás hátránya a csekély általánosító képesség volt [4]. Ebből kifolyólag kíváncsi voltam, hogy az előző fejezetben vizsgálat tárgyát képező modellt mennyire lehetne "univerzálisan" használni. A molekulák beépülését vizsgáló MCBJ-technikán alapuló mérésorozatokból szomahőmérsékleten mért adatok is rendelkezésemre álltak, melyek az alacsony hőmérsékletű adatoknál jóval zajosabbak.

Az alacsony hőmérsékleten mért adatsoron betanított osztályozó hálózatot lefuttattam egy magas hőmérsékletű mérés 1901 darab vezetőképesség-görbéjére, melynek eredményeként a két osztályba sorolt görbék közül a 4.12. és 4.13. ábrákon látható hisztogramokat kaptam. A neurális háló 626 görbét molekulának, 1275 görbét alagútgörbéknek sorolt.

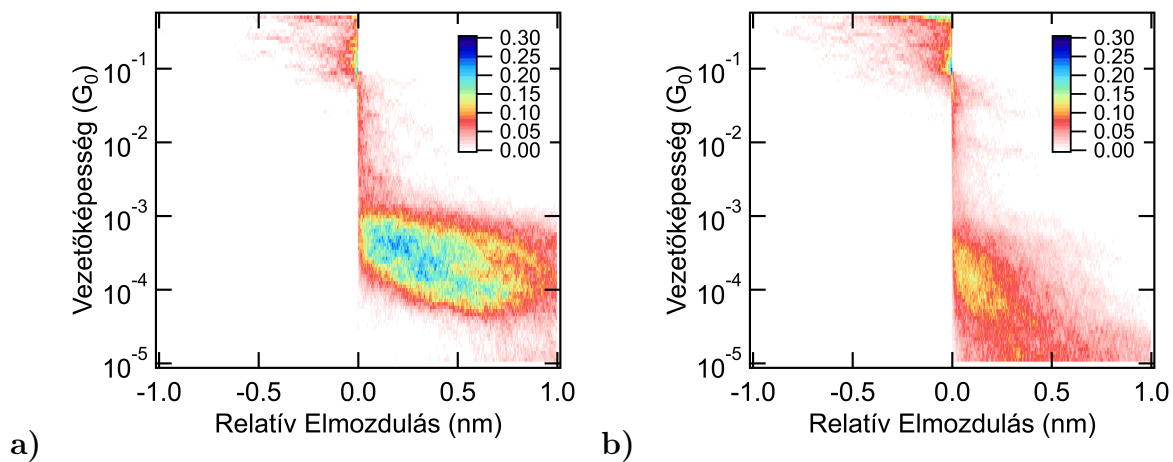
A molekulának osztályozott görbék esetén ismét jól látszik a molekulákra jellemző csúcs az egydimenziós hisztogramon (4.12. a) ábra), azonban az alagút-típusú osztályozott görbék esetén is nagy beütésszám jut a 10^{-3} - 10^{-5} G_0 tartományba. Ennek okai lehetnek a zajosabb görbék, az intervallumban a ténylegesen alagút-típusú görbék alagútáramra jellemző változása miatti nagyobb beütésszám, vagy a hibásan alagútgörbe osztályba sorolt molekulás görbék jelenléte a hisztogramban.

A 4.13. ábra kétdimenziós hisztogramjai is jól mutatják, hogy a molekulának osztályozott görbék valóban tartalmazznak a molekulára jellemző platót, míg az alagútgörbéknek sorolt görbék között kis számban megjelennek olyan görbék is, amelyekről a mért görbékre

rakódó zaj miatt nehéz eldönteni, hogy melyik csoportba sorolhatóak. Összességében elmondható, hogy a neurális hálózat jól alkalmazható azon görbék kiválogatására, amelyek egyértelműen tartalmaznak a molekula beépülésére utaló platót.



4.12. ábra. a) Molekulának, b) alagútgörbének osztályozott vezetőképesség-görbék egydimenziós hisztogramja.



4.13. ábra. a) Molekulának b) alagútgörbének osztályozott vezetőképesség-görbék összetolt, kétdimenziós hisztogramja.

5. Fejezet

Mit tanult meg a hálózat?

Munkám során a 1. fejezetben leírtaknak megfelelően nemcsak az volt a célom, hogy egy jól osztályozó neurális hálózatot hozzak létre molekuláris elektronikai mérések elemzésére. A következőkben az általam vizsgált modellek egyszerűségét kihasználva szeretném demonstrálni, hogy a tanítás során mit tanulhatott meg a hálózat. Az ebből levont következtetésekből pedig egy egyszerű, jól szemléltethető hálózat működését szeretném bemutatni a vezetőképesség-görbékből készített kétdimenziós adatokon.

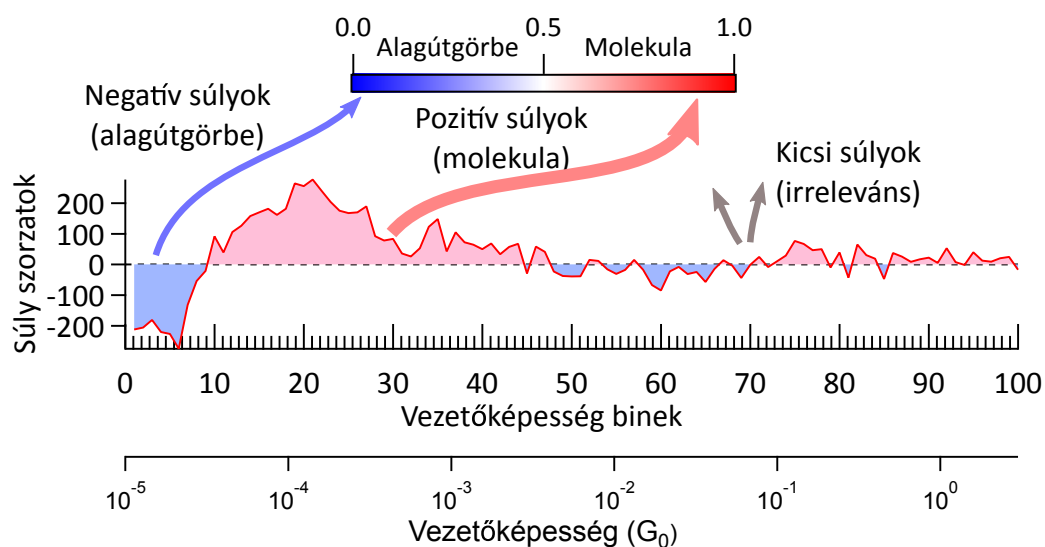
5.1. Hálózat súlyainak értelmezése

A 3.4. fejezetben utaltam rá, hogy a neurális hálózat súlyainak szorzata árulkodó lehet a hálózat döntési mechanizmusáról, illetve arról, hogy mely bemeneti adatok meghatározók.

A 5.1. ábrán ezt egy olyan hálózat súlyainak szorzatával szeretném szemléltetni, melynek bemenetére 100 adatpont kerül, melyet a $10^{-5} - 3 G_0$ vezetőképesség-tartomány 100 binre osztásával kaptam. A vizsgált hálózatban a 100 darab bemeneti neuron 150 darab rejtett rétegbeli neuronhoz kapcsolódik, a rejtett rétegbeli neuronok kimenete pedig súlyozva egyetlen kimeneti rétegbeli neuronra jut. Ha a 3.4. fejezetben leírtaknak megfelelően összeszorozzuk a bemeneti réteg és a rejtett réteg közötti súlyt azon súly értékkel, mely az adott rejtett réteget a kimeneti réteggel köti össze, majd felösszegezzük ezt minden egyes vezetőképesség-binre, akkor a 5.1. ábrán látható eredményt kapjuk. A megfelelő súlyok szorzatának összege egy-egy bemeneti vezetőképesség binre megmutatja, hogy a hálózaton az adott adatpont összességében milyen súlyozással megy végig.

A 5.1. ábrán jól látható, hogy a molekulák beépülésére jellemző tartomány összességében nagy, pozitív súlyokkal járul hozzá a neurális háló kimenetének kialakításához. A pozitív súlyok természetesen a 0 – 1 közötti kimenetet az 1, azaz a molekula osztályra jellemző kimenet felé tolják el. A vártnak megfelelően a neurális hálózat döntése szempontjából nagy szerepe van ennek a tartománynak, ahogy azt a nagy súly-szorzat értékek mutatják.

A vizsgálat eredményeképpen az is megfigyelhető, hogy a $10^{-5} - 3.2 \cdot 10^{-5}$ vezetőképesség-tartománynak megfelelő binek esetén a súly-szorzatok összege negatív. Ez jól magyarázza a vezetőképesség-görbék azon tulajdonságát, hogy a molekuláris kontaktus elszakadásakor a vezetőképesség ugrásszerűen lecsökken, ezáltal minimális beütésszámot eredményezve ezen vezetőképesség-tartományon. Ha azonban alagútgörbe kerül a neurális háló bemenetére, akkor a legtöbb esetben az alagútáram exponenciális távolságfüggése miatt ezen vezetőképesség-tartományban nagyobb beütésszám várható a hisztogram számolásakor. Az ezen vezetőképesség-tartományhoz tartozó binek értékei összességében nagy negatív súlyokkal mennek végig a hálózaton, a kimenetet a 0 érték, azaz alagútgörbe (*tunnel*) osztály felé tolva.



5.1. ábra. A neurális hálózat megfelelő súlyainak szorzata a neurális hálózat bemeneti adatpontjainak függvényében. A bemeneti adatok a vezetőképesség-binek, a hálózat kimenete pedig 0 alagútgörbére, 1 molekulás görbére.

A súlyok szorzatának vizsgálatával a 4.4.3. fejezetben vizsgált $1 G_0$ -os plató hatására is ugyanaz a következtetés vonható le. Az ezen vezetőképesség-érték körüli vezetőképesség-binek kis súlyokkal futnak végig a hálózaton, melyből szintén arra következtethetünk, hogy

a nanovezeték ezen vezetőképesség-tartománybeli elszakadása esetében nem jósolja meg előre a 4,4'-bipyridine molekulák beépülését.

5.2. Két bemenő adaton tanuló hálózat működésének szemléltetése

Az előző, 5.1. fejezetben bemutatott súly szorzatok vizsgálatával lehetőség nyílik arra is, hogy célzottan, a neurális hálózat által nagy súlyokkal ellátott vezetőképesség-tartományokkal tanítsunk be egy nagyon egyszerű, ám nagyon szemléletes neurális hálózatot. A következőkben a vezetőképesség-görbék osztályozásának problémáját kétdimenziós síkon elhelyezkedő pontok osztályozására vezetem vissza, szemléletessé téve a neurális hálózat működését.

A következőkben a neurális hálózataim bemenetén mindössze két adat lesz, melyek hasonlóan foghatók fel, mint a 3.3. ábrákon bemutatott osztályozandó pontok X_1 és X_2 koordinátái. A két bemeneti érték két vezetőképesség tartományba, H és L binbe eső mérési adatpontok skálázott számát jelenti. A következőkben bemutatott két hálózat egyaránt 18 rejtett rétegbeli neuront alkalmaz, 100000 iterációs lépéssel és 0.0005 tanulási rátával tanítottam.

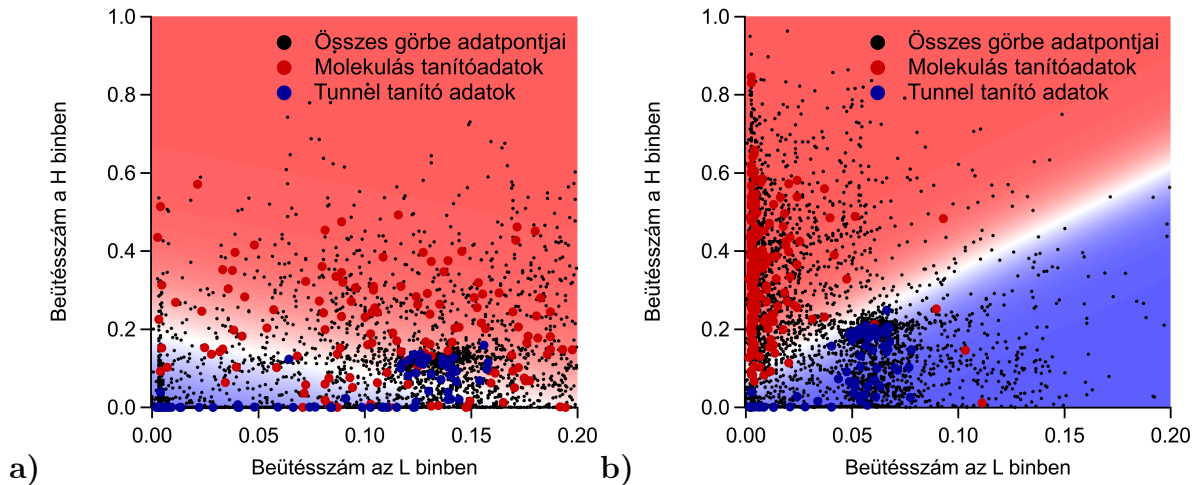
Először az alacsony hőmérsékleten mért, korábbiakban is vizsgált 6000 vezetőképesség-görbéből és a tanító adathalmazból két olyan vezetőképesség-bint vizsgáltam, melyek nagyjából lefedik és elfelezik a molekulákat tartalmazó vezetőképesség-tartományt. Ekkor a $H = 10^{-5} - 2 \cdot 10^{-4} G_0$ és $L = 2 \cdot 10^{-4} - 3.3 \cdot 10^{-3} G_0$, két vezetőképesség-binbe eső mérési adatpontokat határoztam meg és ábrázoltam az 5.2. a) ábrán. Az 5.2. ábrán látható kék és piros pontokkal jelzett tanító adatokkal tanítottam be egy neurális hálót, melynek kimenetét (tunnel (0) -kék és molekula (1) -piros), a kétdimenziós sík minden pontjára kiszámoltam, és az ábra háttérszínéként ábrázoltam. Az adatpontok jól láthatóan keverednek, nem válnak szét két jól meghatározott osztályra, így a betanított hálózat sem tudja megfelelően osztályozni azokat.

Az előző fejezetben vizsgált súly szorzatok alapján szerzett ismeretekből azonban meg tudunk adni olyan vezetőképesség-tartományokat, melyekre az egyes osztályokba tartozó görbék az előzőnél sokkal jobban szétválnak a kétdimenziós síkon (5.2. b ábra). Ezen vezetőképesség-bineket az 5.1. ábra alapján úgy olvastam le, hogy a két bin hatá-

ra azon pont legyen, ahol a súly szorzatok negatívból pozitívba váltanak. A leolvasott vezetőképesség-tartományok: $H = 10^{-5} - 3.2 \cdot 10^{-5} G_0$ és $L = 3.2 \cdot 10^{-5} - 3.3 \cdot 10^{-3} G_0$.

Az 5.2. b) ábra mutatja, hogy a súly szorzatok tanulságai alapján elkészített vezetőképesség binekből a mérési adatpontok és tanító adatpontok jól szétválnak a kétdimenziós síkon. Az ábra háttérszíne ismét megadja a betanított hálózat osztályozási eredményét a sík minden pontjára. Az ábra nemcsak azt szemlélteti jól, hogy a neurális hálózat a jól szétváló adatokat hogyan osztályozza, hanem azt is, hogy korábbi vizsgálatok eredményeképp az osztályozási problémát mennyire jól le lehet egyszerűsíteni.

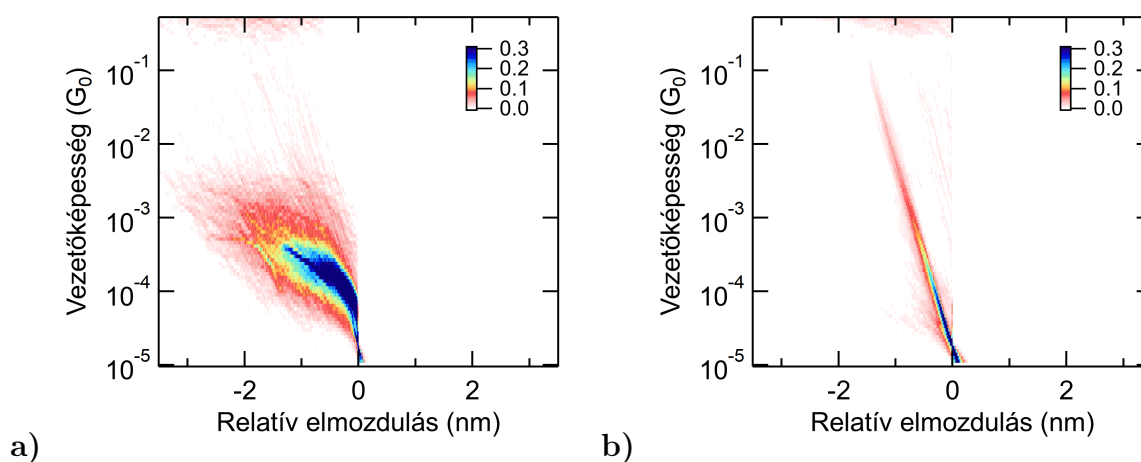
A 5.2. ábra a) és b) grafikonján bemutatott hálózatok osztályozásának pontosságát az 5.1. táblázat tartalmazza. Jól látható, hogy az 5.2. b) ábrán bemutatott két bemenetes neurális hálóval a 12 bemenetes neurális hálóhoz hasonlóan magas pontosságot tudtunk elérni. A tanító és validációs halmazon bemutatott pontosság félrevezető lehet, mert nem tudunk következtetni belőle a modell általánosító képességére. A következőkben ezért a 4.5.1. fejezethez hasonlóan megvizsgálom a 5.2. ábra b) neurális hálójának teljesítőképességét az összes mért görbére.



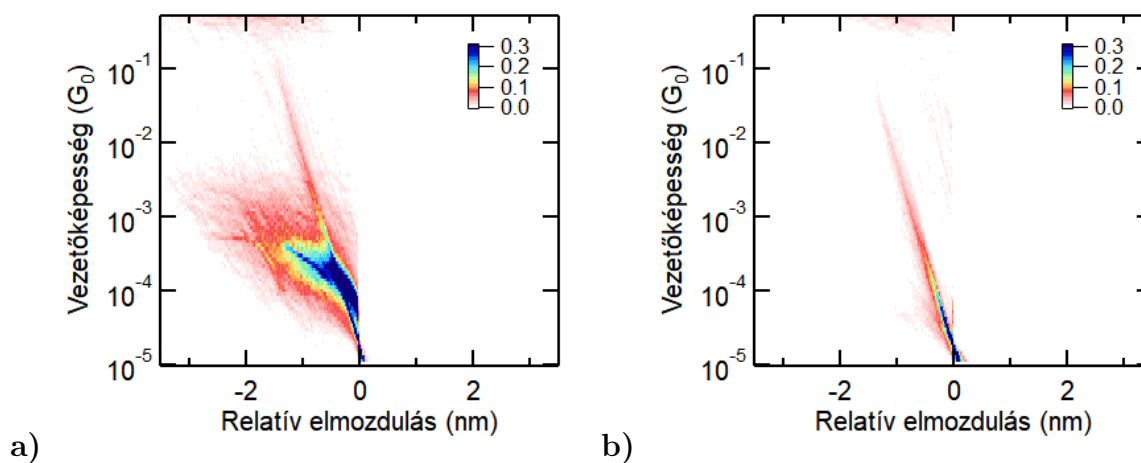
5.2. ábra. Alacsony hőmérsékleten mért MCBJ-mérésekből származó vezetőképesség-görbékből készített, kétdimenziós síkon ábrázolható adatpontok, háttérben az adatpontokra betanított neurális hálózat osztályozásának eredményével. a) $H = 10^{-5} - 2 \cdot 10^{-4} G_0$ és $L = 2 \cdot 10^{-4} - 3.3 \cdot 10^{-3} G_0$ b) $H = 10^{-5} - 3.2 \cdot 10^{-5} G_0$ és $L = 3.2 \cdot 10^{-5} - 3.3 \cdot 10^{-3} G_0$.

Hálózat	Pontosság	
	Tanító halmazon	Validációs halmazon
a)	81.67 %	82.5 %
b)	96.25 %	98.75 %

5.1. táblázat. Az 5.2. ábrán bemutatott neurális hálózatok pontossága a tanító és validációs halmazon.



5.3. ábra. Osztályozás az 5.2. ábra b) neurális hálójával: a) Molekulának b) alagútgörbének osztályozott vezetőképesség-görbék összetolt, kétdimenziós hisztogramja.



5.4. ábra. Osztályozás főkomponens analízissel: a) Molekulának b) alagútgörbének osztályozott vezetőképesség-görbék összetolt, kétdimenziós hisztogramja.

A neurális hálós osztályozás sikerességét szeretném szemléltetni egy másik osztályozási módszerrel történő összehasonlítás során is. Ezen módszer az úgynevezett főkomponens analízis, amit [9]-ban alkalmaztak törőkontaktus mérések osztályozására. Ezen módszer részletes bemutatása meghaladja a dolgozatom kereteit, így most csak röviden ismertetem. A főkomponens analízis során először a mérési görbék hisztogramjai ($N_i(r)$) közötti korrelációs mátrixot kell kiszámolni [10]. A korrelációs mátrix legnagyobb sajátértékéhez tartozó sajátvektor abba az irányba mutat, amerre a legnagyobb variancia figyelhető meg a mérési görbék hisztogramjait tartalmazó adat mátrixban. Az osztályozás úgy történik, hogy a mért görbék hisztogramjait skalárszorozzuk a korrelációs mátrix releváns sajátvektorával, majd a kapott eredmény előjele szerint két csoportba osztályozzuk ezeket a görbéket. Az 5.4. ábrán látható az összes mért görbe osztályozásának eredménye ezen módszerrel. Az 5.4. ábra kétdimenziós hisztogramjain jól látszik, hogy az alagút típusú görbék csoportjába valóban csak olyan görbék kerültek, amelyek valóban alagút-típusúak. Azonban a molekulásnak osztályozott görbék között találhatóak alagút típusú görbék is, tehát alacsony a modell alagútgörbékre vett precizitása.

A neurális hálózatnak a főkomponens analízissel szemben magasabb az alagútgörbékre vett precizitása, azaz a ténylegesen alagút típusú görbék csaknem mindegyikét helyes osztályba sorolja.

A 4.4.4. fejezetben bemutatott eredmények alapján az is érdekes megfigyelés, hogy a molekula-alagútgörbe csoportokba sorolásnál a neurális háló osztályozási pontossága nem függ a rejtett rétegbeli neuronok számától, egyetlen rejtett rétegbeli neuronnal is ugyanolyan jól teljesít a modell. Ez igaz az 5.2. b) ábrán bemutatott két bemenettel rendelkező hálózatra is, egyetlen rejtett rétegbeli neuron alkalmazásával ugyanúgy az 5.3. ábrán bemutatott osztályozást kapjuk. Ez alapján az alagútgörbe-molekula osztályozásnál elegendő egyetlen rejtett rétegbeli neuron alkalmazása is, azonban bonyolultabb osztályozási feladatoknál később szükség lehet több rejtett rétegbeli neuron alkalmazására is.

6. Fejezet

Összefoglalás, kitekintés

A molekuláris elektronikai mérések egyik jelentős területét alkotják az törőkontaktus-méréstechnikával elvégzett mérések, ahol többek között különböző molekulák vezetőképességének vizsgálatára nyílik lehetőség. A mérések eredményeképp kapott nagy adatmennyiségből fontos feladat, hogy a molekulák jelenlétét mutató görbéket kiszűrjük a mért adatokból. Munkám során ezen mérésekből származó vezetőképesség-görbék osztályozását valósítottam meg egy, a területen újszerű módszerrel, neurális hálózatok alkalmazásával. A vezetőképesség-görbékből sikerrel szűrtem egyszerű neurális hálós modellek segítségével a molekulák jelenlétét mutató görbéket alacsony-, illetve magas hőmérsékletű adatokon is.

A 2.5. fejezetben bemutatott rekurrens neurális hálóval történő vezetőképesség-görbe osztályozás egy sok szabad paramétert használó, és bizonyos paraméterek esetén kifejezetten rosszul működő algoritmus. Munkám során kiderítettem, hogy ezen osztályozási problémára a legegyszerűbb neurális háló az egyedi görbékből számolt hisztogram bemenetel kifejezetten jól működik. A rekurrens neurális háló az általam javasolt megoldáshoz képest bonyolultabb, és nagyobb számításigényű algoritmus, továbbá nem ad lehetőséget arra, hogy betekintsünk a döntési mechanizmusába.

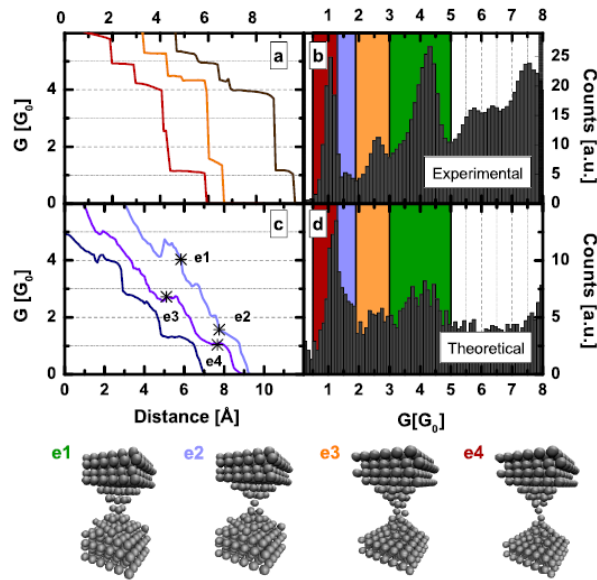
Dolgozatomban csak egy speciális molekula, a 4,4'-bipyridine beépülésére vonatkozó adatokat vizsgáltam. Eredményeim alapján érdemes lehet a neurális hálózatok segítségével megvalósított osztályozást megvizsgálni más molekulák esetén is. Munkám folytatásaként további érdekes kérdéskörként merül fel a különböző molekuláris konfigurációk vizsgálata, vagy akár az is, hogy neurális hálózatok segítségével hogyan lehetne különbséget tenni egymolekulás, vagy két párhuzamos molekulából álló kontaktus között. Reményeim szerint,

a munkám során a vezetőképesség-görbék osztályozására bemutatott gépi tanulási módszer a későbbiekben további struktúrák felismerésére is lehetne használni, ezzel effektív segítséget nyújtva a molekuláris elektronikai mérések értelmezéséhez.

A mérési adatok osztályozásán túl munkám során a neurális hálózat kialakításakor vizsgált paraméterekből és a hálózat súlyainak értelmezéséből fontos következtetéseket tudtam levonni a vizsgált vezetőképesség-görbékkel kapcsolatban. Az egyik ilyen eredmény az, hogy az esetben vizsgált 4,4'-bipyridine molekula beépülése nem függ az egyatomos nanovezeték $1 G_0$ vezetőképesség körüli szakadásának körülményeitől. Vannak azonban például olyan szén-monoxid molekula beépülését vizsgáló eredmények is a szakirodalomban [19], melyekből tudjuk, hogy bizonyos esetekben a molekula beépülése függ a kontaktus "előéletétől". A későbbiekben érdemes lehet neurális hálózatot egy ilyen, úgynevezett prekursor-konfigurációs adatsoron is futtatni. Érdekes lenne megfigyelni, hogy a neurális hálózat felismer-e olyan, a molekula későbbi beépülését előrejelző prekursor-konfigurációkat, melyeket szemre nem vennénk észre az adatsorokon.

Munkám során a neurális hálózatok bemeneti adataként a vezetőképesség-görbékéből készített egydimenziós hisztogram szolgált. Ezen választás kifogásolható, hiszen így elveszítjük a vezetőképesség-görbe időbeliségét illetve az elmozdulás információt. Eredményeim azonban azt mutatják, hogy ennek hiányában is elégséges információ áll rendelkezésre a görbék megfelelő osztályozásához. Szeretném megmutatni, hogy a vezetőképesség-görbék időbeliségének elvesztése nem minden esetben hátrány. Visontai Dávid és társai a Lancasteri Egyetemen például molekuláris dinamikai szimulációkkal készítettek a 6.1. ábrán látható elméleti vezetőképesség-görbéket [20]. Jól látható, hogy a 6.1 ábrán bemutatott kísérleti vezetőképesség-görbétől az elméleti görbék időbeli lefutása jelentősen eltér, azonban az elméleti görbék vezetőképesség-hisztogramja jól követi a kísérleti eredményeket.

A 6.1. ábrán bemutatott elméleti szimulációk alapján kapott vezetőképesség-görbék nagy előnye, hogy a pontos geometriai elrendeződés ismeretében tudjuk vizsgálni a rendszert. Ahogyan a neurális hálózat az alacsony hőmérsékletű adatokon tanítva jól tudja osztályozni a magas hőmérsékletű mérési adatokat, úgy ez esetben is nagy előrelépést jelenthetne egy olyan hálózat, mely az elméleti szimulált görbéken betanulva tudná osztályozni a kísérleti eredményeket. A 6.1. ábrán bemutatottak tükrében az időbeli adatok helyett ezen vizsgálat esetében is érdekesebb lehet a neurális hálózatot a vezetőképesség-hisztogramok alapján betanítani.



6.1. ábra. A felső ábrák kísérleti vezetőképesség-görbéket és az azokból készült hisztogramot, míg az alsó ábrák különböző szimulált vezetőképesség-görbéket és az ezekből készített hisztogramot mutatják [20].

Neurális hálómhoz a tanító adathalmazt kézzel válogattam, amely a későbbi vizsgálatok során, például ha bonyolultabb elrendeződéseket neurális hálók segítségével szeretnék megkülönböztetni, gondot jelenthetne. Kitekintésképpen ezért szeretném megemlíteni, hogy a 2.5. fejezetben bemutatott főkomponens-analízissel a korrelációs mátrix diagonalizálásából kapott fő komponensre való vetítéssel is lehetne a tanításhoz görbéket válogatni. A módszerrel a nagyon egyértelműen alagút és molekuláris görbéket jól szét lehetne választani, amely alapján a neurális hálót betaníthatjuk. A kevésbé egyértelmű görbéket pedig már lehetne a betanított neurális hálózat segítségével osztályozni.

Köszönetnyilvánítás

Szeretném hálásan megköszönni témavezetőimnek, Dr. Halbritter Andrásnak, és Magyar-kuti Andrásnak a munkám során nyújtott támogatást. Köszönöm a sok időt, hasznos tanácsot, konzultációt!

Köszönöm a BME Szilárdtestfizika és Nanokontaktusok Laboratórium összes többi dolgozójának is a segítségét és kedvességét.

Szeretném megköszönni a családom tagjainak a sok-sok biztatást!

Irodalomjegyzék

- [1] Tohru Tsuruoka Kazuya Terabe James K Gimzewski Masakazu Aono Takeo Ohno, Tsuyoshi Hasegawa. Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nature Materials*, 10:591, 08 2011.
- [2] Yuriy V. Pershin Massimiliano Di Ventra. The parallel approach. *Nature Physics*, 9:200, 04 2013.
- [3] Giacomo Indiveri, Bernabé Linares-Barranco, Robert Legenstein, George Deligeorgis, and Themistoklis Prodromakis. Integration of nanoscale memristor synapses in neuromorphic computing architectures. *Nanotechnology*, 24(38):384010, 2013.
- [4] Kasper P. Lauritzen, András Magyarkuti, Zoltán Balogh, András Halbritter, and Gemma C. Solomon. Classification of conductance traces with recurrent neural networks. *The Journal of Chemical Physics*, 148(8):084111, 2018.
- [5] Halbritter András. Elektrontranszport atomi méretskálán. *Akadémiai nagydoktori thesis*, 2014.
- [6] Magyarkuti András. Vezetőképesség és erő egyidejű mérése atomi méretű kontaktusokban. *TDK dolgozat*, 2011. BME.
- [7] G. Rubio, N. Agrait, and S. Vieira. Atomic-sized metallic contacts: Mechanical properties and electronic transport. *Phys. Rev. Lett.*, 76:2302–2305, Mar 1996.
- [8] Nyáry Anna. Egyedi szerves molekulák vezetőképességének vizsgálata. *BSc Szakdolgozat*, 2016. BME.
- [9] J.M. Hamill, X.T. Zhao, G. Mészáros, M.R. Bryce, and M. Arenz. Fast data sorting with modified principal component analysis to distinguish unique single molecular break junction trajectories. *Phys. Rev. Lett.*, 120, 016601, 2018.

- [10] Péter Makk, Damian Tomaszewski, Jan Martinek, Zoltán Balogh, Szabolcs Csonka, Maciej Wawrzyniak, Michael Frei, Latha Venkataraman, and András Halbritter. Correlation analysis of atomic and single-molecule junction conductance. *ACS Nano*, 6(4):3411–3423, 2012.
- [11] Altrichter Márta, Horváth Gábor, and Pataki Béla. *Neurális hálózatok*. Panem Kft., Budapest, 2006.
- [12] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization of the brain. *Psychol. Rev.*, 65:386–408, 1958.
- [13] M. E. Widrow, B. Hoff. Adaptive switching circuits. *IRE WESCON Convention Record*, pages 96–104, 1960.
- [14] J. J. Hopfield. Neurons networks and physical systems with emergent collective computational abilities. *Proc. National Acad. Science*, 79:2554–2558, 1982.
- [15] D. E. Hinton Rumelhart and R. J. G. E. Williams. Learning internal representations by error propagation. *Parallel distributed processing: explorations in the microstructure of cognition, vol. 1, MIT Press*, pages 318–362, 1986.
- [16] Fazekas István. *Neurális hálózatok*. Debreceni Egyetem Informatikai Kar, 2013.
- [17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.
- [18] A. I. Yanson, G. Rubio Bollinger, H. E. van den Brom, N. Agrait, and J. M. van Ruitenbeek. Formation and manipulation of a metallic wire of single gold atoms. *Nature*, 395:783, 1998/10/22/online.
- [19] Zoltán Balogh, David Visontai, Péter Makk, Katalin Gillemot, László Oroszlány, László Pósa, Colin Lambert, and András Halbritter. Precursor configurations and post-rupture evolution of ag-co-ag single-molecule junctions. *Nanoscale*, 6, 2014.
- [20] P. Makk, D. Visontai, L. Oroszlány, D. Zs. Manrique, Sz. Csonka, J. Cserti, C. Lambert, and A. Halbritter. Advanced simulation of conductance histograms validated

through channel-sensitive experiments on indium nanojunctions. *Phys. Rev. Lett.*,
107:276801, Dec 2011.