

---

# Gépi tanulási módszerek a fotometrikus vöröseltolódás-becslésben

---

Fizika BSc szakdolgozat

Horváth Bendegúz

az ELTE TTK Fizika BSc hallgatója

**Témavezető:** Dr. Csabai István egyetemi tanár, Komplex Rendszerek Fizikája Tanszék

Budapest, 2018 május

### *Köszönetnyílvánítás*

Szeretném megköszönni Csabai Istvánnak, hogy elvállalta a témavezetésem, a szerdai megbeszéléseket és hasznos tanácsait, ami segítette munkám és a téma-  
ban való elmélyülést. Szeretném még megköszönni Dobos Lászlónak is a segít-  
séget, tőle is nagyon sok jó tanácsot kaptam.

# Tartalomjegyzék

<b>1. Bevezetés</b>	<b>2</b>
<b>2. Vöröseltolódás-becslés és gépi tanulás</b>	<b>4</b>
2.1. Fotometrikus vöröseltolódás-becslés . . . . .	4
2.2. A gépi tanulási módszerek . . . . .	6
2.2.1. <i>Random Forest</i> . . . . .	7
2.2.2. Mesterséges neurális hálók . . . . .	8
2.3. Eddigi eredmények és módszerek áttekintése . . . . .	12
<b>3. Adatok</b>	<b>15</b>
3.1. Sloan Digital Sky Survey . . . . .	15
3.2. A tanítóhalmaz elkészítése . . . . .	16
3.3. Adatexploráció . . . . .	18
<b>4. Módszerek és eredmények</b>	<b>21</b>
4.1. Neurális hálók és Random Forest kombinálva . . . . .	21
4.2. Konvolúciós háló kiegészítve becsült magnitúdókkal . . . . .	24
4.3. Egy mély konvolúciós háló . . . . .	26
4.4. Teljesen összekötött neurális háló . . . . .	29
4.5. Eredmények összegzése . . . . .	31
<b>5. Összefoglalás</b>	<b>32</b>

---

<b>Függelék</b>	<b>33</b>
A.1. Margaret Gellerék égtérképe . . . . .	33
B.2. Az adattáblához használt lekérdezés . . . . .	34
B.3. A képek megtalálása, kivágása és elmentése . . . . .	35
B.4. A képek ellenőrzése . . . . .	37
<b>Irodalomjegyzék</b>	<b>38</b>

# 1. fejezet

## Bevezetés

Az Univerzum nagy skálás szerkezetének megértéséhez szükséges, hogy térképet tudjunk készíteni a galaxisok elhelyezkedéséről. Két koordinátát, a galaktikusszélességet és galaktikushosszúságot könnyen megkaphatjuk, viszont a távolság meghatározása már nehezebb feladat. A trigonometrikus parallaxis módszer a legjobb technikákkal is csak galaxison belül működik, a jó *seeing* érdekében pedig ūrtávcső kell. A standard gyertya módszerek pontos távolságértéket adnak, viszont a cefeidák, mint viszonyítási pontok csak néhány százmillió fényév távolságon belül alkalmazhatóak, szupernóvákat pedig csak kevés galaxisban lehet látni, így ezt a módszert sem lehet minden alkalmazni, ezért kell egy olyan technika, amivel távolabb is mérhetünk, pontosan, sok galaxisra.

A XX. század elején Edwin Hubble és Vesto Slipher a galaxisok színképének tanulmányozása során észrevették, hogy a színképek eltolódnak a nagyobb hullámhosszak, a vörös színtartomány felé a laboratóriumban mért vonalszerkezethez képest, a galaxis vöröseltolódást szenved. Hubble méréseket készített a galaxisok távolságáról és vöröseltolódásáról, és lineáris összefüggést tapasztalt, amit a később róla elnevezett törvény ír le:

$$v = H \cdot d, \quad (1.1)$$

az arányossági tényező a Hubble-állandó, értéke  $H = 73.45 \pm 1.66 \text{ km/Mpc}$ . Ez az összefüggés lehetőséget ad a pontos távolságmérésre, ha a galaxisok vöröseltolódását meg tudjuk mérni.

Vöröseltolódás mérésére a legfontosabb módszer felvenni az objektum spektroszkópiai képet, és megnézni, hogy a jellegzetes vonalak mennyire csúsztak el. A megfelelő minőségű spektrum felvételéhez akár egy órányi távcsőidő is szükséges lehet, ezért sokáig nem is készültek égbolttérképek. Az 1986-ban Margaret Geller és munkatársai által készítet égtérkép[1] csupán egy vékony szeletet fedett le az égből, de már azon is kirajzolódott (lásd A.1 függelék), hogy a galaxiseloszlás nem egyenletes azon a skálán. Gellerék térképén a legtávolabbi galaxisok körülbelül  $200 \text{ Mpc}$  távolságnyira voltak tőlünk, ezért indokolt volt elkészíteni egy még távolabb látó térképet. A BEKS[2] égfelmérés szűkebb,  $1 \times 1$  négyzetfokos tartományban, ceruzaszerűen<sup>1</sup> nézett  $1000\text{-}1000 \text{ Mpc}$  távolságba mindkét irányba, és ezen a skálán is kirajzolódott struktúra az anyageloszlásban, és a sűrűség ingadozás periodikusnak mutatkozott. Nagy távolságokra csökkent az észlelt galaxisszám, ez a halványabb galaxisok nehezebb észlelhetősége miatt volt. A technikai fejlődésnek köszönhetően elkészülhettek olyan kisérleti berendezések, amelyek lehetővé tették valódi háromdimenzióban a galaxisok pontos helyzetének felmérését. A Sloan Digital Sky Survey egyike ezen eszközöknek, első fázisában 1 millió objektum spektrumát vette fel négy év alatt[3]. Ez a sebeség nem kielégítő, ezért felmerült az igény, hogy a vöröseltolódásokat fotometriai úton mérjék. A fotometriával mért vöröseltolódások kicsit pontatlanabbak, de mérésük gyorsabb mint spektroszkópiával, ezenkívül halványabb objektumokat is lehet vele mérni, magasabban van a magnitúdó korlát. Ezen tulajdonságok vonzóvá teszik a fotometriai a vöröseltolódás-becslő módszereket. A technológiai fejlődés nem csak a csillagászatot érintette, egyre erősebb grafikus proceszorok jelentek meg, amik lehetővé teszik gépi tanulási módszerek szélesebb problémakörre való alkalmazását, mint például a vöröseltolódás-becslést fotometriai adatokból.

Dolgozatom célja az általam készített, gépi tanuláson alapuló fotometrikus vöröseltolódás-becslő módszerek bemutatása, amelyek a galaxisok képeit használjaák fel fotometriaia adatként.

---

<sup>1</sup>pencil-beam survey

## 2. fejezet

# Vöröseltolódás-becslés és gépi tanulás

### 2.1. Fotometrikus vöröseltolódás-becslés

A vöröseltolódás fotometriával történő becslésének kétféle megközelítése van, empirikus és spektrumokon alapuló. Egy spektrumon alapuló módszert először 1962-ban írt le Baum[4], fotoelektronos fotométert használt kilenc sáváteresztő filterrel, amik 3730 Å és 9875 Å közötti hullámhosszú fényt engedtek át. Ezzel a rendszerrel 6 fényes elliptikus galaxis spektrális energia-eloszlását (spectral energy distribution, SED) mérte meg a Virgo halmazból, majd még háromnak egy másik halmazból. A SED-ek átlagát ábrázolta a hullámhossz logaritmusának függvényében, és képes volt észrevenni az eltolódást a két energia sűrűségeleszlás között, így a második klaszternek a vöröseltolódását is megkapta. Mérése pontos volt, de a módszer arra támaszkodott, hogy a spektrumoknak 4000 Å-nél levágása van, melyet a csillag-atmoszférák fémtartalma okoz, ez az elliptikus galaxisoknál jól látható, de például az aktív csillagkeletkezést mutató irreguláris galaxisokban ez a levágás nem figyelhető meg, ezért ez a módszer csak elliptikus galaxisoknál volt használható[5].

David C. Koo egy másik módszert, a szín-szín diagrammok módszerét vezette be 1985-ös cikkében[6]. Négy sáváteresztő szűrőt használt, melyeken mért magnitúdók különbségével létrehozott színtereken ábrázolva az azonos típusú, különböző vöröseltolódású galaxisokat jól definiált görbét kapott. Szín-szín diagrammokat bármilyen kombinációjából lehet készíte-

ni három vagy több színnek, az optimális választás a várt vöröseltolódás-eloszlástól függhet, a gyakorlati választás a rendelkezésünkre álló szűrőktől[6]. Ez a megközelítés a fotometriai vöröseltolódás-meghatározás fontos eleme lett, ugyanis az egyes galaxisok típusának és színszűrőkön mért fényességének ismeretében meghatározható, hogy milyen vöröseltolódást szenved a galaxis.

Egy harmadik, gyakran használt módszer a *template fitting* (sablon illesztés), ez a módszer is a spektrális energiasűrűség-eloszlásra támaszkodik, az ismert vöröseltolódású és SED-ű galaxisok SED-jéből felépül egy könyvtár, és az ismeretlen vöröseltolódású galaxisok SED-jét hozzá lehet párosítani hasonlóság alapján a könyvtárban lévőkhöz. A *template* módserek nagyon hasznosak lehetnek új égboltfelmérésnél, ha nem áll rendelkezésre megfelelő mennyiségű spektroszkópiai adat. A módszer használatával, különösen ha elméleti sablonokat használnak[7], a vöröseltolódáson kívül egyéb fizikai tulajdonságát is ki lehet nyerni a galaxisnak. Megfelelő használatához a sablonkönyvtárnak teljesnek kell lennie, hogy össze lehessen kötni mindegyik mérni kívánt galaxis SED-jét egy sablonnal, különben szisztematikus hiba jelenik meg a mérésben, viszont a túl sok sablon degenerációhoz vezethet. A sablonillesztő és spektrumokon alapuló módszerek egyébb változatait és eredményeik összevetését jól leírják Hildebrandt és társai[8].

Empirikus módszerek alkalmazásához szükség van nagy mennyiségű spektroszkópiával mért vöröseltolódás-adatra és hozzájuk valamelyen, fotometriával mért adatokra. Az egyik legegyszerűbb módszer, hogy a színszűrőkön mért magnitúdóértékekere többváltozós lineáris vagy kvadratikus függvényt illesztenek[9]. Ezeket az adatokat felhasználva a függvényillesztés helyett lehet gépi tanuló eljárásokat is alkalmazni, például *nearest neighbour*[10], *random forest*[11] vagy neurális hálókat[12]. Ezeknek a módszereknek előnye, hogy használatuk viszonylag egyszerű, hatalmas adathalmazokon is működhetnek, illetve nincs szükség a SED-re se, de nagy mennyiségű és jó minőségű tanulóhalmaz kell az előkészületekhez.

## 2.2. A gépi tanulási módszerek

A gépi tanuló módszerek már a XX. század közepén megjelentek, de a számítástechnika és a rendelkezésre álló adatok mennyisége még nem állt olyan szinten, hogy töretlenül fejlődhessen. Az akkori megközelítés szorosan összefüggött a mesterséges intelligencia kutatásával, de az 1990-es években az irány eltolódott a gyakorlati jellegű problémák megoldása felé. Ma már egyre több használható adat és fejlett grafikus processzorok mellett sokféle gépi tanuló algoritmus lett implementálva, így a nehézségek a megfelelő módszer megtalálása és alkalmazása az adatokra, illetve az adatok használható formába hozása. A gépi tanulás célja, hogy a gép *megértsze* az adatok szerkezetét, felismerjen egy szabályt és ez alapján jóslatokat tegyen.

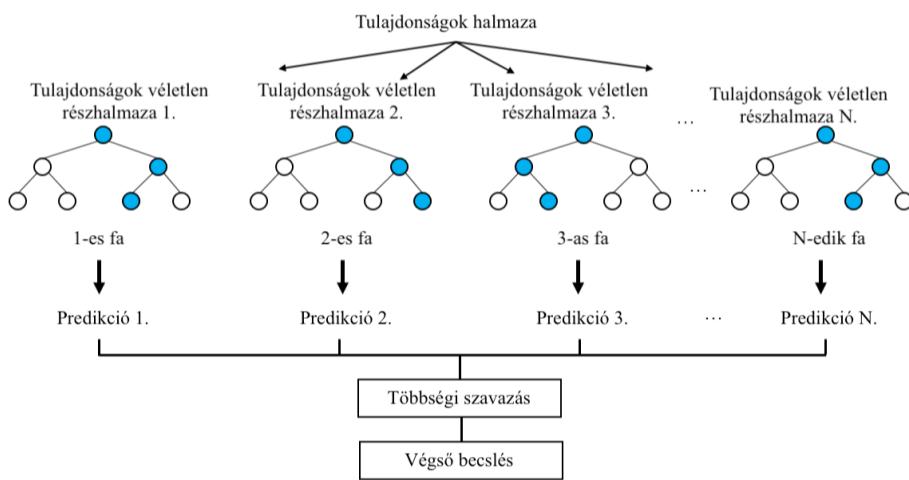
Három nagyobb kategóriába sorolhatjuk a módszereket a rendelkezésre álló adatok és problémák alapján. Az egyik csoport a felügyelt tanulás, ilyenkor rendelkezésünkre álló adatok jelöltek, van egy *ground truth*, amit a modell jóslatainak meg kell közelítenie. A felügyelt tanulási módszereket osztályozás és regressziós problémák megoldásához használják, napjainkban az egyre nagyobb felcímkezett adathalmzoknak köszönhetően egyre több problémára tudják alkalmazni. A felügyelet nélküli tanulási módszerek felcímkezetlen adatokkal dolgoznak, feladatuk, hogy felfedjék a az adtokban rejtt struktúrákat. A *ground truth* hiánya miatt nem lehet jellemezni a tanulás minőségét számértékkal. A fotometrikus vöröseltolódás-becslésben a várt végeredmény jól meghatározott, ezért felügyelt tanulási módszereket alkalmaznak [12], [13]. Ezeknek a módszerek megértéséhez fontosnak tartom bemutatni a munkám során használt eljárások általános működési elvét, és alkalmazhatóságának határait.

Felügyelt tanulásnál rendelkezésünkre áll egy  $(x_1, y_1), (x_2, y_2), \dots (x_N, y_N)$  adathalmaz, ahol  $x_i$  egy mintát jelöl és a tulajdonságok(*features*) számával megegyező dimenziójú vektor,  $y_i$  jelöli az osztálycímét vagy a minta értékét regressziós problámákban, dimenziója az osztályok számával megegyező. A feladat, hogy a gép megtalálja azt a leképzést, a tanítóhalmaz mintái alapján, ami a lehető legkisebb hibával képez  $x_i$ -ből  $y_i$ -be még nem látott minták esetén is. A tanulási hiba mérését az adatokhoz és a feladathoz illő metrikával kell végezni. Fontos, hogy a rendelkezésünkre álló adathalmazunkat ketté kell bontani tanító- és teszt-

halmazra, a tanítóhalmazon belül pedig létrehozhatunk validációs halmazt, amivel a tanulás alatt mérhetjük a modell becslésének pontosságát. A tanító- és tesztszett nem szabad, hogy át fedjen, hogy a modell általánosító képességét tesztelni tudjuk. Túltanulásnak, túlillesztésnek nevezzük, ha a modell becslései túl jól illeszkednek a tanulómintákra, viszont még nem látott minták esetén nem tud általánosítani. Alulillesztésnek nevezzük, ha a becslések nem tudja jól illeszteni a tanulómintákra se. A tanítás ciklus leírásához használt mennyiségek az *epoch*, egy *epoch* alatt a gép az összes elemét látja egyszer a tanítóhalmaznak. A kutatáshoz kétféle módszert használtam, *random forest regressort* és neurális hálókat.

### 2.2.1. Random Forest

A *random forest* algoritmus egyik jó tulajdonsága, hogy alkalmazható klasszifikációs és regressziós problémákra is, alapját a döntési fák sokasága képezi, amiket összefűzve pontosabb becslést tud adni. A döntési fák felépülése a gyökér csomópontnál kezdődik, ami tartalmazza a tulajdonságokat és a célértéket. A csomópontnál meglévő jellemzőket felosztják az így létrejövő utódpontok között, igyekezve a hibát minimalizálni. A folyamatot tovább iterálva létrejön egy rétege a csomópontoknak, amelyek így egy fát alkotnak.



2.1. ábra. A random forest működése. Az ábra forrása:[16]

A fa kialakulását szabályozni lehet paraméterekkel. Ilyen paraméter lehet, hogy hány ré-

tegű, milyen mély legyen a fa, minimum hány elem kerüljön egy levélbe<sup>1</sup>, minimum hány felé legyen osztva a minta egy csomópontnál, vagy milyen módon mérje a szétválasztás minőségét. Véletlenség az erdőbe úgy kerül, ha az egyes fáknál a szétválasztás a csomópontoknál nem a lehető legjobb *split* szerint történik, hanem véletlenszerűen kiválasztott részét kapják meg a tulajdonságoknak. Erre azért van szükség, mert a fontosabb tulajdonságok dominálnának mindegyik fa döntésében, így az egyes fák predikciói korreláltak lennének [14]. Az erdőbe több fát adva az algoritmus nem tanul túl, becslései jobban konvergálnak a kívánt végeredményhez, de határértéke van a hibának [15]. Előnye még, hogy felfedi a prediktálás szempontjából fontos tulajdonságokat, sok attribútummal rendelkező adathalmazoknál ez segíthet az összefüggések megértésében. A fő korlátai, hogy a túl sok fa lassú prediktálást eredményez, illetve regressziós problámáknál a modell nem tud extrapolálni, csak a tanulóhalmaz értékkészletein belüli eredményeket ad. Komplexebb vagy zajosabb adatoknál érdemes máshogyan próbálkozni, ezen esetekben például mesterséges neurális hálókkal jobb eredményt lehet elérni.

### 2.2.2. Mesterséges neurális hálók

A mesterséges neurális hálók megalkotásához a valódi idegrendszer adta a motivációt, de megvalósításához leegyszerűsített, absztrahált neuronokat használtak. Alap építőkövei a neuronok és a súlyok<sup>2</sup>, a neuronok a súlyokon keresztül vannak összekötve egymással és ezek adják meg, hogy az egyik neurontól a másik milyen súllyal kapja meg az értékét. A neuronok rétegekbe vannak rendezve: bemeneti réteg, köztes réteg és kimeneti réteg<sup>3</sup>, a köztes réteg több rétegből is állhat. Az egyazon rétegen lévő neuronok nincsenek összekötve egymással, értéküket az alattuk lévő neuronuktól kapják a súlyokkal számolva. Matematikai formában az értékátadás:

$$z_i = w_{i1}x_1 + w_{i2}x_2 + \dots + w_{in}x_n + b_i, \quad (2.1)$$

---

<sup>1</sup>levél a döntési fának az a része, amiből nem nő ki több ág

<sup>2</sup>biológiai analógiája az axonok

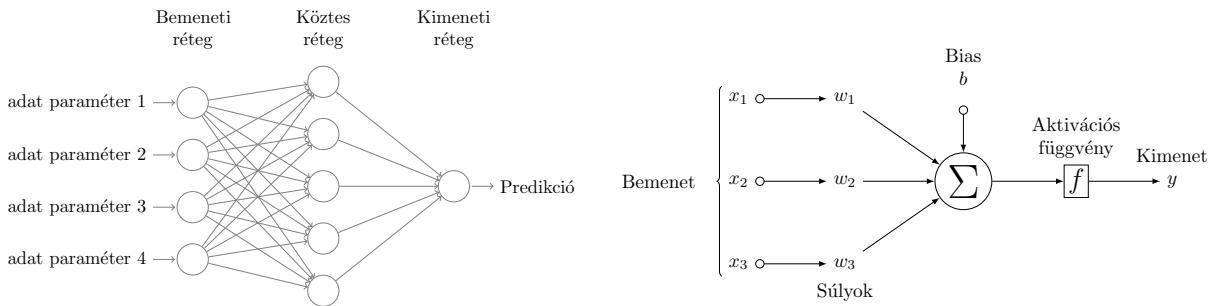
<sup>3</sup>input layer, hidden layer és output layer

az egyenletben  $z_i$  a rétegben az  $i$ -edik neuron,  $w_{ij}$  az  $x_j$  előző rétegbeli neuron és  $z_i$  neuront összekötő súly értéke,  $b_i$  pedig a *bias* vektor  $i$ -edik eleme.

Ez átírható egy egész rétegre:

$$\underline{z} = \mathbf{W}\underline{x} + \underline{b} \quad (2.2)$$

A  $\mathbf{W}$  a  $w_{ij}$  súlyokból képzett súlymátrixot jelöli. Ahhoz, hogy a háló bonyolultabb problémákat is meg tudjon oldani, szükséges nemlineáritást vinni a rendszerbe. Ehhez aktivációs függvényt alkalmazunk, ami eldönti, hogy a neuron aktivizált legyen vagy sem. Gyakran



2.2. ábra. Egy nerális háló vázlatos modellje és egy neuron aktivációjának a folyamata.

alkalmazott aktivációs függvény a *sigmoid*:

$$S(z) = \frac{1}{1 + e^{-z}} \quad (2.3)$$

Illetve a *rectified linear unit(ReLU)*:

$$R(z) = \max(0, z) \quad (2.4)$$

A *ReLU* előnye, hogy a súlyok optimalizálásánál történő deriválásoknál az eltünő gradiens problémája nem áll fenn, így gyorsabb tanulást eredményez és a *sigmoid*hoz képest nem olyan szűk intervallumon ad vissza értékeket. A bemenő adatok ilyen transzformációkon mennek keresztül, míg kimenő adatként össze lehet hasonlítani a várt kimenettel. Az összehasonlítás a hibafüggvénnyel történik, ami kiszámolja a két érték közötti távolságot valamilyen metrikával. Legtöbbször a hibafüggvényt több különböző adat becslése után értékeltetjük ki, a háló *batch*-okban kapja meg az adatot. Regressziós problémákban kedvelt hibafüggvény a jóolt és a

valós értékek közötti átlagos négyzetes eltérés<sup>4</sup> használata:

$$MSE = \frac{1}{n} \sum_{i=0}^n (y_i - p(x_i))^2 \quad (2.5)$$

a képletben a  $p(x_i)$  az  $x_i$  adatokból prediktált érték,  $n$  pedig a *batch* méret. A hiba minimalizálása egy optimalizációs probléma, a  $p$  függvény a súlyuktól függ, amiket úgy kell megváltoztatni, hogy a hiba minimális legyen. A súlyok optimalizálása *backpropagation*-nel történik, a hibafüggvényt deriválva a súlyok szerint a láncszabállyal, az utolsó rétegtől kezdve az elejéig, megkapjuk a gradiensmátrixokat rétegenként, amit szorozva egy kis számmal(tanulási ráta<sup>5</sup>), levonva a súlymátrixkból megkapjuk az új súlyokat. A súlyok frissítése *batch*-onként történik, így a *batch*mérettel és a *learning rate*-tel is lehet a tanulást gyorsítani, majd pontosabbá tenni [18].

A hálók architektúrájának kialakítására nincs általános szabály, intuícióinkra és hasonló problémákon jól szereplő hálók mintájára kell hagyatkozni, tanulásából következtetéseket levonni és alakítgatni. Általanosságban a komplexebb problémákhoz *mélyebb*, több neuronból álló hálók kellenek, de ekkor a beállítandó paraméterek száma is nagyobb lesz, a tanítás lassabb lesz a megnövekedett számítási igény miatt. Képi adatoknál a paraméterek száma nagyon magas lenne, valamint a képnek nem az egész része érdekes, csupán részletek, és ezek a részletek bárhol lehetnek a képen, ezért nem célszerű teljesen összekötött neurális hálókat használni. Ezeknek a problémáknak a megoldására találták ki a konvoluciós neurális hálókat.

A konvoluciós hálókban a rétegek között nincs teljes összeköttetés, a neuron csak az alatta lévő neuronnal, és annak szomszédaihoz kötődik. Az objektumfelismerő-osztályozó problémáknál az objektum bárhol előfordulhat a képen, ezért a neuronokat összekötő súlyoknak is eltolás invariánsank kell lennie, ezért egy rétegen minden neuron ugyanazokkal súlyokkal összegzi az allata lévő neuron szomszédainak kimenetét. A figyelembe vett szomszédok száma meghatároz egy *filter* méretet, a *filter* technikailag egy tensor, aminek elemei a súlyok. A bemeneti képet végig páztázza a *filter*, és végrehajta az összegzéseket, az egyes kimene-

---

<sup>4</sup> mean squared error

<sup>5</sup> learning rate

tek egy *aktivációs térképet alkotnak*<sup>6</sup>, amire alkalmazhatjuk a következő réteg konvolúciós réteget. Ez a műveleletet kétdimenziós diszkrét konvolúció a matematikában. A kimeneti *aktivációs térkép* formája egy  $n$  széles és hosszú képen alkalmazott  $f \times f$  méretű,  $s$  ugrással mintavételező *filterrel*, a képre  $p$  *padding*<sup>7</sup>-et alkalmazva a konvolúció során a következő képpen alakulnak:

$$a = \frac{n + 2p - f}{s} + 1, \quad (2.6)$$

a képleteben  $a$  létrejövő *aktivációs térkép* szélessége,  $k$  *filtert* alkalmazva  $a \times a \times k$  alakú *aktivációs térképet* kapunk. Az *aktivációs térképek* tartalmazzák az információt, de sok közte az elhanyagolható, így a térképek méretét csökkenteni kell, hogy ne legyen túl sok felesleges érték, ami megnöveli a számítási időt. Erre egy módszer az alulmintavételező rétegek<sup>8</sup> használata, ami leszűkíti az *aktivációs térképek* méretét állítandó paraméterek használata nélkül, és újabb eltolási invarianciát is visz a rendszerbe. A *poolingoknak* sok fajtája van, legnépszerűbb a *Maxpooling* és az *AveragePooling*, alapja, hogy a *filterekhez* hasonlóan végig pásztázza a képet egy *kernel*, *Maxpooling* esetében a kernel méreten belüli elemek közül a maximális nagyságút adja át az *aktivációs térképnak*, *Averagepoolingnál* pedig a kernelen belüli elemek átlagát adja át.

Ezekkel az építőkövekkel sokrétegű hálózatok hozhatóak létre, melyeknek nem szükséges az adatokat, képeket értelmes változóvá transzformálni, a hálózat meg tudja találni a prediktáláshoz szükséges információkat, az ilyen hálózatokat a mélységük miatt szokás *deep learning* hálózatoknak nevezni. Ezekben a nagy hálózatokban a túlillesztés elkerülése és a tanulás gyorsítása érdekében szükséges regularizációkat is alkalmazni. Az egyik általam is használt regularizáció a *Dropout*, melyet berakva teljesen összekötött retegek közé, az ki-kapcsolja az általunk megadott százaléknyi neuront az előző rétegből, így a következő réteg nem kapja meg az összes előző információt, nem engedi túlilleszteni a hálótt. Működési elvéhez hasonló a füvvényillesztésnél a túl magas fokszámú polinomok mellőzése, ha túl magas rendű polinomot illesztünk az adatponokra, a hibánk minimális lehet, de nem biztos, hogy

<sup>6</sup>szokásos feature map-nek is nevezni.

<sup>7</sup>kép szélén nullákkal létrehozott keret

<sup>8</sup>*pooling* réteg

jól írja le az illesztett függvény a relációt, a *Dropout* ezt a túl illesztést akadályozza meg a neurális hálóknál, azáltal, hogy egyszerűbb módon, kevesebb neuronnal próbál illeszteni. A *Dropoutok* általában csak a tanulás során vannak bekapcsolva, prediktálásban az összes neuron részt vesz. Ezenkívül egy hasznos normalizáció a *Batchnormalization*, ami növeli a háló stabilitását azáltal, hogy az előző réteg aktivációjából kijővő értékekből levonja a *batch* átlagát, majd leosztja a szórásával. A *Batchnormalizationnel* nagyobb *learning ratet* lehet alkalmazni, mert biztosítja, hogy egyik aktiváció se lesz túl magas, ezenkívül az eltérő tanító és teszthalmaz eloszlás problémán is javít. Ez egy viszonylag új találmány, így az ismertebb, jól működő hálók architektúrái még nem tartalmazzák.

### 2.3. Eddigi eredmények és módszerek áttekintése

Ahhoz, hogy munkámat és eredményeit kontextusba tudjam helyezni, szükséges az eddigi eredményekről és módszerekről áttekintést nyújtanom. Először az eredmények irodalomban használt kiértékelés módjait mutatom be.

Vöröseltolódás-becslésnek a jóságát leggyakrabban a reziduális négyzetösszegek átlagának a négyzetgyöke<sup>9</sup> szokták mérni:

$$rmse = \sqrt{\frac{1}{N} \sum_{i=1}^N (p(x_i) - y_i)^2}, \quad (2.7)$$

a képletben  $p(x_i)$  jelöli az  $x_i$  mintából prediktált értéket,  $y_i$  az  $i$ -edik célváltozó,  $N$  pedig a prediktált értékek száma. Az így kapott szám jól jellemzi a prediktálást, de a kiugró értékek<sup>10</sup> okozhatnak torzítást. Az *outlierek* arányának mérésére van a következő mód:

$$\frac{|p(x_i) - y_i|}{1 + y_i} > a, \quad (2.8)$$

ahol  $a$  legtöbbsször 0.15. Az egyenlőtlenségen  $a$ -nál nagyobb értékeket adó  $p(x_i)$ -ket *outliereknek* nyilvánítjuk, így lehet számolni, hogy a becslés milyen arányban tartalmaz a jótól

<sup>9</sup> továbbiakban az angol megfelelőjét, a *root mean squared error* (*rmse*) használom

<sup>10</sup> továbbiakban az angol megfelelőjét, az *outlier* használom

jelentősen eltérő értékeket. Ezeken jellmezőkön kívül az irodalomban a prediktálást abrázolni is szokták, mert ez lehetőséget ad a modell becslésének hibáinak felfedésére. Ábrázolni a fotometriával becsült vöröseltolódás-értékeket ( $z_{phot}$ ) lehet a spektroszkópiai vöröseltolódás-értékek függvényében ( $z_{spec}$ ), tökéletes becslés esetén ez egy  $45^\circ$ -os egyenest adna.

A vöröseltolódás-becslést gépi tanulási módszerekkel való megvalósítása már régóta működő módszer, de ezek kinyert fotometriai adatokból tanultak és prediktáltak, nem pedig képekből. Az egyik első ilyen munka neurális hálókat használt [19], a tesztszettükben a vöröseltolódás-értékek 0 és 3.3 között mozogtak, és 20 000 galaxisból állt, a 0.3 feletti vöröseltolódás-értékek szimulációkból származtak<sup>11</sup>. A háló magnitúdóadatokat használt bemeneként, és jóval kevesebb neuronból állt, mint a ma használlatos modellek. A szerzők próbálkoztak több háló becslésének összetársításával, így  $rmse = 0.113$  hibát értek el.

Szintén a magnitúdóadatok felhasználával dolgozva, de más módszereket alkalmaztak a [10] szerzői, másodrendű polinom illesztést, illetve *nearest neighbour* módszert használtak az SDSS *Early Data Release*ből származó adatokra. A polinom illesztéssel  $rmse = 0.0318$ , *nearest neighbourral*  $rmse = 0.0365$ -es eredményt értek el. Megjegyezendő, hogy a galaxisok vöröseltolódása 0 és 0.6 közöttiek voltak, ezért is lehetett jóval kisebb  $rmse$ , mint a neurális hálóval.

Egy fejlettebb neurális hálókat használó módszer [12]  $rmse = 0.0230$  hibával tudott becsülni legjobb esetben, a bemeneti adatok nem csak magnitúdóadatok voltak, hanem az *angular radiival*<sup>12</sup> és *concentraiton indexsel*<sup>13</sup> ki voltak egészítve. A szerzők kiemelik, hogy a ezzel a kiegészítéssel, jobb eredményt lehet elérni, viszont okosan kell megválasztani a tulajdonságokat. A tulajdonságok effektív kiválasztásával foglalkozik egy viszonylag új cikk [20], de a számunkra érdekes és mostanában használt módszerek a becsléshez szükséges tulajdonságok kinyerését a neurális hálóra bízzák, és csak képeket használnak ehhez.

Az egyik első képi adatokból becslő módszer [21] az  $g, r, i$  és  $z$  szűrőkön keresztül készí-

<sup>11</sup> 1.5 feletti vöröseltolódású galaxisokat jelenleg nem lehet jól észlelni, ezért megbízható vöröseltolódásbecslő módszereket sem lehet tesztelni rajtuk.

<sup>12</sup> közvetetten információt tartalmaz a galaxis szögméretéről

<sup>13</sup> fényeség profiljának meredeksége

tett képeket használta, a csatornákat  $g-r$ ,  $r-i$ ,  $i-z$  és  $r$  módon alakította ki. A kivágott képek  $72 \times 72 \times 4$ -szeresek voltak, melyekből véletlenszerűen kivágott egy  $60 \times 60 \times 4$  méretű, és ezt kapta meg a mély konvolúciós háló. A szerző kitér arra is, hogy  $32 \times 32 \times 4$  méretű képek használatával a becslés teljesítménye 30%-ot romlik. Az adathalmaz vöröseltolódás-eloszlása viszonylag egyenletes volt, 0 és 1 közötti értékekkel, 64 647 objektumból állt és az SDSS *DR10*-ből származtak, ezt osztotta fel tanító, teszt és validációs részekre. A modell osztályozó volt, vékony vöröseltolódás-intervallumok alkották az osztályokat. Ezzel a módszerrel  $rmse = 0.030$  pontosságot ért el, ami jó eredménynek számít figyelembe véve az adathalmaz vöröseltolódás-értékeit.

Egy másik megközelítés [22] a galaxisokhoz  $28 \times 28$ -as képeket használt minden az öt színszűrővel, képezte még az egyes színszűrőkkel készített képek egymással vett különségét, és ezeket is felhasználva egy 15 csatornás képet kapott. A háló nem osztályozó vagy regressziós kimenetű volt, hanem egy eloszlás paramétereit adta vissza, amiből vissza lehet állítani a vöröseltolódásértéket. Az adathalmazának vöröseltolódás-eloszlása egyenletes volt, a becslése  $rmse = 0.0365$  pontosságú volt.

A kutatásomban az SDSS *DR7*-es adatait fogom használni, így érdemes megnézni, hogy azon az adathalmazon milyen eredményeket értek el. Az SDSS *DR7*-en a [13] munkán alapuló, *nearest neighbour* technikát alkalmazva csináltak vöröseltolódásbecslést [23]. A bemeneti adatok a színindexek voltak, és 100 szomszédot vett figyelembe az algoritmus, az *outlier*-ket is jelölte. A szerzők megjegyzik, hogy az adathalmazban a vörös galaxisok dominálnak, amelyeknek a vöröseltolódását könyebb mérni, és a vöröseltolódásuk 0.2-nél kisebb. Az elért  $rmse = 0.025$ , munkám során ezt az értéket kell figyelembe vennem az összehasonlíthatóság érdekében.

## 3. fejezet

# Adatok

Az eredményes gépi tanuláshoz a nagy mennyiségű és jó minőségű adat majdnem annyira fontos, mint a jó modell megválasztása. Általában nem áll rendelkezésünkre egyből az algoritmusnak adható adat, a nyers adatokat előbb preprocesszállni kell, ki kell nyerni a fontos tulajdonságokat, melyek előzetes ismereteink alapján fontosak lehetnek, és össze kell állítani az adathalmazt, amelyet majd szétválasztunk tanuló- és teszthalmazra.

A kutatómunkámhoz sok, jó minőségű galaxis képére volt szükségem, és mindegyik galaxisnak a spektroszkópiával mért vöröseltolódására is, ezekhez az adatokhoz a Sloan Digital Sky Survey adatbázisában fértem hozzá.

### 3.1. Sloan Digital Sky Survey

A Sloan Digital Sky Survey az eddigi legrészletesebb égboltfelmérés, mély, több színávos képekkel az ég egyharmadáról, 500 millió asztronómiai objektumról, 3 millió felvett spektrumadattal. A felmérést 2000-ben kezdte, több ciklusban, 2014-ben kezdődött a negyedik fázis (SDSS-IV), és már elkezdődtek a megbeszélések az SDSS-V elindításáról[24]. A megfigyelési adatokat egy külön erre a célra megépített 2.5 m széles optikai teleszkóp szolgáltatja az Apache Point Obszervatóriumból, Új Mexikóban. Öt színsávban mér, a látható fény és az infravörös tartománya között,  $u$ ,  $g$ ,  $r$ ,  $i$  és  $z$  színszűrőkkel (ultraviolet, green, red,

near infrared és infrared), amik  $3000 \text{ \AA}$  és  $10000 \text{ \AA}$  közötti tartományt fedik le. A különböző szűrőkön keresztül a CCD chipek egymás után rögzítenek, 71.2 másodperc késéssel,  $r$ ,  $i$ ,  $u$ ,  $z$  és  $g$  sorrendben, így a különböző színű képeken előfordulhat, hogy egy objektum kicsivel arrébb szerepel a pixelkoordinátarendszerben. Ennek volt előnye is, mozgó objektumokat könnyebben lehetett azonosítani, például létre tudtak hozni aszteroida katalógusokat. Az SDSS képek alapvető egysége a *field*, ami 10-szer 13 szögperces szeletet tartalmaz az égből, és 1489-szer 2048 pixelt tartalmaz és az egyes *fieldeket* három jelzőszám teszi egyedivé, a *field number*, a *run*, ami a szkennelés száma, a *camera column*, ami a megmutatja melyik oszlop CCD kamera készítette a képet, ez egy 1 és 6 közötti szám, minden egyik oszlop egy *fieldet* készít, ami 128 pixel szélességben fed át a szomszédossal. Az elkészült felvételek az SDSS képfeldolgozó pipelinejába kerülnek, ahol kalibrált FITS formátumú képet csinálnak, és a katalógushoz hozzá adják a képi paramétereiket.

A megfigyeléseket folyamatosan végzik, az adatbázisokat viszont csak évente frissítik, ezeket a *data release*-nek (DR) hívják és tartalmazzák az előző megfigyelésekkel származó adatokat is. Az SDSS virtuális oszervatórium keretrendszerben működik, adatai publikusak. Az objekumok és paramétereik relációs adatbázisba vannak rendezve, SQL nyelvet használva lehet lekérdezéseket indítani.

## 3.2. A tanítóhalmaz elkészítése

A tanítóhalmaz elkészítését a *SciServeren*[25] végeztem, amely *Jupyter Notebook* keretrendszerben nyújtott adatfeldolgozó és számoló felületet, valamint közvetlen elérést az SDSS *DR7*-es *fieldekhez*. A szerveren egy *API* segítségével közvetlenül lehetett *python* környezetben SQL lekérdezéseket végezni az SDSS adatbázisából, az így kapott táblákat elmenteni. Két tanítóhalmazt állítottam össze, az 1-es adathalmaz 150 000 képből áll, a galaxisok minden egyik színszűrőn mért magnitúdója  $15^m$ -nál halványabb és  $22^m$ -nál fényesebb. A legtöbbször ezt az adathalmazt használtam tanításra és tesztelésre, viszont a vöröseltolódás-eloszlásban lévő csúcs miatt létrehoztam egy 2-es adathalmazt is, melyben a vöröseltolódás-eloszlást próbáltam kicsit egyenletesebbé tenni, így nem csináltam felső határt a magnitúdónak, de a

vöröseltolódásokat határok közé szorítottam és több lekérdezés eredményéből állt össze az adattábla.

Feltételként szabtam meg, hogy a galaxisok vöröseltolódás 0.05-nél nagyobb legyen, hogy a nagy fényességük miatt a képeken szaturációt okozó csillagok ne keveredjenek bele, valamint 1-nél kisebb legyen a vöröseltolódás, hogy az SDSS adatbázisában tévesen galaxisnak osztályozott kvazárok se kerüljenek bele. A képméret kiválasztásánál figyelembe kellett venni, hogy a túl nagy kép túl sok haszontalan információt is tartalmazhat, valamint a közeli szomszédos objektumok is félrevezethetik a neurális hálót, de a túlságosan kicsi képméret miatt lehet lemarad fontos információ. Ezért vizualizálva a galaxisokat különböző képmérettel, az optimális választásnak az  $50 \times 50$  pixeles méret tűnt.

A *SciServer*-en az egyes *field*ek olyan könyvtárstruktúrába vannak rendezve, hogy a *run*, *rerun*, *camcol*, *fieldID* és színszűrő alapján kereshetőek. Ezért az SQL lekérdezésnél a magasság, vöröseltolódás és objektum azonosító addatai mellett lekérdeztem *field* könyvtárban megtalálásához szükséges adatokat, valamint az egyes színszűrőkkel készített képen hol van a galaxis közepe<sup>1</sup>. A galaxis közepének lehelyezkedésére az volt a kikötés, hogy ne a kép szélénél 25 pixeles szomszédságában helyezkedjen el, így a kivágásnál nem fog gondot okozni a hiányzó információ. Az 1-es adathalmazhoz tartozó táblához a lekérdezést mellékeltem a B.2 függelékben. Az így kapott táblákkal előtudtam hívni a FITS formátumú *field*eket, melyekből ki tudtam vágni a galaxis  $50 \times 50$  méretű képét, levontam a *Softbias-t*<sup>2</sup>, és az eredeti könyvtárstruktúrához hasonló módon elmentettem a képet, egy *python dictionary*be pedig az elérési útvonalat és az objektum azonosítóját, így azonosító alapján előhívható volt a kivágott kép. Az eljáráshoz tartozó kódot mellékelem a B.3 függelékben. Ezt mindegyik szűrővel készült képre külön megcsináltam, majd az összes képet egy nagy tömbbe tettem a könnyebb mozgatás és elérés érdekében. A tömb dimenziója objektumok száma  $\times 12\,500$  lett, a különböző színű, azonos objektumról készült képeket egymás után tettem, így az adatokat tartalmazó

<sup>1</sup> sor és oszlop pixelben, a különböző színszűrőkön készített képeken máshol volt az objektum közepe a SDSS képalkotása miatt.

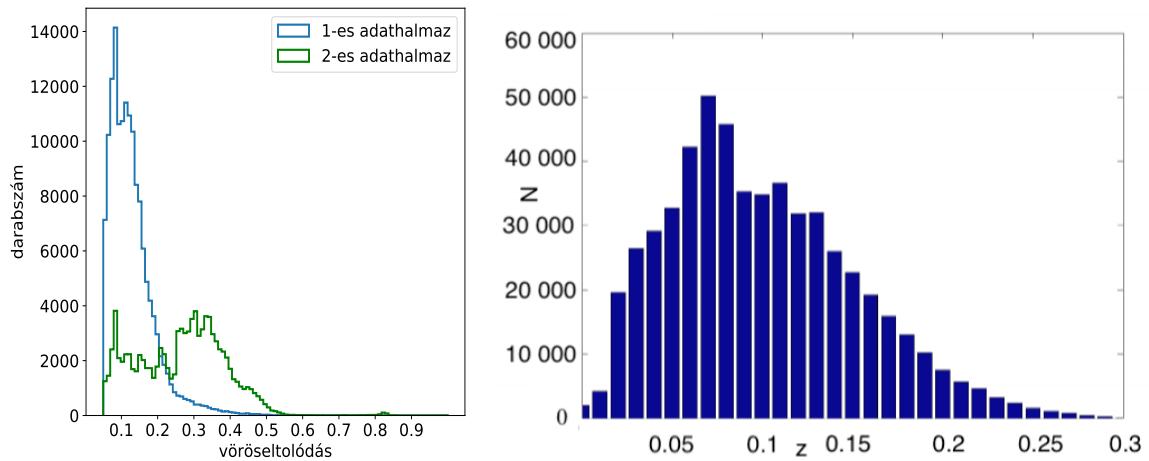
<sup>2</sup> a *field*ekhez hozzáadtak egy 1000-es nagyságrendű számot a kép készülése után, hogy ne legyen negatív pixelérték sehol.

tábla sorának sorszámával lehetett megtalálni a képeket. Ez a tárolási mód okozhatna olyan hibát, hogy elcsúsznak, és nem a megfelelő indexnál lesznek a képek, de ezt véletlen mintavételezéssel megjelenítve a képet és az azonosítót, össze tudtam hasonlítani az SDSS képeivel (lásd B.4 függelék). A képi adattömb az 1-es adathalmazhoz 14 GB méretű lett, aminek a betöltéséhez nagy memóriájú gép kellett, a betöltött tömb formája már könnyen alakítható volt a feladat igényeihez.

### 3.3. Adatekploráció

Mielőtt elkezdenénk a tanítást, érdemes megvizsgálni a tanulóhalmazunk statisztikai jellemzőit, megnézni mennyire jól reprezentálja a valóságot, valamint feltárni az összefüggéseket. Ehhez az SDSS *DR7*-es adatbázisának vöröseltolódás-eloszlásával hasonlítottam össze a tanulóhalmaz eloszlását, alkalmaztam egy *random forest regressor* az *ugriz* magnitúdóértékekre, és készítettem egy *korrelációs térképet* a magnitúdókkal és vöröseltolódásokkal.

Az SDSS *DR7*-ben a galaxisok vöröseltolódásának mediánja  $z \sim 0.07$  [26], a teszthalazomé  $z \sim 0.11$ , ami a 0.05-ös vöröseltolódásbeli vágásomnak tudható be. Összehasonlítva



3.1. ábra. Az általam előállított tanulóhalmaz és az SDSS *DR7* vöröseltolódás-eloszlása. A jobboldali kép forrása: [26]



3.2. ábra. A korrelációs mátrix, a Random forest tanulásában a különböző tulajdonságok relatív fontossága és a magnitúdókból becslése. A szaggatott vonal a rmse nagyságát ábrázolja.

a 3.1. ábrán a baloldalon az 1-es adathalmaz hisztogramját a jobboldalival, a két eloszlás jó hasonlóságot mutat, a vágásban van különbség, a legtöbb galaxisnak mind a kettő esetben 0.3-nál kisebb a vöröseltolódása. .

A korrelációs mátrix elemeiből az látszik, hogy az egymás melletti színek magnitúdója erősen korrelál a szomszédossal. A magnitúdok és a vöröseltolódás nem korrelál ilyen szinten, legmagasabb korrelációs együtthatója a  $g$  magnitúdoknak van a vöröseltolódással. Ezek után alaklmaztam az adatok közül az első 100 000-re a *Random forest*-et, majd prediktáltattam az utolsó 50 000 mintán. A véletlen erdő  $rmse = 0.029$  jósággal prediktált, a becsült vöröseltolódás-értékeket a 3.2 ábra bal szélén ábrázoltam a spektroszkópiai vöröseltolódások függvényében. Az ábrán látható, hogy a pontok nagyrésze ráfekszik az ideálist jelző vonalra, viszont ahol a legsűrűbb a pontok elhelyezkedése, szisztematikus hibája van a rendszernek, felül becsül. A legfontosabb paraméternek a becslés során a  $g$  magnitúdóérték bizonyult, ennek relatív fontossága 0.59 volt, ami jóval magasabb, mint a második legfontosabb  $u$  értéknek, ami 0.24. A tulajdonságok relatív fontosságának sorrendje összhangban van az egyes fényeségek vöröseltolódással vett korrelációjával.

A magnitúdó adatok elemzése azért indokolt, mert a bemeneti adataink az öt csatornás képeket lesznek, és a magnitúdók az egyes csatornákhoz köthető származtatott adatok. A legtöbb

gépi tanuláson alapuló vöröseltolódás-becslő módszer ezeket az adatokat használja, ezért ha a képekből ki tudjuk nyerni gépi tanulással megfelelő pontossággal a fényességértékeket, alkalmazhatunk már bevált módszert rá.

## 4. fejezet

# Módszerek és eredmények

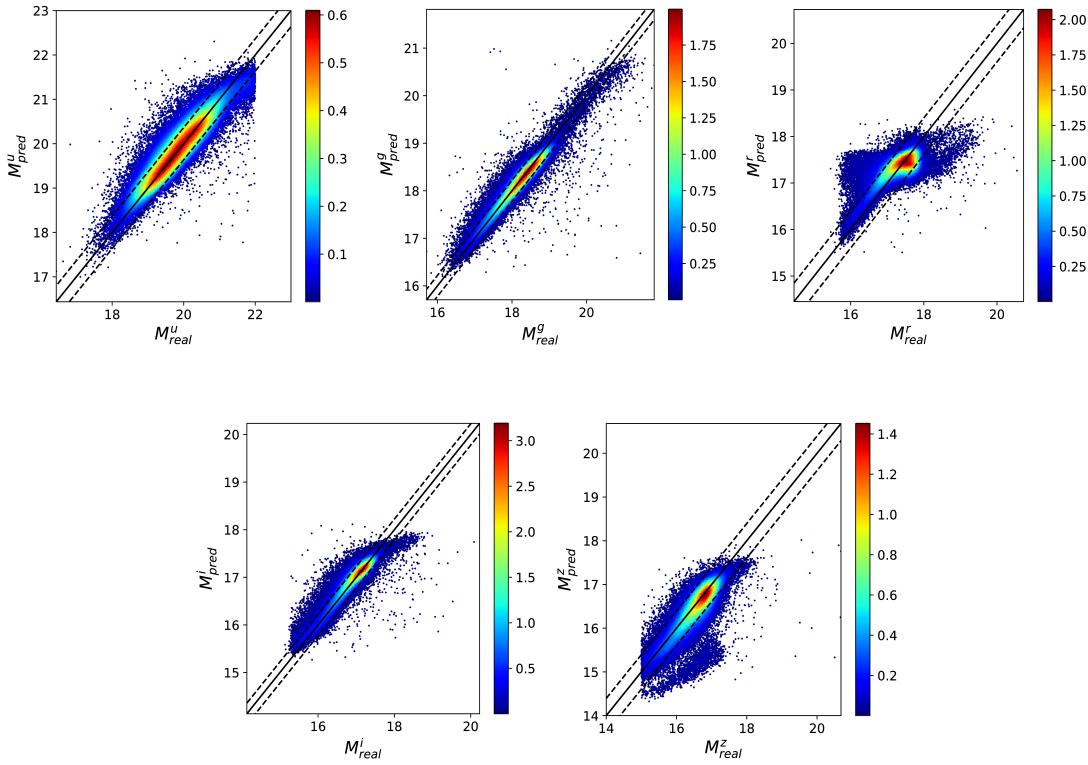
A kutatáshoz az egyes gépi tanulási módszereket *python* nyelven a *keras* (neurális hálók) és *sci-kit learn (randomforest)* könyvtárakkal valósítottam meg. A tanulás gyorsítása érdekében a tanítást a *Google Cloud* felhő alapú számítási szolgáltatást nyújtó platformon végeztem NVIDIA Tesla K80-as GPU-val.

### 4.1. Neurális hálók és Random Forest kombinálva

A *random forest* algoritmus jól teljesít a magnitúdókból prediktálás során, így célszerű megvizsgálni, hogy mennyire működik jól, ha a magnitúdóértékek is becslésből származnak. A képekből való fényességbecsléshez mind az öt színszűrőhöz elkészítettem egy-egy konvoluciós neurális hálót, ezeket a hálókat az első 50 000 képen tanítottam be. A következő 50 000 képpel prediktáltam magnitúdóértékeket, ezeket a becsült magnitúdókat használtam, a *random forest regressor* tanításához. A harmadik 50 000 képnek is megbecsültem a magnitúdóit, és ezekből predikált a véletlen erdő vöröseltolódás-értékeket, így nem volt átfedés a tanuló- és teszthalmazok között. A neurális hálók tanításánál 80 *epoch*-ot használtam mindegyik hálóra, de nem egyben ment végig a tanulás, a *learning rate*-t csökkentettem, ha a hibafüggvény értéke nem csökkent, illetve a *batch* méretet is növelte közben. Az egymás melletti<sup>1</sup> mag-

---

<sup>1</sup> hulámhossz szerint



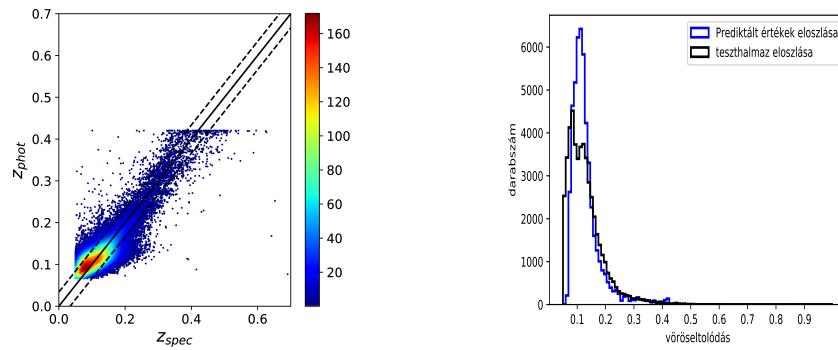
4.1. ábra. Az egyes neurális hálók magniúdópredikciója ábrázolva a valódi magnitúdóértékek függvényében.

nitúdók korrellálása, illetve a célváltozók és a bemeneti adatok hasonlóság miatt az volt a feltételezésem, hogy ugyanolyan architektúrájú neurális hálókat lehet használni minden egyik fényességérték számításához.

Az egyes hálók bemenetként megkapták az  $50 \times 50 \times 1$  méretű, az adott színszűrőn keresztül készített képet inputként. Az első réteg egy konvolúciós réteg volt, 32 darab  $2 \times 2$ -es filterből állt, alkalmaztam rá a *ReLU* aktivációt és utána egyből egy  $2 \times 2$  *MaxPooling*-ot. Utána a következő konvolúciós réteg 16 db  $2 \times 2$ -es filterből állt, *sigmoid* aktivációval, majd egy  $2 \times 2$ -es *MaxPooling* következett. A kijövő értékek kilapítása után következtek a teljesen összekötött rétegek, az első réteg 1024 neuronból állt, *sigmoid* aktivációval. Utána 15%-os *Dropout* regularizáció következett, ami a túllillesztés elkerülésére szolgált. Ezt követően 512 neuron, *ReLU* aktiváció és 10%-os *Dropout*, 256 neuron *ReLU*-val és végül az egy darab

neuron. A hibafüggvény *mean squared error* volt. A *random forest regressor* 250 fából állt, maximális mélysége 15, egy levélbe minimum 120-elemnek kellett kerülnie, egy szétválasztáshoz legalább 28 minta volt szükséges és a szétválasztás jóságát *mean squared error*-ral mérte.

A 4.1 ábrán látható, hogy az  $r$  illetve  $z$  magnitúdókat nagy hibával becsülte, de az adat-exploráció felfedte, hogy a *random forest*-nek ezek az adatok kevésbé fontosak. A magnitúdbecslésekben az *rmse*-k a következők lettek:  $rmse_u = 0.372$ ,  $rmse_g = 0.220$ ,  $rmse_r = 0.397$ ,  $rmse_i = 0.223$ , és  $rmse_z = 0.393$ , az alsó indexek a színt jelölik. Ekkora torzítással



4.2. ábra. A *Random forest* prediktált vöröseltolódás-értékei a spektroszkópiai vöröseltolódások függvényében és a két eloszlás.

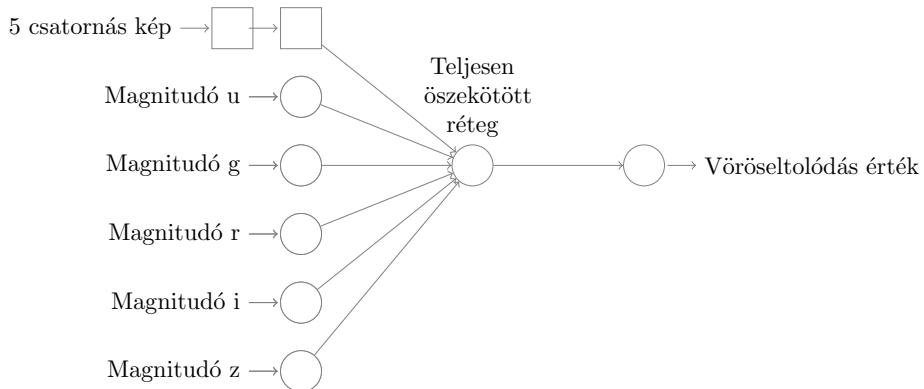
a magnitúdóadatokon a *random forest*  $rmse = 0.0338$  pontossággal tudott becsülni. A becsült és a spektroszkópiai vöröseltolódás-eloszlásokban látszódik a különbség, a prediktált eloszlás görbéje szűkebb, és a csúcs kicsit el van tolódva, ez szisztematikus hiba. A 4.2 bal oldali ábráján látszik, hogy 0.4-es vöröseltolódás körül vágása van, ez a véletlen erdők extrapolációs képességének hiánya illetve a levélképződéshez szükséges minimális mintaszám következménye.

Az *outlier rate*  $\frac{|z_{phot} - z_{spec}|}{1+z_{spec}} > 0.15$ -nél 0.082%, 0.05-nél 7.208% lett.

## 4.2. Konvolúciós háló kiegészítve becsült magnitúdókkal

Neurális hálók tudnak magnitúdókból vöröseltolódást becsülni, de érdekes kipróbálni, hogy hogyan teljesítenek akkor, ha a fényességértékeket csak extra információként kapják meg, a fő adat az ötcsatornás galaxiskép. A magnitúdó adatokat is képekből kell kinyerni, így a *random forest*-nél használt magnitúdó-becslő, tanított hálókat bekötöttem az új háló oldalába, közvetlenül a teljesen összekötött réteg előtt. A magnitúdóbecslőket nem tanítottam, hogy ne legyen feleslegesen túl nagy a paraméterszám, ami lassítaná a tanulást.

A tanítást az 1-es adathalmaz első 100 000 képén végeztem, a maradék 50 000-en a tesztelést, majd megvizsgáltam mire képes a 2-es adathalmaz utolsó 20 425 galaxisán. A 1-es

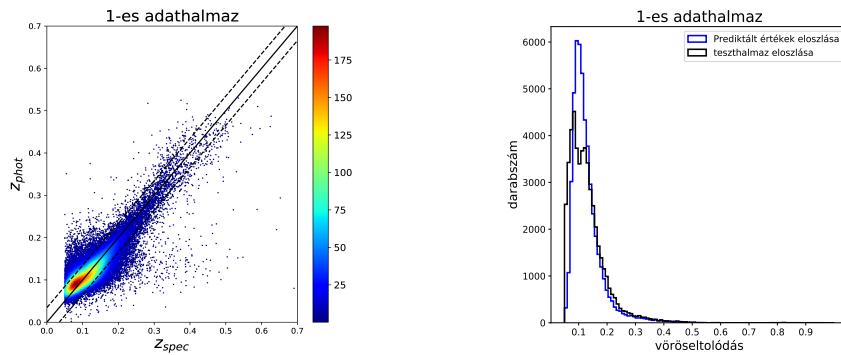


4.3. ábra. A háló sematikus modellje. A magnitúdóbecslő hálók kimeneteit és az ötcsatornás képből tanuló konvolúciós rész kimenete közvetlenül a teljesen összekötött réteg előtt simul egybe.

adathalmazon tanult hálót 10 *epoch* erejéig tanítottam még az 2-es halmon mielőtt kiértékeltem volna, mert az eltérő tanító- és tesztszett eloszlás problémát jelent.<sup>2</sup>

A haló bementként megkapta az ötcsatornás képet egybe, valamint az öt képet egyesével a magnitúdóbecslő hálóknak. Az első kettő konvolúciós réteg ami az ötcsatornás képen lett alkalmazva 28 darab  $3 \times 3$ -as méretű *filterekből* állt, utána egy  $2 \times 2$ -es *Maxpooling* réteg

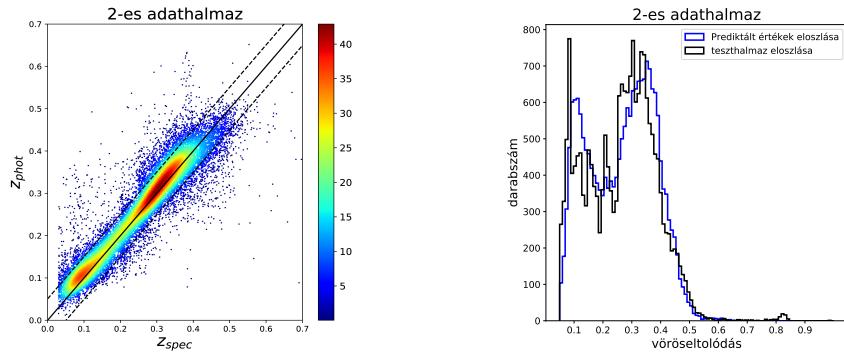
<sup>2</sup> A feliügyelt tanulásnál gyakran alkalmazott az az előfeltételezés, hogy a célváltozó tanító- és teszthalmaznak az eloszlása megegyezik. Elterő esetben, ha tudjuk, hogy a tanuló- és tesztminták nem azonos eloszlásból származnak *covariate shift* módszert lehet alkalmazni a korrigáláshoz [27].



4.4. ábra. A magnitúdóbecslésekkel kiegészített konvolúciós neurális háló becslései a spektroszkópiai vöröseltolódás-értékek függvényében az 1-es adathalmazon, és a vöröseltolódás-eloszlások.

következett, majd 2 darab  $64 \times 3$ -as konvolúciós *filtert* tartalmazó réteg, aztán  $2 \times 2$ -es *Maxpooling*. Utána 2 réteg 128 darab  $3 \times 3$ -as *filterből* álló konvolúció, majd  $3 \times 3$ -as *Maxpooling*, megint egy 128-szor  $3 \times 3$  konvolúciós réteg,  $2 \times 2$ -es *Maxpooling* és végül egy 28 darabos  $3 \times 3$  *filteres* réteg. A rétegekben végig *ReLU* aktivációt alkalmaztam. A vektorrá lapítás után kötöttem be az 5 becsült magnitúdóértéket, és következtek a teljesen összekötött réteg. Az első 256 neuronból állt, és *sigmoid* aktivációt alkalmaztam rá, utána egy 50 neuronból álló réteg *ReLU* aktivációval, majd a kimeneti neuron következett, aktiváció nélkül.

Az 1-es teszthalmazon 50 000 képen tesztelve a  $rmse = 0.0345$  lett, ami egy kicsit gyengébb, mint a *random forest* becslése. A 4.4 ábrát összhasonlítva a *random forest* prediktálásával a 4.2 ábrán, látható, hogy a neurális háló tud  $\approx 0.4$  feletti vöröseltolódás-értékeket prediktálni, viszont több *outlier* adatpont van a nagyobb spektrális vöröseltolódások alulbecslése miatt. Alacsony vöröseltolódásokon ez a modell is szisztematikusan felülbecsül. A lényegesen más vöröseltolódás-eloszlású 2-es adathalmazon 20 425 galaxison tesztelve a  $rmse = 0.0505$ , viszont átfedés volt a tanuló- és tesztszett között, így az átfedő galaxisok hibájának a becslését nullának véve a  $rmse = 0.056$ , ami nem túl jó eredmény. A hiba nagysága fakadhat abból, hogy a magnitúdóbecslő hálók is más magnitúdóeloszláson tanultak, így a becslésük pontatlanabb. Megnézve a 4.5 ábrát, látható, hogy a vöröseltolódás-eloszlásokban



4.5. ábra. A magnitúdóbecslésekkel kiegészített konvolúciós neurális háló becslései a spektroszkópiai vöröseltolódás-értékek függvényében az 2-es adathalmazon, és a vöröseltolódás-eloszlások.

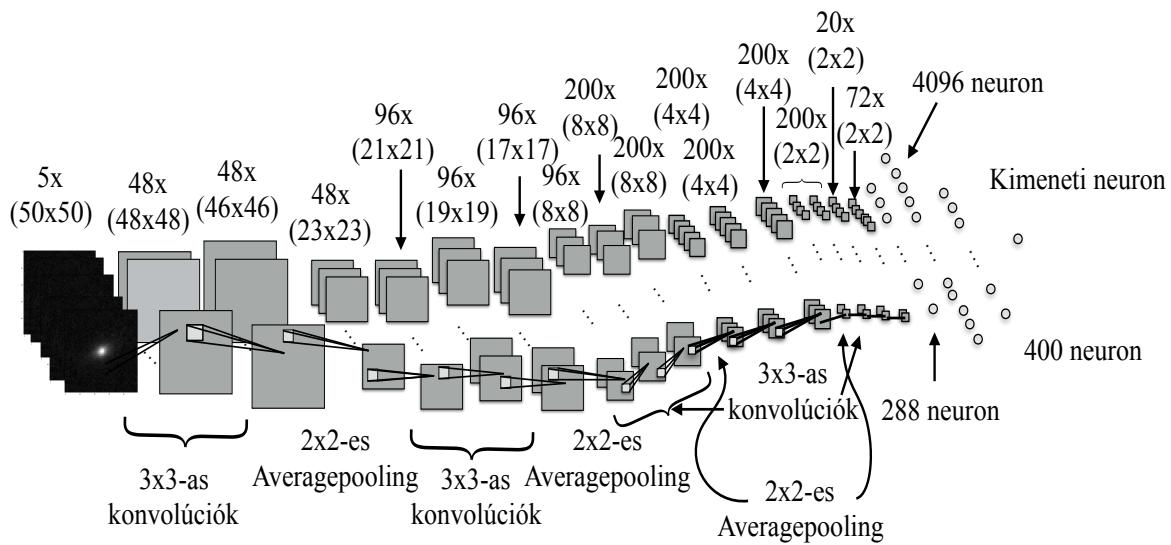
található két csúcs körül enyhén felülbecsül, között viszont jól eltalálja az értékeket. 1-es halmazra az *outlier* arányok 0.15-nél 0.238%, gyengébb mint a *random forestnél*, 0.5-nél 6.564%, ami viszont kicsivel jobb nála. A 2-es adathalmazra rendre 0.8661% és 10.83% az *outlier* arány.

### 4.3. Egy mély konvolúciós háló

A *random foresttel* láthatóan működött a kinyert adatokból a vöröseltolódás-becslés, de kézenfekvőbb olyan módszert megvalósítani, ami kitalálja, hogy mi a fontos tulajdonság a prediktálás szempontjából, és ezt ki is vonja automatikusan a képből.

A képi adatokból jól prediktáló konvolúciós hálók<sup>3</sup> architektúrájához hasonló modellt célszerű készíteni kiindulásként, de ezeket mind osztályozó, objektumfelismerő problémák megoldására használták, tanítására több adat és számítási kapacitás állt rendelkezésre, ezért a vöröseltolódás-becslő háló nem lesz olyan *mély*. A háló tanítását a 2-es adathalmaz előző 80 000 képével végeztem, majd a maradék 20 425 képpel teszteltem. Valamint végeztem tesztelést az összehasonlítás édekében az 1-es adathalmaz 100 000-től 150 000-ig elhelye-

<sup>3</sup> például VGG16, GoogLeNet

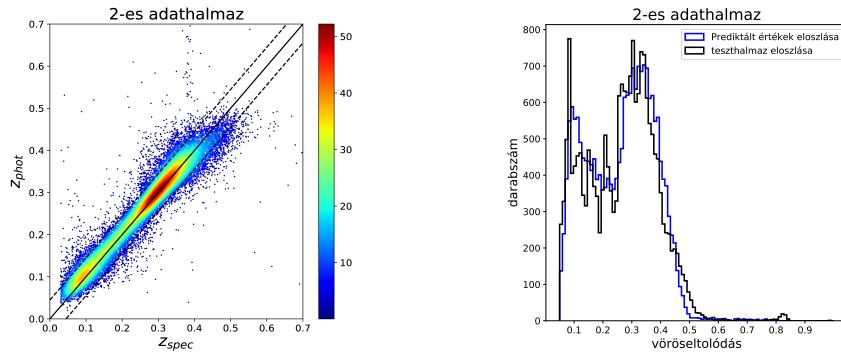


4.6. ábra. A mély konvolúciós hálóban az egyes rétegek alakulása, a háló szerkezete.

zekdő képeivel is.

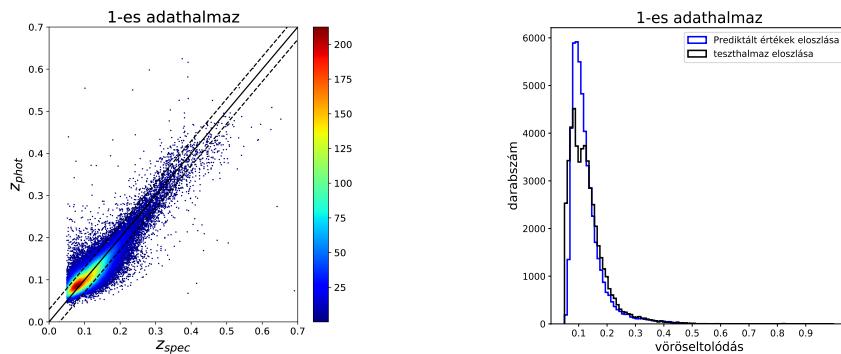
A modell architektúrája a következő volt: a bemenet az  $50 \times 50 \times 5$  méretű kép volt, utána kétszer egymás után, 48 darab  $3 \times 3$ -as filterből álló konvolúciós réteg, majd  $2 \times 2$  Averagepooling. Utána következett 3 egymás utáni 96  $3 \times 3$  filterből álló réteg, majd megint egy  $2 \times 2$ -es Averagepooling, majd kétszer 200 darab  $3 \times 3$  filter,  $2 \times 2$  Averagepooling, egy 200 filteres,  $3 \times 3$  konvolúciós réteg, egy BatchNormalizáció, majd megint egy 200 darab,  $3 \times 3$ -as filteres réteg. Aztán  $2 \times 2$ -es Averagepooling, majd egy 200 és egy 20 darabos  $3 \times 3$  filteres réteg, utána 72 darab  $1 \times 1$  filteres dimenzió csökkentő konvolúciós réteg. Mindegyik rétegen ReLU aktivációt alkalmaztam. A teljesen összekötött rész 4096 neuronnal kezdődött, majd 400 neuron a következő rétegen, mind ReLU aktivációval, végül az 1 neuronból álló kimeneti réteg, aktiváció nélkül.

A tanításhoz a megnövekedett paraméterszám miatt több epochra volt szükség, mint a magnitúdóbecslésnél, összesen 100 epoch-ra, a learning rate-ot csökkentettem, ha nem csökkent a hiba értéke, illetve a batch size-t növelte. A 2-es adathalmazon a 20 425 galaxison tesztelve a  $rmse = 0.0447$  lett. A 4.7 ábra bal oldalán látható, hogy a prediktált értéke-



4.7. ábra. A konvolúciós neurális háló becslései a spektroszkópiai vöröseltolódás-értékek függvényében a 2-es adathalmazon, és a vöröseltolódás-eloszlások.

ket jelölő pontok szimmetrikusak a  $45^\circ$ -os egyenesre a 0.2 és 0.45 közötti vöröseltolódás-tartományon, nincsen szisztematikus hibája. A 0.2-nél kisebb vöröseltolódásokat gyengén fejlőlbecsli, a 0.45-nél nagyobb vöröseltolódásúakat pedig alul. Érdemes volt megnézni, hogyan teljesít a háló az 1-es adathalmazon, ahol több alacsony vöröseltolódású galaxist tartalmaz a tesztszett, valamint így összehasonlítható a *random forest* módszerrel. Mielőtt prediktáltattam volna a modellt, tanítottam 10 *epoch*-ot az 1-es adathalmazon.



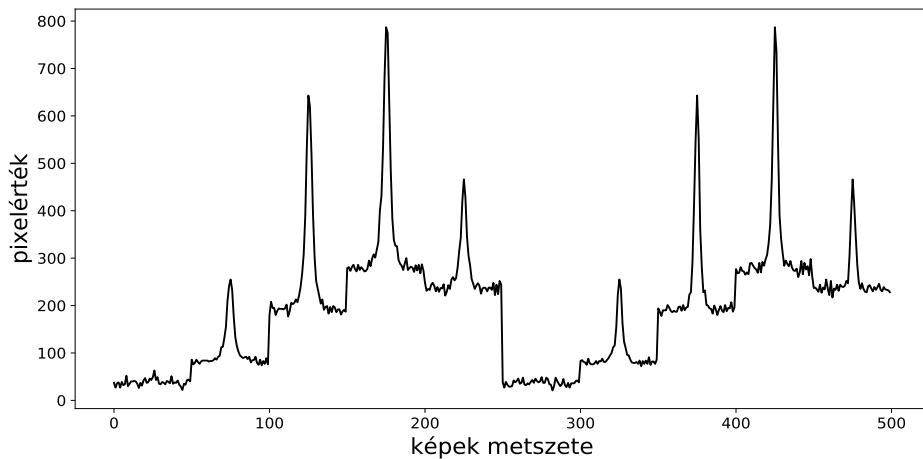
4.8. ábra. A konvolúciós neurális háló becslései a spektroszkópiai vöröseltolódás-értékek függvényében a 1-es adathalmazon, és a vöröseltolódás-eloszlások.

A tesztadatokon  $rmse = 0.0295$  hibát produkált, ami jobb mint a *random forest*é, viszont

az alacsony vöröseltolódásoknál még mindig enyhén felülbecsül. Megjegyezendő, hogy az 2-es tanítóhalmaz és az 1-es teszthalmaz 2.6%-ban átfed, ez okozhat  $rmse$  csökkenést, de nem nagy mértékben, a tanítóhalmazban a modell által már látott galaxisok vöröseltolódásbecslésének hibáját 0-nak véve, a csak teszthalmazban megkapott mintákon a  $rmse = 0.0298$ . Az *outlier rate*-ek az 1-es adathalmazra rendre 0.104% és 4.358%, a 2-es adathalmazra pedig 0.484% és 7.529%, amelyek jobbak, mint a magnitúdóbecslésekkel kiegészített rendszeré.

#### 4.4. Teljesen összekötött neurális háló

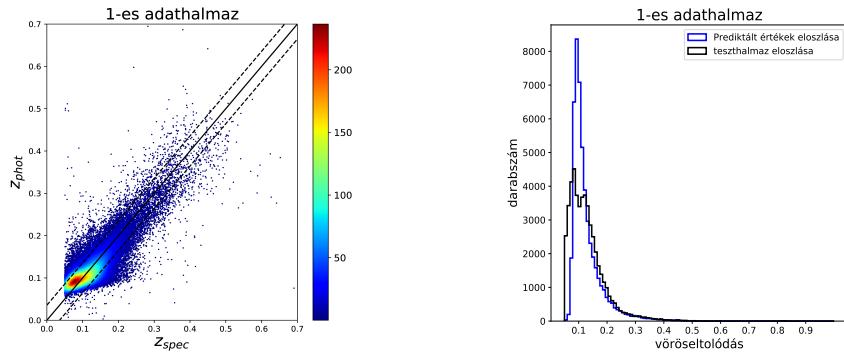
Érdemes megvizsgálni, hogy a képek redukciójával mennyi használható információ marad a vöröseltolódásbecsléshez, hogy csupán a galaxisok kiterjedése és fényessége, illetve fényességének az objektum közepéhez viszonyított csökkenése mennyi információt tartalmaz. Ehhez az egyes színszűrők által készített képek két irányból vett, közepén átmenő metszetét használtam, így az  $50 \times 50 \times 5$  méretű képből egy  $500 \times 1$ -es vektor lett.



4.9. ábra. A képek metszete egymás mellé téve. Sorrendben az  $u$ ,  $g$ ,  $r$ ,  $i$ ,  $z$  horizontális metszete, majd vertikális metszetiük.

A bemenet formája miatt teljesen összekötött hálót készítettem, aminek az architektúrája a következő volt: az 500 neuronból álló bemeneti réteg után egy 1500 neuronból álló réteg

következett, majd 5%-os *Dropout*, utána 800 neuronos réteg, 5%-os *Dropout*, majd még egy 800-as réteg és egy *Dropout*. Utána a méretcsökkentő, 400 neuronos, majd egy 65 neuronos réteg, és végül az egyetlen kimeneti neuron. A rétegekre *tangens hiperbolikusz* aktivációs függvényt alkalmaztam. A tanításhoz  $\approx 200$  epoch volt szükséges, de az adatok relatíve kis



4.10. ábra. A teljesen összekötött neurális háló becslései a spektroszkópiai vöröseltolódás-értékek függvényében a 1-es adathalmazon, és a vöröseltolódás-eloszlások.

mérete és az előző modellekhez képest alacsonyabb szabad paraméterszám miatt ez nem volt sok idő. Az 1-es adathalmaz első 100 000 képén végeztem a tanitást, és az utolsó 50 000 képen a tesztelést. A 4.10ábrán látható, hogy becslései viszonylag szimmetrikusak a  $45^\circ$ -os egyenesre, de nem pontosabb mint a konvolúciós háló. A  $rmse = 0.0355$  lett, ami nem rossz eredmény a képek redukciójának mértékének fényében. Az *outlier rate* 0.15-nél 0.174%, 0.05-nél 7.112% volt.

## 4.5. Eredmények összegzése

Az módszerek kényelmesebb összehasonlítása érdekében a lenti táblázatban összegzem az egyes eredményeket. A táblázati adatokból és az ábrákon is látható, hogy a mély konvolúciós háló teljesített a legjobban, mind a kettő adathalmazt figyelembe véve is.

módszer	<i>rmse</i>	$\frac{ z_{photo} - z_{spec} }{1+z_{spec}} > 0.15$	$\frac{ z_{photo} - z_{spec} }{1+z_{spec}} > 0.05$
1-es teszthalmaz			
Neurális hálók és <i>random forest</i>	0.0338	0.082%	7.208%
Konvolúciós háló magnitúdókkal	0.0345	0.238%	6.564%
Mély konvolúciós háló	0.0298	0.104%	4.358%
Teljesen összekötöt	0.0355	0.174%	7.112%
2-es teszthalmaz			
Konvolúciós háló magnitúdókkal	0.0560	0.861%	10.83%
Mély konvolúciós háló	0.0447	0.484%	7.529%

4.1. táblázat. Az eredmények összefoglalása.

## 5. fejezet

### Összefoglalás

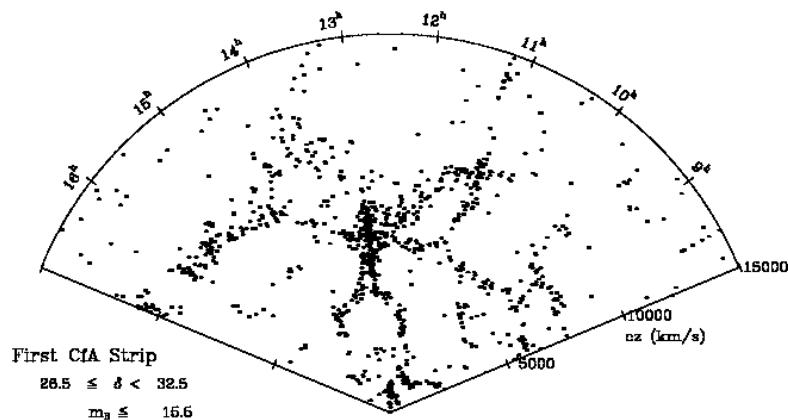
Szakdolgozatomban bemutattam a gépi tanulási módszerek alkalmazásait vöröseltolódás-becslésre. Ismertettem a *random forest* algoritmus és a neurális hálók működését és használthatóságát, áttekintettem a fotometrikus vöröseltolódás-becslő módszereket, valamint előállítottam két nagyobb tanítóhalmazt is, amiket a további munkára illetve más feladatokra is lehet használni.

Elkészítettem és prezentáltam négy módszert, melyekkel lehetőség van vöröseltolódást becsülni a galaxisok képéből. Az általam megvalósított módszerek közül a mély konvolúciós háló prediktálásainak a *root mean squared errorja* megközelítette a *DR 7*-en alkalmazott módszerek hibáját. Megjegyezendő, hogy a neurális hálók architektúrájának megalkotása heurisztikus folyamat volt, ami nem garantálta a legjobb megoldást, csak egyet a jók közül, ezért az architektúrák finom hangolásával illetve merészebb regularizációs és normalizációs rétegek használatával lehet még növelni a pontosságot.

Az, hogy a gépi tanuló eljárások jól tudnak teljesíteni tudományos célú, komplexebb problémákon, ezek fejlődése és a növekvő mennyiséggű elérhető mérési adat előrevetítik, hogy a jövőben még nagyobb szerep jut nekik a tudományban.

# Függelék

## A.1. Margaret Gellerék égtérképe



1. ábra. Az ábrán látható, hogy a sűrűség eloszlás nem egyeneletes, megfigyelhetőek üregek és egy nagy fal. A kép forrása [28]

## B.2. Az adattáblához használt lekérdezés

Az 1-es adathalmaz adattáblájához használt lekérdezés. Ennek az eredménye  $\approx 580\,000$  objektumot adott, ebből az első 150 000-et használtam fel.

```
SELECT TOP 1E6
    p.objId, p.run, p.rerun, p.camcol, p.field, p.rowc_r,
    p.colc_r, p.rowc_u, p.colc_u, p.rowc_g, p.colc_g, p.rowc_i,
    p.colc_i, p.rowc_z, p.colc_z,
    p.u, p.g, p.r, p.i, p.z,
    s.z as redshift, s.zErr
FROM PhotoObjAll AS p
JOIN SpecObj AS s ON s.bestobjid = p.objid
WHERE
    p.u BETWEEN 15 AND 22
    AND p.g BETWEEN 15 AND 22
    AND p.r BETWEEN 15 AND 22
    AND p.i BETWEEN 15 AND 22
    AND p.z BETWEEN 15 AND 22
    AND (s.objTypeName = 'Galaxy' )
    AND s.z between 0.05 AND 1

    AND p.colc_u BETWEEN 25 and 2023
    AND p.colc_g BETWEEN 25 and 2023
    AND p.colc_r BETWEEN 25 and 2023
    AND p.colc_i BETWEEN 25 and 2023
    AND p.colc_z BETWEEN 25 and 2023
    AND p.rowc_u BETWEEN 25 and 1463
    AND p.rowc_g BETWEEN 25 and 1463
    AND p.rowc_r BETWEEN 25 and 1463
```

---

```
AND p.rowc_i BETWEEN 25 and 1463
AND p.rowc_z BETWEEN 25 and 1463
```

### B.3. A képek megtalálása, kivágása és elmentése

Kellett egy függvény ami előállítja az elérési útvonalát a képnek az adatokból.

---

```
def filePath(run, camcol, field, rerun, color):
    if run >= 1000 and field >= 100:
        image_file =
            '/home/idores/workspace/sdss_das/das2/imaging/%i' %run
            +'/' +str(rerun) +'/corr/' +str(camcol) +'/fpC-00%i' %run + '-' +color +
            str(camcol) +'-0' +str(field) +'.fit.gz'
    if run >= 1000 and field < 100:
        image_file =
            '/home/idores/workspace/sdss_das/das2/imaging/%i' %run
            +'/' +str(rerun) +'/corr/' +str(camcol) +'/fpC-00%i' %run + '-' +
            color+str(camcol) +'-00' +str(field) +'.fit.gz'
    if run < 1000 and field < 100:
        image_file =
            '/home/idores/workspace/sdss_das/das2/imaging/%i' %run
            +'/' +str(rerun) +'/corr/' +str(camcol) +'/fpC-000%i' %run + '-' +color +
            str(camcol) +'-00' +str(field) +'.fit.gz'
    if run < 1000 and field >= 100:
        image_file =
            '/home/idores/workspace/sdss_das/das2/imaging/%i' %run
            +'/' +str(rerun) +'/corr/' +str(camcol) +'/fpC-000%i' %run + '-' +color +
            str(camcol) +'-0' +str(field) +'.fit.gz'
    return image_file
```

---

Ez a függvény kivágja az általunk megadott méretű részt az általunk megadott koordináták

körül.

---

```
def cutOut(rowc, colc,image_data, outputSize):
    halfWidth = int(outputSize/2)
    cutout = image_data[rowc-halfWidth:rowc+halfWidth,
                        colc-halfWidth:colc+halfWidth].copy()
    return cutout
```

---

Az előző két függvény felhasználásával előallítja a képet, elmenti, és az elérési útvonalát egy *python dictionarybe* menti.

---

```
from astropy.io import fits

for i in range(0, len(data)):
    image_file = filePath(data.run[i], data.camcol[i],
                          data.field[i], data.rerun[i], 'u')

    hdu_list = fits.open(image_file)
    image_data = fits.getdata(image_file)
    image_data -= hdu_list[0].header['Softbias']# ez a softbias
    filename = '../scratch/' + str(data.run[i]) +
               '/' + str(data.rerun[i]) + '/' + str(data.camcol[i]) + '/'
               'u' + str(data.objId[i]) + '.fits'
    a = cutOut(data.rowc_u[i], data.colc_u[i], image_data, 50)
    pathDictU50.update({str(data.objId[i]):filename})
    zDictU50.update({str(data.objId[i]):data.z[i]})

    hdu_list.close()

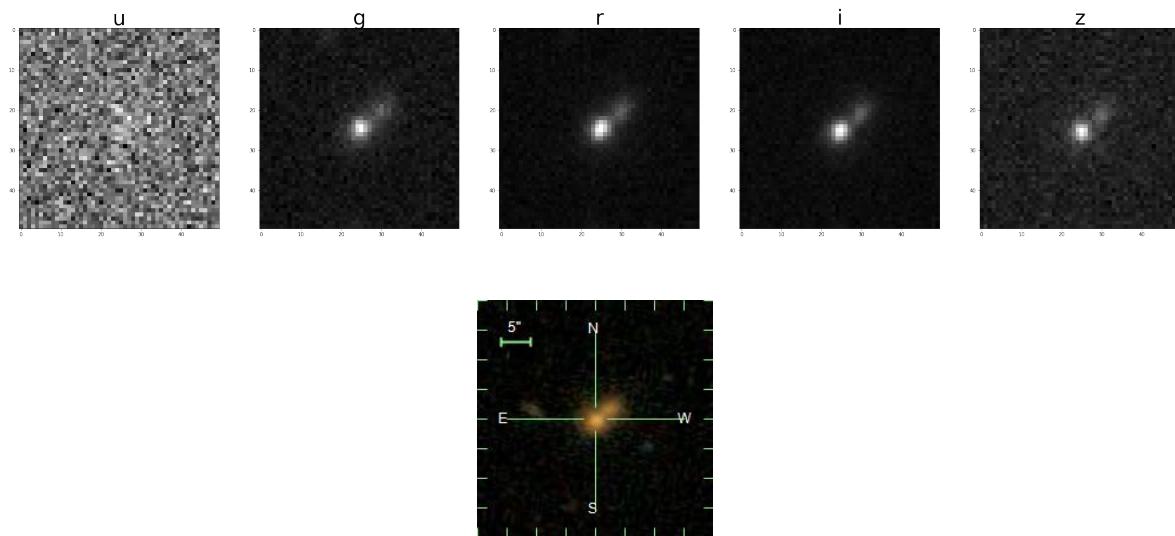
    saveToDir(filename,a)

save_obj(pathDictU50, 'pathU50m')
```

---

## B.4. A képek ellenőrzése

A tanulóhalmazunk helyes indexelését úgy vizsgálhatjuk, hogy véletlenszerűen megjelenítünk galaxisokat, kiíratjuk az elméletileg hozzáartozó objektum azonosítót, az objektum azonosító alapján megkeressük az SDSS *DR7 explore* felületén[29] a galaxis képét, így szemrevételezzel összehasonlíthatóak.



2. ábra. Felül az 5 színszűrőn keresztül készített kép szürkeárnyalatosan, alul az ugyanehhez az objektumhoz tartozó, [29] található színessé transzformált jpeg kép.

# Irodalomjegyzék

- [1] M.J, Geller, et al. “A Slice of the Universe.” SAO/NASA ADS: ADS Home Page, 1 Mar. 1986, [adsabs.harvard.edu/doi/10.1086/184625](http://adsabs.harvard.edu/doi/10.1086/184625).
- [2] Szalay, A. S., et al. “Redshift Survey with Multiple Pencil Beams at the Galactic Poles.” Proceedings of the National Academy of Sciences, vol. 90, no. 11, 1993, pp. 4853–4858., doi:[10.1073/pnas.90.11.4853](https://doi.org/10.1073/pnas.90.11.4853).
- [3] Z. Frei and A. Patkós, Inflációs Kozmológia: (Typotex, Budapest, 2005).
- [4] Baum, W. A.: 1962, Problems of Extra-Galactic Research, Proceedings from IAU Symposium no. 15. Edited by George Cunliffe McVittie. International Astronomical Union Symposium no. 15, Macmillan Press, New York, p.390
- [5] “Photometric Redshifts.” NASA/IPAC Extragalactic Database - NED, [ned.ipac.caltech.edu/level5/Glossary/Essay\\_photredshifts.html](http://ned.ipac.caltech.edu/level5/Glossary/Essay_photredshifts.html).
- [6] Koo, D. C. “Optical Multicolors - A Poor Person’s Z Machine for Galaxies.” The Astronomical Journal, vol. 90, 1985, p. 418., doi:[10.1086/113748](https://doi.org/10.1086/113748).
- [7] Bruzual, G., and S. Charlot. “Stellar Population Synthesis at the Resolution of 2003.” Monthly Notices of the Royal Astronomical Society, vol. 344, no. 4, 2003, pp. 1000–1028., doi:[10.1046/j.1365-8711.2003.06897.x](https://doi.org/10.1046/j.1365-8711.2003.06897.x).
- [8] Hildebrandt, H., et al. “PHAT: PHoto-ZAccuracy Testing.” Astronomy & Astrophysics, vol. 523, 2010, doi:[10.1051/0004-6361/201014885](https://doi.org/10.1051/0004-6361/201014885).

- [9] Connolly, A. J., et al. “Slicing Through Multicolor Space: Galaxy Redshifts from Broadband Photometry.” *The Astronomical Journal*, vol. 110, 1995, p. 2655., doi:10.1086/117720.
- [10] Csabai, I., et al. “The Application of Photometric Redshifts to the SDSS Early Data Release.” *The Astronomical Journal*, vol. 125, no. 2, 2003, pp. 580–592., doi:10.1086/345883.
- [11] Carliles, S., et al. “Random Forests For Photometric Redshifts.” *The Astrophysical Journal*, vol. 712, no. 1, 2010, pp. 511–515., doi:10.1088/0004-637x/712/1/511.
- [12] Collister, Adrian A., and Ofer Lahav. “ANNz: Estimating Photometric Redshifts Using Artificial Neural Networks.” *Publications of the Astronomical Society of the Pacific*, vol. 116, no. 818, 2004, pp. 345–351., doi:10.1086/383254.
- [13] Csabai, I., et al. “Multidimensional Indexing Tools for the Virtual Observatory.” *Astronomische Nachrichten*, vol. 328, no. 8, 2007, pp. 852–857., doi:10.1002/asna.200710817.
- [14] Ball, Nicholas M., and Robert J. Brunner. “Data Mining And Machine Learning In Astronomy.” *International Journal of Modern Physics D*, vol. 19, no. 07, 2010, pp. 1049–1106., doi:10.1142/s0218271810017160.
- [15] “Random Forests Leo Breiman and Adele Cutler.” Statistics at UC Berkeley, [www.stat.berkeley.edu/~breiman/RandomForests/](http://www.stat.berkeley.edu/~breiman/RandomForests/).
- [16] <http://www.globalsoftwaresupport.com/wp-content/uploads/2018/02/ggff5544hh.png>
- [17] Sadeh, I., et al. “ANNz2: Photometric Redshift and Probability Distribution Function Estimation Using Machine Learning.” *Publications of the Astronomical Society of the Pacific*, vol. 128, no. 968, 2016, p. 104502., doi:10.1088/1538-3873/128/968/104502.

- [18] L., Samuel, et al. “Don’t Decay the Learning Rate, Increase the Batch Size.” SAO/NASA ADS: ADS Home Page, 1 Nov. 2017, [adsabs.harvard.edu/cgi-bin/bib\\_query?arXiv%3A1711.00489](http://adsabs.harvard.edu/cgi-bin/bib_query?arXiv%3A1711.00489).
- [19] Firth, Andrew E., et al. “Estimating Photometric Redshifts with Artificial Neural Networks.” *Monthly Notices of the Royal Astronomical Society*, vol. 339, no. 4, 2003, pp. 1195–1202., doi:[10.1046/j.1365-8711.2003.06271.x](https://doi.org/10.1046/j.1365-8711.2003.06271.x).
- [20] D’Isanto, A., et al. “Return of the Features. Efficient Feature Selection and Interpretation for Photometric Redshifts.” *Astronomy & Astrophysics*, 2018, doi:[10.1051/0004-6361/201833103](https://doi.org/10.1051/0004-6361/201833103).
- [21] Hoyle, B. “Measuring Photometric Redshifts Using Galaxy Images and Deep Neural Networks.” *Astronomy and Computing*, vol. 16, 2016, pp. 34–40., doi:[10.1016/j.ascom.2016.03.006](https://doi.org/10.1016/j.ascom.2016.03.006).
- [22] D’Isanto, A., and K. L. Polsterer. “Photometric Redshift Estimation via Deep Learning.” *Astronomy & Astrophysics*, vol. 609, 2018, doi:[10.1051/0004-6361/201731326](https://doi.org/10.1051/0004-6361/201731326).
- [23] Abazajian, Kevork N., et al. “The Seventh Data Release Of The Sloan Digital Sky Survey.” *The Astrophysical Journal Supplement Series*, vol. 182, no. 2, 2009, pp. 543–558., doi:[10.1088/0067-0049/182/2/543](https://doi.org/10.1088/0067-0049/182/2/543).
- [24] Zasowski, Gail. Science Blog from the SDSS, [blog.sdss.org/2018/02/21/sdss-v-is-underway/](http://blog.sdss.org/2018/02/21/sdss-v-is-underway/).
- [25] “SciServer – Collaborative Data-Driven Science.” SciServer, [www.sciserver.org/](http://www.sciserver.org/).
- [26] Verevkin, A. O., et al. “The Non-Uniform Distribution of Galaxies from Data of the SDSS DR7 Survey.” *Astronomy Reports*, vol. 55, no. 4, 2011, pp. 324–340., doi:[10.1134/s1063772911020089](https://doi.org/10.1134/s1063772911020089).

- [27] Mcgaughey, Georgia, et al. “Understanding Covariate Shift in Model Performance.” F1000Research, vol. 5, 2016, p. 597., doi:10.12688/f1000research.8317.3.
- [28] Huchra, John P. “THE CfA REDSHIFT SURVEY.” [www.cfa.harvard.edu/~dfabricant/huchra/zcat/](http://www.cfa.harvard.edu/~dfabricant/huchra/zcat/).
- [29] <http://skyserver.sdss.org/dr7/en/tools/explore/obj.asp>

# Nyilatkozat

**Név:** Horváth Bendegúz

**ELTE Természettudományi Kar, szak:** Fizika BSc

**Neptun azonosító:** ZNL3LK

**Szakdolgozat címe:** Gépi tanulási módszerek a fotometrikus vöröseltollódás-becslésben

A **szakdolgozat** szerzőjeként fegyelmi felelősségem tudatában kijelentem, hogy a dolgozatom önálló munkám eredménye, saját szellemi termékem, abban a hivatkozások és idézések standard szabályait következetesen alkalmaztam, mások által írt részeket a megfelelő idézés nélkül nem használtam fel.

Budapest 2018. május 31.

---