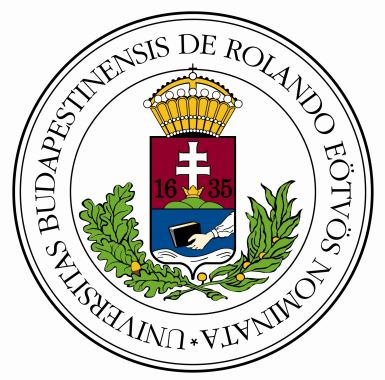
EÖTVÖS LORÁND TUDOMÁNYEGYETEM
TERMÉSZETTUDOMÁNYI KAR

Vas Bernadett

Retinaképek klasszifikálása konvolúciós hálóval

Önálló projekt III.

Témavezető
Lukács András



Budapest, 2022

Tartalomjegyzék

Bevezetés

Az önálló projekt harmadik félévében retinafelvételeket tartalmazó adathalmazokon végeztem egy szembetegség, a diabetikus retinopátia detektálását, illetve súlyosságánk megállapítását. A vizsgált feladatban minden retinafelvétel rendelkezik egy címkével, ami azt írja le, vannak-e jelei a betegségnek a retinán, és ha igen, az mennyire előrehaladott állapotú. A célunk olyan mélytanulási algoritmust építeni, ami megjósolja egy input retinafelvételre a betegség súlyosságát.

Az adathalmazban megtalálható címkék a következők:

- 0-s osztály: teljesen egészséges
- 1-es osztály: enyhe
- 2-es osztály: közepesen súlyos
- 3-as osztály: súlyos
- 4-es osztály: poliferatív

Az alapvető hozzáállás az lenne, hogy ötosztályú klasszifikációként értelmezzük a feladatot. Azonban figyelembe kell vennünk azt a tényt, hogy az osztálycímkék skálaszerűek, tehát egyfajta sorrendiség jelenik meg közöttük, mivel egy romló folyamatot írnak le. Ezt az információt nem szabad elhanyagolni, mert a modell által vétett hibák nem tekinthetőek ugyanolyan mértékűnek. Például ha az x adatpont valódi címkéje 1-es, és a modell a 2-es súlyossági osztályba sorolja, az kevésbé rossz, mintha a 4-esbe sorolná.

Az adathalmazban tehát olyan osztálycímkék tartoznak a adatpontokhoz, amik bár diszkrétek, mégis van köztük rendezettségi kapcsolat, és így valamiféle távolság is. Hogyan is érdemes hozzáállni ehhez a feladathoz? A többosztályú klasszifikációval egyrészt az a baj, hogy független osztályokat feltétlez, másrészt pedig ezzel a megközelítéssel elveszítenénk az osztályok közötti rendezettségi információt. Ekkor minden félreklasszifikálást ugyanolyan mértékűnek tekintenénk, valamint azt feltételeznénk, hogy a modell egy hibázásnál ugyanolyan valószínűséggel mondja a valódi címkén kívűl bármely másikat, de ez sem lesz feltétlenül igaz. Ha a modell egy input képre rossz osztályt jósol, akkor nagyobb valószínűséggel mondja helyette valamelyik hozzá közelebb álló, szomszédos címkét, mint egy, az adott valódi címkétől távolabb levő osztályt. Ösztönösen jönne az ötlet, hogy használjunk regressziót a feladatra, azonban ennek a megközelítésnek a hátránya, hogy folytonos numerikus értékként tekint a címkékre, valamint azt feltételezi, hogy a szomszédos osztályok ugyanolyan távol vannak egymástól, ami szintén nem feltétlenül igaz.

Azt a fajta feladatot, amikor a felügyelt tanulás során diszkrét, rendezett címkéink vannak, az irodalomban *ordinal classification*-nek hívják. Erre a feladatra több példa is adható. Betegségek súlyosságának

osztályozásán kívűl ilyen feladat még az *age estimation*, tehát az emberek korának megjóslása, illetve különböző termékek és szolgáltatások értékelése is. Az önálló projektem utolsó félévében ezen feladat irodalmát dolgoztam fel, és azt vizsgáltam, melyik a leghatékonyabb módszer a diabetikus retinopátia megfelelő klasszifikációjához. Az első fejezetben áttekintem az ordinal classification megoldási módszereit, majd a második fejezetben prezentálom a retinafelvételeken végzett mérési eredményeimet.

1. fejezet

Ordinal classification

1.1. Bináris feladatokra bontás

Hagyományos gépi tanulási algoritmusoknál az ordinal classification megközelítése gyakran bináris részfeladatokra bontással valósul meg. Ezen motiváció alapján Niu és tsai. [6] konvolúciós hálókra is alkalmazták ezt az ötletet. A megvalósításukban egyetlen hálót tanítottak, aminek az utolsó sűrű rétege után $K-1$ bináris klasszifikátort helyeztek el, melyek mindegyike két neuronból áll. A t-edik klasszifikátor feladata eldönteni, hogy $\mathbf{x}^{[i]}$ adatpont y_i címkéje nagyobb-e mint r_t, azaz a t rangú címke. Mindegyik bináris feladatnál egy valószínűséget prediktálunk, ezt használjuk fel a veszteségfüggvényhez és a megjósolt címke kiszámolásához is. A tanításnál használt veszteségfüggvény:

$$-\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\lambda_t \mathbb{1}\{o_i^t = y_i^t\} log(p(o_i^t))$$

ahol λ_t a t-edik feladat fontossági paramétere, N az adatpontok száma, o_i^t pedig a t-edik feladat kimenete. A módszer a címkéknek egy specifikus, ordinal one-hot vektoros elkódolását használja. Ez azt jelenti, hogy az y_i címke vektoros reprezentációjában a t-edik koordináta azt jelöli, az y_i megugorja-e a t-edik súlyossági szintet: $y_i^t = \mathbb{1}\{y_i > r_t\}$. Mivel $T = K-1$, ahol K az osztályok számát jelöli, ezért y_i egy $K-1$ hosszú vektorként áll elő.

A címkére való predikció számolása a következőképpen történik. A t-edik bináris klasszifikátornál legyen $\hat{y}_i^t$ az $\mathbf{x}^{[i]}$ adatpontra a $P(y_i > r_t)$ valószínűség. Ekkor a predikció

$$q = 1 + \sum_{t=1}^{K-1} \mathbb{1}\{\hat{y}_i^t \geq 0.5\}.$$

1.2. CORAL (Consistent Rank Logits)

A bináris részfeladatokra bontással az a probléma, hogy a modell kimeneti valószínűségei, mivel egymástól függetlenül számolódnak, nem feltétlenül konzisztensek. Ez azt jelenti, hogy nem mutatnak értelmezhető mintázatot, például egy osztályra kaphatunk nagy valószínűséget, miközben az őt a sorrendben megelőző osztályra pedig alacsony értéket. Ez nem felel meg annak az elvárásnak, hogy ha $y_i > r_t$ akkor $y_i > r_l$ ahol $l \leq t-1$. Ezenkívül ha orvosi célra használjuk a modellt, akkor azt is

elvárnánk, hogy az egyes osztályokra adott kimeneti valószínűségek hasznos információt szolgáltassank a szakorvosnak, minél inkább magyarázható legyen a kapott predikció, de ez nem fog teljesülni ha nem monoton valószínűségeket kapunk. Egy modell kimeneti valószínűségei akkor konzisztensek, ha $P(y_i > r_1) \geq P(y_i > r_2) \geq ... \geq P(y_i > r_{K-1})$ igaz rájuk. Ezen problémára a CORAL framework [3] kínál megoldást.

A módszer a címkék ordinal one-hot vektoros elkódolását használja, tehát az y_i címke vektoros formájában a k-adik koordináta azt jelöli, az y_i megugorja-e a k-adik súlyossági szintet: $y_i^k = \mathbb{1}\{y_i > r_k\}$. A konvolúciós modell utolsó rétegében $K-1$ bináris klasszifikátort tanítunk (K az osztályok száma), amiknek a w súlyparaméterei megegyeznek, de a b torzítások klasszifikátoronként eltérnek. Ezek után mindegyik klasszifikátorra alkalmazunk egy szigmoid föggvényt, hogy minden kimenet 0 és 1 közé essen, és ezeket az értékeket valószínűségekként interpretáljuk. Tehát a k-adik neuron kimenete azt a valószínűséget fogja megadni, hogy az input súlyossága nagyobb-e mint a k-adik rang:

$$P(y_i > r_k) = P(y_i^k = 1) = \sigma(\sum_{j=1}^{m} w_j a_j + b_k) = \sigma(g(\mathbf{x}^{[i]}, \mathbf{W}) + b_k)$$

ahol $a = (a_1, ..., a_m)$ jelöli az utolsó előtti réteg kimeneti vektorát. A tanítás alatt a $K-1$ bináris klasszifikátor súlyozott cross-entropy függvényét minimalizáljuk:

$$-\sum_{i=1}^{N}\sum_{k=1}^{K-1} \lambda_k \left[log(\sigma(g(\mathbf{x}^{[i]}, \mathbf{W}) + b_k)) y_i^k + log(1 - \sigma(g(\mathbf{x}^{[i]}, \mathbf{W}) + b_k))(1 - y_i^k) \right]$$

ahol λ_k a k-adik feladat fontossági paramétere. A méréseimben $\forall k : \lambda_k = 1$ uniform súlyozást használtam. Prediktáláshoz megvizsgáljuk a szigmoid által adott valószínűségeket, és összeszámoljuk, hány helyen nagyobb a prediktált valószínűség mint 0.5, azaz:

$$q = 1 + \sum_{k=1}^{K-1} \mathbb{1}\{P(y_i^k = 1) > 0.5\}.$$

A cikk szerzői bebizonyították a következő tételt, ami garantálja a kimnetek monotonitását:

1.2.1. Tétel. *A fent definiált veszteségfüggvény minimalizálásával az optimális $\boldsymbol{W}, \boldsymbol{b}$ kielégíti $b_1 \geq b_2 \geq ... \geq b_{K-1}$.*

Következésképp $P(y_i^1 = 1) \geq P(y_i^2 = 1) \geq ... \geq P(y_i^{K-1} = 1)$. Így $\mathbb{1}\{P(y_i^k = 1) > 0.5\}$, $k = 1, ..., K-1$ is rang-monoton.

1.3. CORN (Conditional Ordinal Regression for Neural Networks)

A CORAL-t bemutató cikk szerzői egy évvel később megjelent cikkükban javítottak munkájukon, az új megközelítésüket CORN néven jelentették meg [8]. Ebben kiemelik, hogy a CORAL hátrányos abból a szempontból, hogy korlátozza a háló flexibilitását és kifejezőképességét. Így a CORN célja az, hogy súlymegosztás és a modell komplexitásának növelése nélkül garantálja a rang-monotonitást.

Abban megyezik a CORAL és CORN megközelítése, hogy mindekettő ordinal one-hot vektorral elkódolt címkéket használ a tanításhoz, illetve a modell kimenete $K-1$ neuron, ahol K továbbra is

az osztályok számát jelöli. Azonban a CORN framework feltételes valószínűségeket fog prediktálni, amiket azután felhasználunk a feltétel nélküli valószínűségek számolásához. Konkrétan, legyenek a szigmoid függvény által adott valószínűségek a $h_k(\mathbf{x}^{[i]}) = P(y_i > r_k | y_i > r_{k-1})$, $k = 1, .., K-1$, ahol a megfelelő események részhalmazai egymásnak: $\{y_i > r_k\} \subseteq \{y_i > r_{k-1}\}$. Innen a feltétel nélküli valószínűségek a láncszabállyal számolhatók:

$$P(y_i > r_k) = \prod_{j=1}^{k} h_j(\mathbf{x}^{[i]}).$$

Mivel $0 \leq h_j(\mathbf{x}^{[i]}) \leq 1$ ezért $P(y_i > r_1) \geq P(y_i > r_2) \geq ... \geq P(y_i > r_{K-1})$, a rang-konzisztencia teljesül. A predikció számolása ugyanúgy megy, mint a CORAL esetében, a feltétel nélküli valószínűségekből. Maga a predikciónak a a számolása nem követeli meg a rang-monotonitást, de ez a tulajdonság mindenképpen előnyös ehhez a számoláshoz.

A feltételes valószínűségek modellezéséhez a tanítóhalmazt szétosztjuk, és a veszteségfüggvény számolásakor csak az adott részhalmazára koncentrálunk a tanítóadatoknak. A szétbontás a következőképpen történik:

- S_1 : minden $(\mathbf{x}^{[i]}, y^{[i]})$
- $S_2 : \{(\mathbf{x}^{[i]}, y^{[i]}) | y^{[i]} > r_1\}$
- ...
- $S_{K-1} : \{(\mathbf{x}^{[i]}, y^{[i]}) | y^{[i]} > r_{K-2}\}$.

Az S_k halmazt a $P(y_i > r_k | y_i > r_{k-1})$ valószínűség kiszámításához használjuk. A veszteségfüggvény:

$$-\frac{1}{\sum_{j=1}^{K-1} |S_j|} \sum_{j=1}^{K-1} \sum_{i=1}^{|S_j|} \left[log(h_j(\mathbf{x}^{[i]})) \mathbb{1}\{y_i > r_j\} + log(1 - h_j(\mathbf{x}^{[i]})) \mathbb{1}\{y_i \leq r_j\} \right]$$

ahol $\mathbf{x}^{[i]}$ a megfelelő S_j halmaz i-edik adatpontja.

1.4. Unimodális eloszlások

A rank-monotonitás mellett másik megközelítési mód az unimodális eloszlásra való kényszerítés. Tehát egy adatpontra olyan kimeneti eloszlást szeretnénk az osztályokon, aminek egy módusza van, ez a valódi osztály valószínűségének feleljen meg, és a módusztól elindulva mindkét irányban monoton csökkenjenek a többi osztály valószínűségei. Az unimodális kimenet orvosi alkamazás szempontjából is előnyös, hiszen mutatja, mennyire magabiztos a modell a prediktált osztályban, illetve annak a szomszédjaiban.

Az unimodális kimenetek többféleképpen is előállíthatóak, így veszteségfüggvény-módosítással, illetve az architektúra átalakításával is. Utóbbi csoportba tartozik a Beckham [2] által adott módszer. Lényege, hogy a konvolúciós modellnek egy kimeneti neuronja van, ami egy diszkrét unimodális eloszlás paraméterét hivatott megjósolni. A cikkben kétfajta eloszlást vizsgáltak, a binomiális és a Poisson eloszlást. Előbbiben a p-t becsüljük a hálóval, utóbbiban λ-t. A kapott paramétert felhasználva

az adott diszkrét eloszlás alapján számolhatóak a kimeneti valószínűségek. Hátrányt jelent azonban, hogy erős megszorítás lehet megkövetelni a kimeneti valószínűségektől, hogy egy specifikus eloszlást kövessenek, illetve a Poisson eloszlás esetében nehéz a szórásnégyzetét kontrollálni. Utóbbi problémára a cikk szerzői bevezettek egy hiperparamétert, azonban az eloszlás szórásnégyzetét megfelelően befolyásoló paraméter értékének megtalálása összetettebb hiperparaméter keresést eredményez [5].

1.4.1. A veszteségfüggvény módosítása

Az ordinal classification feladatot több forrás is a veszteségfüggvény átalakításával közelíti meg. Az átalakítás célja, hogy az egy adatpontra adott kimeneti valószínűségek unimodális eloszlást kövessenek. A modell ebben a megközelítésben osztályszámnyi kimeneti neuronnal rendelkezik, így maga az architektúra úgy épül fel, mint egy szokásos többosztályú klasszifikációs feladatnál. Azonban ebben az esetben hátrányos a megszokott cross-entropy veszteségfüggvényt használni a tanításhoz, mert ezáltal elveszik az osztályok közötti rendezettségi információ. Ugyanis, a cross-entropy-nál a címkék one-hot vektoros alakját használjuk: $L(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_{k=1}^{K} y_{i,k} log(\hat{y}_{i,k})$ ahol $y_{i,k}$ jelöli $\mathbf{x}^{[i]}$ one-hot címkéjének k-adik koordinátáját, és $\hat{y}_{i,k}$ az $\mathbf{x}^{[i]}$-re adott predikció k-adik koordinátáját. Ekkor, ha k^* jelöli az adatpont valódi osztályának indexét, a cross-entropy: $L(\mathbf{y}, \hat{\mathbf{y}}) = -log(\hat{y}_{i,k^*})$. Ebből az látszik, hogy csak az igazi osztályra adott predikciót használja fel a függvény, és nem veszi figyelembe a többi osztályra kapott valószínűségeket. Emiatt az sem jelenik meg benne, hogy az egyes félreklasszifikálások különböző mértékű hibát generálnak. Továbbá nem ösztönzi a modellt, hogy a kimeneti valószínűségek unimodális eloszlást kövessenek, tehát előfordulhatnak inkonzisztenciák a modell kimeneteiben.

A [1] cikkben egy olyan regularizációs tagot vezetnek be a cross-entropy veszteségfüggvénybe, ami bünteti az unimodális eloszlástól való eltérést. Legyen $\lambda \geq 0$ és $\delta > 0$. A veszteségfüggvény:

$$CO2(\mathbf{y}, \hat{\mathbf{y}}) = L(\mathbf{y}, \hat{\mathbf{y}}) + \lambda \sum_{k=1}^{K-1} \mathbb{1}\{k \geq k_i^*\} RELU(\delta + \hat{y}_{i,k+1} - \hat{y}_{i,k}) + \lambda \sum_{k=1}^{K-1} \mathbb{1}\{k \leq k_i^*\} RELU(\delta + \hat{y}_{i,k} - \hat{y}_{i,k+1})$$

ahol a λ együttható kontrollálja a plusz tagok szerepét. Ezek a tagok arra ösztönzik a prediktált valószínűségeket, hogy a valódi osztálycímkétől távolodva monoton csökkenőek legyenek. A büntetések arányosak az egymást követő valószínűségek különbségével. A δ bevezetése azért szükséges, hogy a kapott eloszlások ne legyenek túl laposak, tehát az egyes szomszédos osztályok valószínűségei közötti különbség legalább δ legyen. A méréseimben $\delta = 0.05$ értéket használtam.

A [7] cikk szerzői a cross-entropy függvényt alakították át, hogy az eltévesztett predikciók valószínűségei jelenjenek meg a veszteségfüggvényben, és ezáltal a rossz predikciók büntetésén legyen a hangsúly, ne pedig a jó predikcióban való magabiztosságot erősítsük, ahogy a sima cross-entropy teszi. Jelöljük $\hat{y}$-vel a modell predikcióit,és jelölje c az adatpont valódi címkéjét. Ekkor a veszteségfüggvény (Class-distance weighted loss) alakja:

$$CDW - CE = -\sum_{i=0}^{K-1} log(1 - \hat{y}_i) \cdot |i - c|^{\alpha}.$$

Mivel ebben az esetben one-hot elkódolással használjuk az címkéket, így csak a valódi osztálynak megfelelő tagok maradnak meg a fenti kifejezésben miden egyes adatpontra. A prediktált valószínűségekből a maximumot véve választjuk ki a predikciót, így ha a megfelelő valószínűséggel már eltaláltuk a jó osztályt, akkor $|i - c|^{\alpha}$ kinullázza az adatpont veszteségét, ha pedig nem találtuk el, akkor arra

ösztökéli a modellt, hogy csökkentse annak az osztálynak a valószínűségét az adott adatpontra. A $|i - c|^{\alpha}$ szorzó miatt minél távolabbi osztályt mondtuk a valóditól, az annál nagyobb büntetést jelent a veszteségfüggvényben. Ez a büntetés a címkepárok távolságával arányos. Ezzel azt is ösztönzi, hogy a tévesztések is inkább a valódi osztály szomszédainál történjenek meg. Az α hiperparaméterrel a büntetések mértékét változtathatjuk. A méréseimben én az $\alpha = 2$ értéket találtam a legjobbnak. A fenti veszteségfüggvény előnye, hogy semmi módosítást nem igényel sem a modell architektúráját, sem a címkézést illetően.

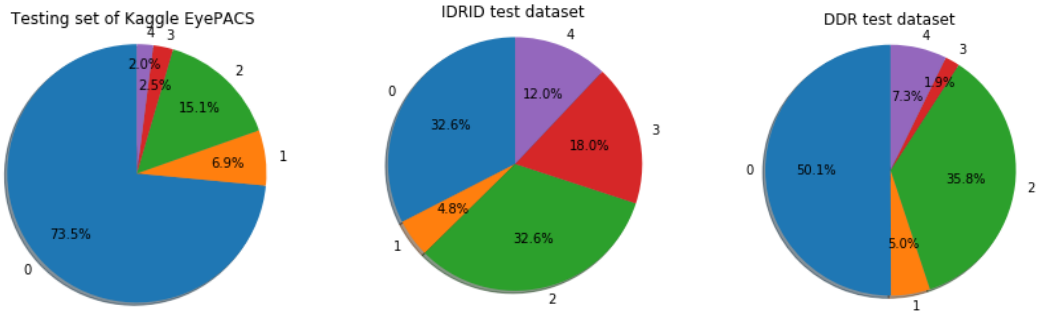
2. fejezet

Mérések

2.1. Az adathalmaz és a modell

A modellek tanításakor a Kaggle EyePACS diabetikus retinopátia detektálására és osztályozására való adathalmazát használtam. Annak megállapításához, mennyire általánosítanak jól a kipróbált technikák, kiértékeléskor több, más forrásból származó teszthalmazt is használtam. Az első a Kaggle EyePACS-ből leválasztott adathalmaz volt, ugyanolyan címke eloszlással, mint ami a tanítóhalmazban van jelen. A második és harmadik pedig az IDRID, illetve a DDR adathalmazok voltak. Ezek szintén publikusan elérhetőek, kisebb méretűek mint az EyePACS, viszont gondosabban összerakottak, jobb minőségűek. A megfelelő teszthalmazok eloszlása a 2.1 ábrán látható.

Minden modell alapja egy EfficientNet-B3 architektúra volt, ennek felépítésén és tanítási mechanizmusán módosítottam a méréseimben. Korábbi, retinafelvételes adathalmazon végzett kísérleteim alapján ez a konvolúciós háló megfelelő eredményeket ért el, ezért választottam ezt mostani vizsgálódásaimhoz is.



(a) (b) (c)

2.1. ábra. A teszthalmazok címkéinek eloszlása

2.1.1. Az adat előfeldolgozása

A Kaggle EyePacs az egyik legnagyobb publikusan elérhető diabetikus retinopátia adathalmaz, azonban hátránya, hogy sok rossz minőségű, különböző kamerákból származó, zajos felvételt, illetve

félrecímkézett képet tartalmaz. A félreannotálás orvosi hibából ered, így azt feltételezzük, valószínűleg inkább a szomszédos osztályhatárokon történhettek a félrecímkézések, a súlyosság növekedésének fokozatossága miatt.

A képeken előfeldolgozást végeztem, hogy növeljem a bemeneti képek minőségét. A felvételek különböző megvilágításokkal rendelkeztek, az adathalmazban találtam elég sötét képeket is, így próbáltam megszüntetni ezen körülmények negatív hatásait. A zaj csökkentését Gauss elmosással végeztem, ehhez segítségül felhasználtam a [4] kódot. A kontraszt javításához CLAHE módszert alkalmaztam a felvételekre. Úgy találtam, hogy a legjobb, ha csak a zöld színcsatornára alkalmazzuk a módszert, mert így rajzolódnak ki legjobban az erek és sérülések a retinán, amik a legfontosabb részek a betegség detektálásához. Végezetül a képek jobb és bal széléből vágtam a hátteret tartalmazó, nem informatív részekből, hogy majd a végső átméretezéskor a képek minél nagyobb felülete legyen a betegség felismeréséhez hasznos.

2.2. Metrikák

A modellek teljesítményének kiértékeléséhez olyan metrikákat kell választani, amik figyelembe veszik a tévesztések különböző típusait. A méréseimben az egyik ilyen mérőszám a kvadratikus Cohen-kappa, a másik pedig a négyzetes eltérés (MSE) volt. Ezenkívül tévesztés mátrixot használtam a félreklasszifikálások szemléltetéséhez.

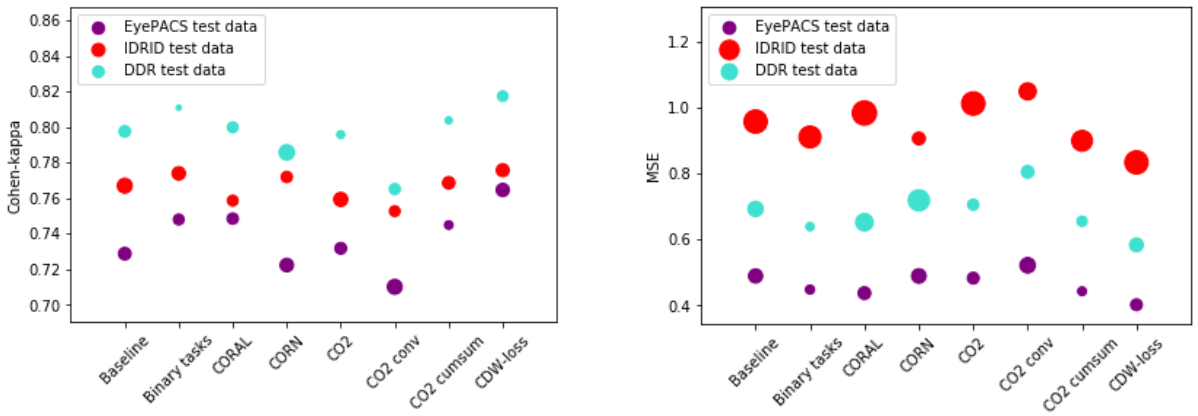
A Cohen-kappa azt méri, mennyire ért egyet két kiértékelő, jelen esetben mennyire egyezik a modell által, illetve az orvosok által megállapított címkeeloszlás. Ahol nem egyezik meg a két vélemény, ott a valódi címke és a predikció közti különbség négyzetével súlyozza a téves klasszifikálásokat. A Cohen-kappa egy -1 és 1 közötti számot ad, ahol 1 a teljes egyetértést jelenti, 0 a véletlen megegyezést, -1 pedig a teljes egyet nem értést. A számításban felhasználja az O tévesztés mátrixot, a W súlymátrixot és egy E mátrixot, amiben a véletlen megegyezés valószínűségei szerepelnek. Legyen K az osztályok száma, ekkor a négyzetes Cohen-kappa:

$$\kappa = 1 - \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} w_{i,j}o_{i,j}}{\sum_{i=1}^{N}\sum_{j=1}^{N} w_{i,j}e_{i,j}} \qquad\qquad w_{i,j} = \frac{(i-j)^2}{(K-1)^2}.$$

2.3. Mérési eredmények

Az első fejezetben kifejtett technikákkal tanított modellek eredményei a 2.2 ábrán láthatóak. Mindegyik mérést 4 különböző random maggal futtattam, és az eredményeket átlagoltam, az így kapott értékek láthatóak az ábrákon. A körök középpontjai jelentik az átlagos metrika értéket, a körök sugarai a szórást. Ezenkívül különöző színekkel tüntettem fel az egyes teszthalmazok eredményeit. Elsőként megnéztem, hogyan teljesít hiperparaméter optimalizálás után az alap EfficientNet-B3, 5 kimeneti neuronnal és cross-entropy veszteségfüggvénnyel, ez lett a baseline modellem. A 2.3 ábrán látható a háló által adott tévesztés mátrix, illetve véletlenül választottam az EyePACS teszthalmazából adatpontokat, és kirajzoltattam a modell által prediktált valószínűségeiket. Utóbbin látszódik, hogy előfordulnak inkonzisztens kimenetek is. Ezek után tanítottam a Niu [6] által felépített bináris részfeladatokra bontott

modellt, majd a CORAL, CORN hálót, valamint a CO2 és CDW módosított veszteségfüggvényekkel is végeztem méréseket. Az alap teljesítményt, illetve a legjobb Cohen-kappa értékeket a 2.1 táblázatban külön is feltüntettem.



(a) Cohen-kappa értékek

(b) MSE értékek

2.2. ábra. Eredmények

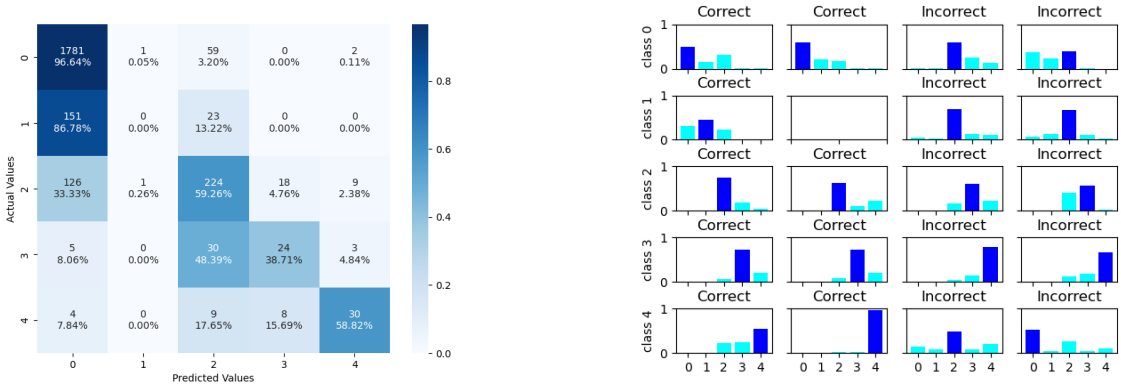
A CORAL bár nem jelentősen, de jobban teljesített a baseline-nál. A 2.4 ábrán látható a modell által adott tévesztés mátrix, illetve néhány adatpont valószínűségei. Látszik, hogy a rang-monotonitást elértük a modellel, és ez pozitív eredmény. A CORN nem teljesített jobban, mint az alap háló, pedig ezt vártuk volna. A legmagasabb Cohen-kappát a CDW veszteségfüggvénnyel tanított háló érte el, mindhárom teszthalmazon. A tévesztés mátrixa és kimeneti valószínűségek a 2.5 ábrán láthatóak. A kimenetek eloszlásai mutatják azt, amit elvártunk, hogy amennyire lehet, unimodális legyenek. Ezenkívül jó eredményei lettek a bináris részfeladatokra bontott modellnek is. Ez utóbbi két tanítási módszerre megvizsgáltam, mi történik ha two-phase learning technikával kombinálom őket. Az önálló projektem előző félévében vizsgáltam ezt a technikát az adathalmaz kiegyensúlyozatlansága miatt. Azonban nem tudta javítani se a részfeladatokra bontott, se a CDW veszteségfüggvénnyel tanított modell eredményét. Mindhárom teszthalmaz esetében vagy a sima részfeladatokra bontott és CDW-vel tanított modell szórásán belüli értékeket kaptam, vagy náluk valamivel rosszabb eredményeket.

	EyePACS teszthalmaz	IDRID teszthalmaz	DDR teszthalmaz
Alap teljesítmény	0.7288	0.7670	0.7975
Bináris részfeladatok	0.7480	0.7739	0.8108
CDW loss	0.7646	0.7757	0.8173

2.1. táblázat. Cohen-kappa értékek a három teszthalmazon

A CO2 függvényben a λ hiperparamétert kellett állítani. Bár a cikkben ez az eggyüttható a regularizációs tagok előtt szerepel, megnéztem mi történik, ha az $L(\mathbf{y}, \hat{\mathbf{y}})$ cross-entropy szorzójaként használom a λ-t, illetve ha a regularizációs tagok és a cross-entropy különböző konvex kombinációját veszem. Azt tapasztaltam, amit nagyjából el is vártunk a kimeneti valószínűségektől, hogy minél nagyobb súlyt kapott a regularizációs tag, annál unimodálisabbak lettek az adatpontok kimeneti valószínűségei, ugyanakkor lassabban is tanult a modell, mint mikor kisebb együtthatót kaptak. A legjobb beállításnak $\lambda = 0.5$ értéket találtam, a regularizációs tagok együtthatójaként. A konvex kombinációs kísérleteim célja (az ábrán *CO2 conv*-val jelölve) az volt, hátha lehet finomabb beállítást találni a cross-entropy és a regularizációs tag kombinálására, de a metrikákat tekintve nem jártam sikerrel, a legjobb beállítás

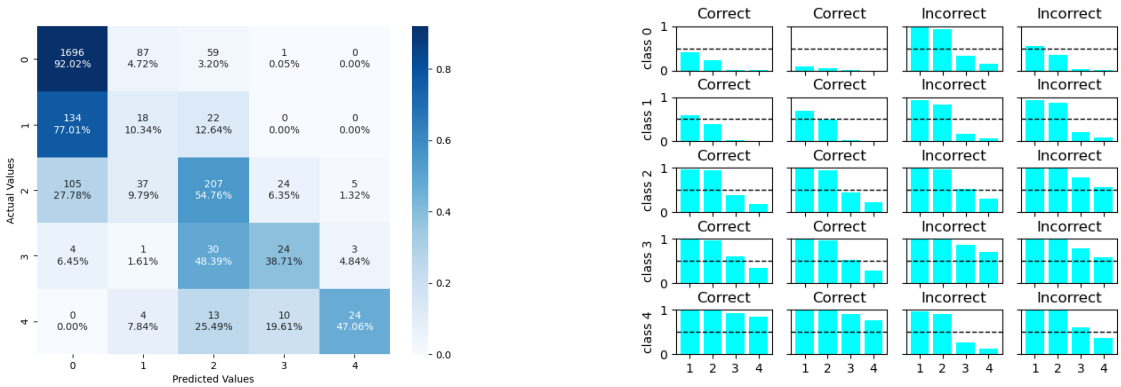
is rosszabb eredményt ért el mindegyik teszthalmazon, mint a sima CO2 λ-jára megtalált legjobb beállítás. Ezenkívül változtattam a predikció számolásának módját is. Az alap beállítás szerint a modell egy adatpontra visszaad egy vektort, ami megad minden osztályra egy valószínűséget, majd ezekből választjuk ki a legnagyobb értékűt. Ennek az indexe lesz a prediktált címke. Számolhatjuk azonban úgy is az osztály indexét, hogy a kapott valószínűségekkel súlyozzuk az osztály címkéit (expectation trick [2]): $\sum_{i=0}^{K-1} p_i c_i$. Ezzel az összegzéssel az eredményeim jelentősen rosszabbak lettek, azonban a tévesztés mátrixokon megfigyelhető volt, hogy amíg az eddigi modellek nagyon nehezen különböztették meg egymástól a 0-s és 1-es osztályt, addig ez a modell viszonylag jól osztályozta az 1- es címkéjű adatpontokat. Ez ötletet ad arra, hogy egy ensemble modellt építve a 0-1-2 osztályokat szétválogató modell tanításához és predikciójának számolásához érdemes lehet ezt a módszert használni. Egy harmadik aggregálási módszer, hogy kumulatívan összegezzük a valószínűségeket $(p_i, i = 0, .., K-1)$, és megnézzük, az így kapott számok közül hány kisebb 0.5-nél: $\sum_{i=0}^{K-1} \mathbb{1}\{q_i < 0.5\}$ ahol $q_i = \sum_{j=0}^{i} p_j$. A kapott eredmények az ábrán *CO2 cumsum* néven láthatóak.



(a) Tévesztés mátrix

(b) Kimeneti valószínűségek véletlenül választott adatpontokon

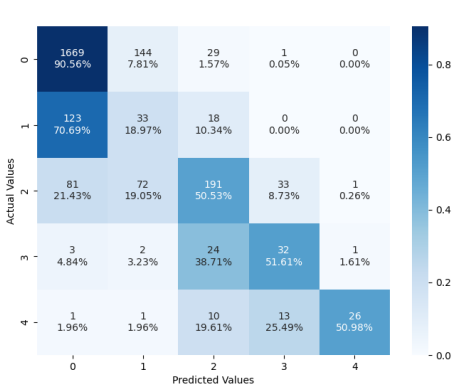
2.3. ábra. A baseline modellel végzett mérések eredményei



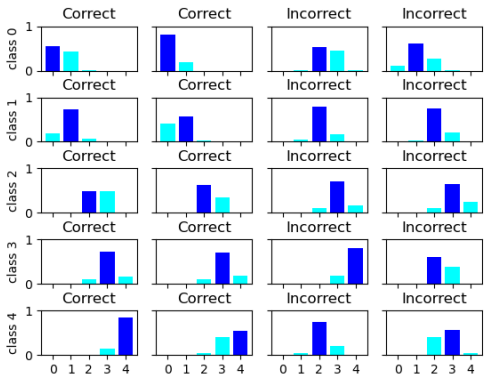
(a) Tévesztés mátrix

(b) Kimeneti valószínűségek véletlenül választott adatpontokon

2.4. ábra. A CORAL modellel végzett mérések eredményei

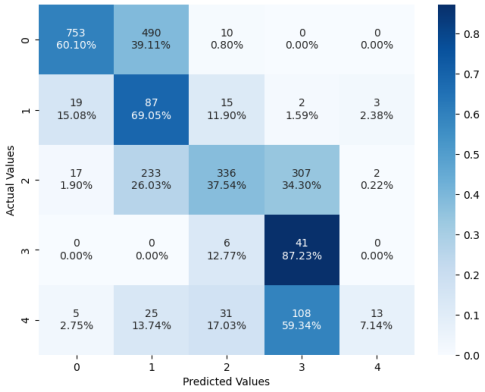


(a) Tévesztés mátrix



(b) Kimeneti valószínűségek véletlenül választott adatpontokon

2.5. ábra. A CDW veszteségfüggvénnyel végzett mérések eredményei



2.6. ábra. A CO2 függvénnyel és expectation trick-el kapott tévesztés mátrix

Összességében az mondható, hogy ígéretesnek bizonyultak a fent bemutatott módszerek, mind az elért eredmények, mind a modell kimeneteinek magyarázhatósága szempontjából. Kiderült az is, hogy a különböző osztályok megtanulásához nem feltétlenül ugyanazt a módszert kell alkalmazni, így természetesen adódik az ötlet, hogy egy megfelelően összerakott ensemble modellel lehetne tovább javítani az eredményeinket. Ehhez nem is feltétlenül csak EfficientNet alapú modellt lehetne használni, hiszen előfordulhat, hogy más architektúra családok más információkat tudnak kinyerni az adatból. Érdemes lenne további veszteségfüggvény módosításokat is megvizsgálni, hiszen a fenti mérések alapján ezek önmagukban is képesek lehetnek teljesítményjavulást elérni, az architektúra módosítása nélkül is.

Irodalom

[1] Tomé Albuquerque, Ricardo Cruz és Jaime S Cardoso. “Ordinal losses for classification of cervical cancer risk”. *PeerJ Computer Science* 7 (2021), e457.

[2] Christopher Beckham és Christopher Pal. “Unimodal probability distributions for deep ordinal classification”. *International Conference on Machine Learning*. PMLR. 2017, 411–419. old.

[3] Wenzhi Cao, Vahid Mirjalili és Sebastian Raschka. “Rank consistent ordinal regression for neural networks with application to age estimation”. *Pattern Recognition Letters* 140 (2020), 325–331. old.

[4] Ben Graham. “Kaggle diabetic retinopathy detection competition report”. *University of Warwick* (2015), 24–26. old.

[5] Xiaofeng Liu és tsai. “Unimodal regularized neuron stick-breaking for ordinal classification”. *Neurocomputing* 388 (2020), 34–44. old.

[6] Zhenxing Niu és tsai. “Ordinal regression with multiple output cnn for age estimation”. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, 4920–4928. old.

[7] Gorkem Polat és tsai. “Class Distance Weighted Cross-Entropy Loss for Ulcerative Colitis Severity Estimation”. *arXiv preprint arXiv:2202.05167* (2022).

[8] Xintong Shi, Wenzhi Cao és Sebastian Raschka. “Deep Neural Networks for Rank-Consistent Ordinal Regression Based On Conditional Probabilities”. *arXiv preprint arXiv:2111.08851* (2021).