Képosztályozó modellt támadó irányított zaj elleni védekezés Gauss képpiramissal és adversarial tanítással

Szűcs Gábor[1], Kiss Richárd[2]

[1] Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék,
Budapest
[2] Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Balatonfüredi Hallgatói Kutatócsoport
{szucs@tmit.bme.hu, richard.kiss@edu.bme.hu}

Absztrakt. Napjainkban a képfelismerési feladatokra nagyon pontos megoldást tudnak nyújtani a mély neurális hálózatok mindaddig, amíg a bemeneti minták a tanítóhalmaz eloszlásából származnak. Azonban, ha a betanított modellnek olyan bemeneteket adnak, amik hasonlítanak ugyan egy-egy eredeti eloszlásból származó képre, de szándékosan egy irányított zajt adnak hozzá, akkor azokat a modell tévesen rossz osztályba sorolhatja. Ez az érzékenység nem kívánatos viselkedés, hiszen egy támadó fél ezt kihasználva kárt tud okozni az ilyen megoldásokat alkalmazó rendszerekben. A cikkünkben egy új védekezési módszert javaslunk a képosztályozó modelleket támadó irányított zaj alapú támadások ellen. A módszerünk Gauss képpiramisokon, az azt használó mély neurális hálókon és adversarial tanításon alapul. Két adathalmazon mértük a módszerünk jóságát és összehasonlítottuk más védekezési módszerekkel. A futási eredmények azt mutatták, hogy a Gauss piramisok és adversarial tanítás kombinációja (GP+AT) érte el a legjobb eredményt a pontosság szempontjából. Továbbá megvizsgáltuk a modelleket transzferabilitás szempontjából is, ami a black-box típusú támadások elleni ellenállóképességre utal, és a javasolt GP+AT bizonyult a legrobusztusabb módszernek.

Kulcsszavak: Gauss képpiramis, adversarial támadás, mély neurális hálózat, képosztályozás

1 Bevezetés

Neurális hálózatok tanítását általában gradiens módszerrel végezzük, ehhez pedig meg kell határozni egy hibafüggvényt, ami adott bemenet mellett minősíti a hálózatot. A hálózat súlyait a hibafüggvény minimalizálásával, (a hibafüggvény hálózat paraméterei szerint vett deriváltjának megfelelő irányba való lépésekkel) iteratívan frissítjük. A minősítésre azonban alkalmasabb mérték, ha a veszteség összes pontra vonatkozó átlagos értékét, a veszteség várható értékét, határozzuk meg [1]; ennek a kiszámításához viszont szükségünk lenne a minták (bemenet, kimenet párok) együttes sűrűségfüggvényére, ezt általában nem ismerjük, tanításhoz egy véges halmaz áll csak rendelkezésünkre. A tapasztalati kockázat minimalizálásának (ERM - Empirical Risk

Minimization) elve azt mondja ki, hogy a tanítási eljárás eredményének egy olyan megoldást fogadunk el, ami a tapasztalati kockázatot minimalizálja (ez ugyanis elég nagy számú minta mellett tart a valós kockázathoz [14].

Az ERM segítségével tehát tudunk olyan modelleket tanítani, amikkel az adott problémát jól meg tudjuk oldani, viszont ezek nagyon gyakran nem lesznek robusztusak adversarial támadásokkal szemben [10]. Adversarial támadáskor a támadó olyan (minimális) perturbációt keres, amit a bemenetre keverve a modell kimenetét tetszőlegesen befolyásolhatja. A támadást a gépi tanuló modell ismerete alapján két csoportba soroljuk:

1. White-box támadásról [11] akkor beszélünk, ha rendelkezésünkre áll az osztályozó modell architektúrája és összes paramétere. Ilyen esetben általában gradiens alapú támadást használnak, ugyanis azt hatékonyan ki tudják számítani.
2. Black-box támadás [12] esetén csak a címkék valószínűsége, vagy akár csak a jósolt osztály áll rendelkezésre. Vannak olyan támadók (algoritmusok), amik ennek ellenére viszonylag hatékonyan (pár ezres mintavételezéssel) tudnak adversarial mintákat készíteni. Egy másik módszernél egy helyettesítő modellt tanítanak be, a tapasztalat azt mutatja, hogy az azt megtévesztő minták gyakran megtévesztik az eredeti modellt is; ezt a tulajdonságot transzferabilitásnak nevezzük, és a védekezési módszerek arra irányulnak, hogy ezt csökkentsék.

A támadó célja alapján szintén két csoportra oszthatók a támadások, melyek a következők:

1. Célzott támadás esetén a támadó olyan minimális zajt keres valamilyen távolság-metrika szerint (d), amivel egy kiválasztott osztályba ($\tilde{c}$-be) juttathatja el a mintát (azaz a modell ezt fogja predikcióként adni):
$$\min_{\tilde{\boldsymbol{x}}} \mathrm{d}(\tilde{\boldsymbol{x}}, \boldsymbol{x}) \text{ és } C(\tilde{\boldsymbol{x}}) = \tilde{c}.$$
2. Nem célzott támadás esetén a támadó célja az, hogy bármely olyan osztályba juttathatja el a mintát, ami nem egyezik az eredeti osztállyal:
$$\min_{\tilde{\boldsymbol{x}}} \mathrm{d}(\tilde{\boldsymbol{x}}, \boldsymbol{x}) \text{ és } C(\tilde{\boldsymbol{x}}) \neq C(\boldsymbol{x}).$$

Általában a perturbáció keresésekor megelégszünk egy szuboptimális megoldással, amivel a kívánt osztályba juttatható el a minta. Bár ennek a problémakörnek a szakirodalma még nagyon friss, de már néhány ilyen támadó jellegű zaj keresésére (illetve kivédésére) alkalmas algoritmust kidolgoztak, ezek közül mutatjuk be a legfontosabbakat a következő részben.

2 Támadó és védekezési módszerek az adversarial támadásoknál

2.1 Fast Gradient Sign Method

Az Fast Gradient Sign Method (FGSM) [2] módszer használható white- és black-box támadási modellek esetén is (utóbbinál szükséges egy helyettesítő modell

betanítása, a megtévesztendő modell transzferablilitásától függően lesz hatékony az FGSM ilyen esetben). Az osztályozó modell – melynek paramétereit θ jelöli – feladata a bemenet (x) a megfelelő osztályba (y) sorolása N osztály közül. Első lépésben egy betanított modellt (ami lehet a megtévesztendő, ú.n. célmodell vagy lehet egy helyettesítő modell) felhasználva kiszámoljuk az N osztály feletti eloszlást (azaz a modell szerint melyik osztályba, milyen valószínűséggel tartozik a bemenet), majd definiálunk egy hibafüggvényt (J), ami a helyes osztálytól való eltérést bünteti (általában keresztentrópiát használunk), végül kiszámítjuk ennek gradiensét a bemenet (x) szerint. Ebből kapjuk meg azt az irányt, ami felé módosítva a bemenetet a legnagyobb növekedést érhetünk el a hibafüggvényen (azaz a bemenetet így lehet legkisebb változtatással hibás osztály felé juttatni). Az FGSM a bemenet egy ℓ_∞ határolt környezetében keresi a támadó perturbációt úgy, hogy a bemenet szerint számított gradiensre alkalmazott előjelfüggvénnyel kapott irányba ε-nyit lép:

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} + \varepsilon \cdot \mathrm{sign}\big(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y, \boldsymbol{\theta})\big).$$

2.2 Iterative FGSM

Az Iterative Fast Gradient Sign Method (Iterative FGSM, vagy röviden I-FGSM) módszer az FGSM finomított változata [6]. Az FGSM módszer iterált alkalmazásával keres egy optimális zajt a bemenet ℓ_∞ környezetében. Ez egy gradiens eljárást futtat és az FGSM-hez hasonlóan számítja a gradienst; az iterációs lépések a következő két egyenleten alapulnak (ahol clip egy olyan levágási módot jelöl, melyet a 3. egyenlet mutat):

$$\tilde{\boldsymbol{x}}_{k+1} = \mathrm{clip}_{x,\varepsilon}\left(\tilde{\boldsymbol{x}}_k + \varepsilon \cdot \mathrm{sign}\left(\nabla_{\tilde{\boldsymbol{x}}_k} J(\tilde{\boldsymbol{x}}_k, y, \boldsymbol{\theta})\right)\right)$$

$$\tilde{\boldsymbol{x}}_0 = \boldsymbol{x}$$

$$\mathrm{clip}_{x,\varepsilon}(A_{i,j}) = \begin{cases} X_{i,j} + \varepsilon \text{, ha } A_{i,j} > X_{i,j} + \varepsilon \\ X_{i,j} - \varepsilon \text{, ha } A_{i,j} < X_{i,j} - \varepsilon \\ A_{i,j} \qquad\qquad \text{egyébként.} \end{cases}$$

2.3 Projected Gradient Descent

A Projected Gradient Descent (PGD) [9] módszer ugyanúgy működik, mint az iteratív FGSM azzal a különbséggel, hogy $\tilde{x}_0$-t nem az eredeti bemenetre (x-re) inicializálja, hanem véletlenszerűen választja annak ℓ_∞ környezetéből:

$$\tilde{\boldsymbol{x}}_0 \sim \boldsymbol{x} + \varepsilon \boldsymbol{a},$$

ahol az $\boldsymbol{a}$ vektort véletlenszerűen választjuk az egységgömbből (ℓ_2 környezet esetén) vagy az egység hiperkockából (ℓ_∞ környezet esetén). ε ebben az esetben ugyanazzal a jelentéssel bír, mint az FGSM-nél (azaz a támadás amplitúdója).

2.4 Adversarial tanítás a védekezéshez

A fentebb említett támadó módszerek után bemutatjuk azokat a lehetőségeket, amikkel előzetesen védekezhetünk. Mivel a támadó által előállított adversarial mintákkal a modellt nem tanítottuk, így a tanult osztályozó függvényre ezeken a helyeken direkten nincsenek korrekciók. Ennek következtében lehet az, hogy ilyen mintákra a modell válasza teljesen eltérő a legközelebbi valós mintákhoz képest. Ezt a jelenséget publikálták egy képosztályozás problémán demonstrálva [2]; és a cikkben egy adversarial tanítási módszert javasoltak a robusztus modell tanítására a következőképpen: a tanító mintahalmazt virtuálisan augmentálják (kibővítik) olyan mintákkal, amik tartalmaznak támadó perturbációt. Majd a kibővített tanulóhalmazon történik a tanítás, melyet a hibafüggvény kiegészítésével érnek el, ahogy az az alábbi képletekben látszik (*J'* a módosított hibafüggvény, *J* az eredeti hibafüggvény):

$$J'(\boldsymbol{x}, y, \boldsymbol{\theta}) = \alpha J(\boldsymbol{x}, y, \boldsymbol{\theta}) + (1 - \alpha) J(\tilde{\boldsymbol{x}}, y, \boldsymbol{\theta})$$
$$\tilde{\boldsymbol{x}} = \boldsymbol{x} + \varepsilon \cdot \operatorname{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y, \boldsymbol{\theta})\right) \quad \text{(FGSM)}$$
$$\alpha \in [0; 1].$$

2.5 Defenzív Desztilláció

A bemutatott támadó módszerek a hibafüggvény gradiensének kiszámításával keresnek (szub-)optimális perturbációt. Egy másik védekezési módszer ezt a számítást nehezíti meg az úgy nevezett desztillációs eljáráson alapulva – a desztillációt eredetileg hálózatok tudásának tömörítésére alkalmazták [3]. A Defenzív Desztilláció [13] védekezési módszer is hasonló elven működik, annyi eltéréssel, hogy a modell architektúráján nem változtatnak (azaz nem tömörítenek), amikor a soft-címkékkel (ez a 0 és 1-eseket tartalmazó diszkrét címkevektor helyett 0 és 1 közötti értékeket tartalmaz) tanítják. A Defenzív Desztilláció lépései a következők:

1. *M* modell inicializálása egy adott *A* architektúra szerint.
2. *M* modell tanítása a tanítóminták és diszkrét osztálycímkéken vett keresztentrópia alapján.
3. Soft-címkék képzése *M* modellel minden tanítómintához.
4. *M'* modell inicializálása *A* architektúra szerint.
5. *M'* modell tanítása a tanítóminták és soft-címkék szerint számított keresztentrópia szerint.

Az *M* és *M'* modell utolsó rétege után egy *softmax* aktiváció található:

$$\operatorname{softmax}(X)_i = \frac{e^{\frac{x_i}{T}}}{\sum_{k=0}^{N-1} e^{\frac{x_k}{T}}},$$

ahol *M* tanításakor a *T* paramétert 1-nek választjuk meg, míg *M'* tanításakor *T* értékét minél nagyobbra célszerű megválasztani, ugyanis belátható [13], hogy a *softmax* függvény *T* paraméterének növelésével a modell Jacobi mátrixának abszolút értékét csökkenthetjük, ami azt jelenti, hogy a modell kimenete kevésbé lesz érzékeny a bemeneten való kis változásokra, így a támadó perturbációra is. Következtetéskor *M'* modell *T* paraméterét visszaállítjuk 1-re (habár a címkék sorrendezését ez nem változtatja).

3 Gauss képpiramis és adversarial tanítás kombinációja a védekezéshez

A Gauss képpiramist régóta használják a képfeldolgozásban [8]; egy képből állít elő több (kisebb) méretű képet egy Gauss szűrő és kicsinyítés ismételt alkalmazásával. Cikkünkben egy olyan ötlet megvalósítását mutatjuk be, amely ennek a Gauss képpiramisnak a védekezésbe való beemelését célozta meg. Ez ugyanis olyan adversarial támadások ellen lehet alkalmas, amelyek képek osztályozását végző modelleket támadnak, így kizárólag csak a képosztályozás területével foglalkoztunk. A javasolt módszer azon az intuíción alapul, hogy mivel a Gauss simítás csökkenti a zaj mértékét, így várhatóan ezzel csökkenthető az FGSM zaj amplitúdója is, illetve a kicsinyítés következményeképpen kevésbé lesz érzékeny.

Mivel a Gauss szűrő (Gauss simítás) egy Gauss kernellel (K) történő konvolúciót jelent, így a simított kép (x') i, j pixele a következőképpen kapható meg az eredeti képből, ahol w és h a K kernel szélessége és magassága:

$$\mathrm{blur}(\boldsymbol{x})_{i,j} = x'_{i,j} = \sum_{k=0}^{w}\sum_{l=0}^{h} x_{i-\left\lfloor\frac{w}{2}\right\rfloor+k,\,j-\left\lfloor\frac{h}{2}\right\rfloor+l} K_{k,l}.$$

Ha ezt az FGSM támadásra írjuk fel, ahol $\boldsymbol{\delta} = \mathrm{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y, \boldsymbol{\theta})\right)$, akkor a következőket kapjuk:

$$\begin{aligned}\mathrm{blur}(\boldsymbol{x}+\varepsilon\boldsymbol{\delta})_{i,j} &= \sum_{k=0}^{w}\sum_{l=0}^{h}\left(x_{i-\left\lfloor\frac{w}{2}\right\rfloor+k,\,j-\left\lfloor\frac{h}{2}\right\rfloor+l} + \varepsilon\delta_{i-\left\lfloor\frac{w}{2}\right\rfloor+k,\,j-\left\lfloor\frac{h}{2}\right\rfloor+l}\right)K_{k,l}\\ &= \sum_{k=0}^{w}\sum_{l=0}^{h} x_{i-\left\lfloor\frac{w}{2}\right\rfloor+k,\,j-\left\lfloor\frac{h}{2}\right\rfloor+l} K_{k,l} + \varepsilon\sum_{k=0}^{w}\sum_{l=0}^{h}\delta_{i-\left\lfloor\frac{w}{2}\right\rfloor+k,\,j-\left\lfloor\frac{h}{2}\right\rfloor+l} K_{k,l}\\ &= \mathrm{blur}(\boldsymbol{x})_{i,j} + \varepsilon\cdot\mathrm{blur}(\boldsymbol{\delta})_{i,j},\end{aligned}$$

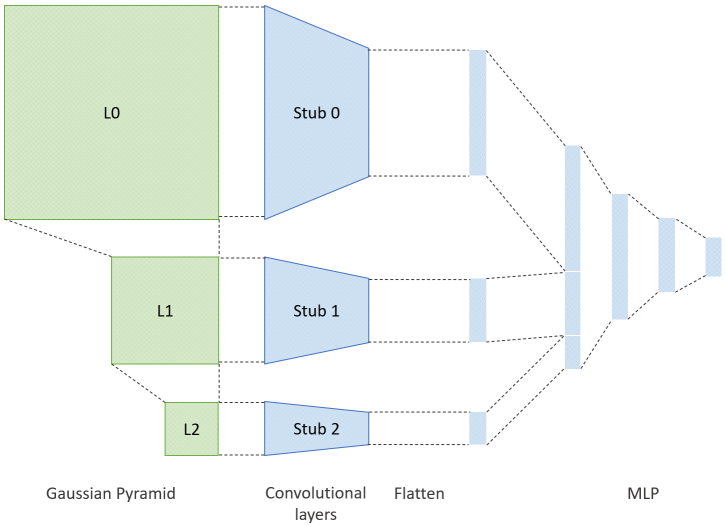
ahol a második tag a támadó zaj nagyságának csökkentését mutatja, és az első tag ugyan a kép információit rontja, de a gépi tanulónak még van esélye ezzel is rátanulni a legjellemzőbb információkra.

A javasolt módszerünkben a piramis szintjeinek száma egy hiperparamétere a védekezésre irányuló modellnek. A piramis minden szintjéhez tartozik egy konvolúciós neurális hálózat, amiknek a kimeneteit konkatenáljuk, és ezt egy előrecsatolt mély neurális hálón, pontosabban egy Multi Layer Perceptronon (MLP) keresztül átvezetve kapjuk meg a modell kimenetét.

Az 1. ábra baloldali részén látható Gauss piramis szintjeihez tartozó konvolúciós neurális hálózatokat Stub_n-el jelöltük (ld. 1. ábra középső részén). Úgy alakítottuk ki ezeknek a neurális hálózatoknak a struktúráját, hogy a piramis *n*. szintjéhez tartozó Stub_n utolsó *N-n-1* rétegének architektúrája megegyezik az *n+1*. szintjéhez tartozó teljes Stub_{n+1}-el, ahol *N* a legnagyobb konvolúciós hálózatban (Stub_0) a rétegek száma.

Ezt a Gauss piramissal rendelkező konvolúciós neurális hálózat együttest röviden GP-nek jelölhetjük, a hozzá tartozó paraméterek pedig legyenek röviden $\boldsymbol{\theta}_{\boldsymbol{GP}}$. A Gauss piramis mellett a korábban említett adversarial tanítási módszer alkotja az általunk javasolt eljárást, ahogy ez az alábbi képletben látszik; e két módszer kombinálásból áll a módszerünk, melyet röviden GP+AT-vel jelölünk:

$$J'(\boldsymbol{x}, y, \boldsymbol{\theta}_{\boldsymbol{GP}}) = \alpha J(\boldsymbol{x}, y, \boldsymbol{\theta}_{\boldsymbol{GP}}) + (1-\alpha) J(\tilde{\boldsymbol{x}}, y, \boldsymbol{\theta}_{\boldsymbol{GP}}).$$



1. ábra. Gauss piramist használó konvolúciós hálózat architektúrája

4 Tesztelés és a módszerek összehasonlítása

4.1 A teszteléshez használt jósági mértékek és adathalmaz

A védekezési módszerek összehasonlítására két mértéket használtunk, ami a modell robusztusságát jellemzi. Az egyik a transzferabilitás, másik pedig a pontosság (accuracy), ami azt mutatja meg, hogy mennyi a jól osztályozott minták aránya az összeshez képest. A transzferabilitás [4] mérték azt fejezi ki, hogy egy másik (helyettesítő) modell felhasználásával konstruált támadó zaj (ami sikeresen megtéveszti a helyettesítő modellt), milyen arányban téveszti meg a célmodellt is:

$$\text{transzferabilitás} = \frac{\#\ \text{célmodellt megtévesztő minták}}{\#\ \text{helyettesítő modellt megtévesztő minták}}$$

Itt a célmodellt tehát csak azon minták alapján értékeljük, amelyek a helyettesítő modellt sikeresen megtévesztették. Ha két modellt összevetünk és mindkét irányba kiszámoljuk a transzferabilitást, akkor képet kaphatunk a modellek relatív robusztusságáról – az a modell, amelyiknek kisebb a transzferabilitása a másikhoz képest (és eközben a másik modell transzferabilitása magas), azt robusztusabbnak mondhatjuk. Minden védekezési módszer célja az, hogy a modell transzferabilitását csökkentse és a modell pontosságát támadás jelenléte közben is maximalizálja.

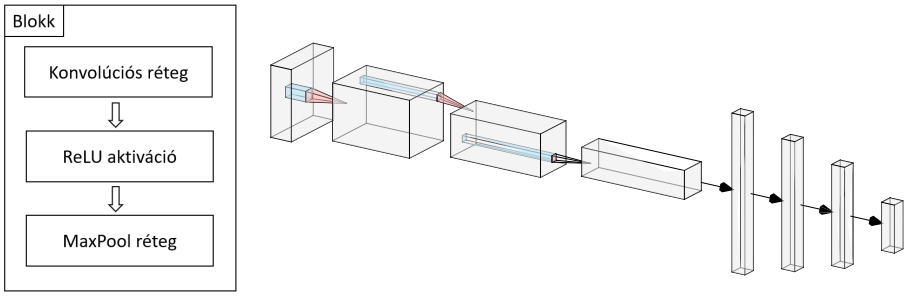
A módszerek összehasonlítását két képadathalmazon végeztük. Az egyik a kézzel írt számjegyeket tartalmazó MNIST adathalmaz [7], melynél a feladat felismerni, hogy milyen számjegyet ábrázol a kép, a másik adathalmaz pedig a CIFAR-10 [5], melyek részletei az 1. táblázatban láthatók.

Az MNIST adathalmazon a feladat egyszerűsége miatt Fully-Connected Feed-Forward hálózatokat használtunk a kiértékelésre. A hálózat rétegeinek mérete rendre 784, 512, 256, 128, 64, 10 voltak. A hálózatok kimenetén softmax aktivációt, a rétegek között pedig az alábbi ReLU aktivációs függvényt használtuk:

$$\mathrm{ReLU}(x) = \begin{cases} x, & \text{ha } x > 0 \\ 0, & \text{különben.} \end{cases}$$

1. táblázat. Kiértékeléshez használt adathalmazok.

Adathalmaz	Képek száma	Képek mérete	Osztályok száma
MNIST	70000	28×28×1	10
CIFAR-10	60000	32×32×3	10



2. ábra. Konvolúciós blokk (baloldalon) és a hálózat architektúrája (jobboldalon)

A CIFAR-10 adathalmaznál az osztályozásra konvolúciós hálózatokat használtunk. Ezeket a hálózatokat a 2. ábrán látható blokkokból építettük fel, a 2. táblázatban látható paraméterekkel, majd az utolsó réteg sorosítása után egy Fully-Connected Feed-Forward hálózat következett 2048, 512, 128, 10 réteg méretekkel.

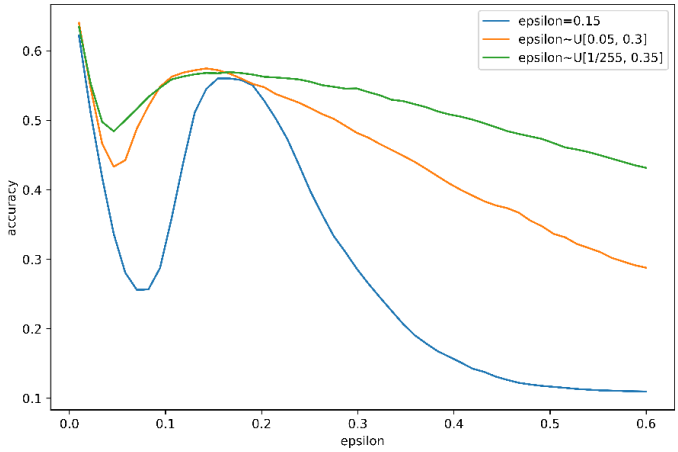
2. táblázat. Konvolúciós blokkok paraméterei

Réteg	Kernel méret	Kernelek száma
1. blokk	5x5	3x64
2. blokk	3x3	64x128
3. blokk	3x3	128x256

4.2 Epszilon értékének vizsgálata a tesztelés előtti tanításnál

A tesztelés elkezdése előtt megvizsgáltuk az epszilon beállítási lehetőségeit, hogy megállapíthassuk milyen értékeket érdemes használni a tanításkor. A mérési eredményeket a 3. ábrán ábrázoltuk (egyszer fix 0,15-nek választva ε értékét, majd két különböző nagyságú intervallumból választva véletlenszerűen), ahol azt láthatjuk, hogy amennyiben ε értékét a tanításkor egy fix értéknek választjuk, akkor a modell pontossága támadás jelenlétében a támadás amplitúdójával nem monoton csökken, hanem van egy lokális maximum annál az ε-nál, amilyen ε értéket a tanuláshoz

választottunk. Ha epszilon értékét egyenletes eloszlásból mintavételezzük minden tanítási *batch*-ben egy szélesebb tartományból, akkor általánosan (kisebb és nagyobb ε-ra is) robusztusabb modellt kapunk. Ebből kiindulva az összes mérésnél véletlenszerű ε-t használtunk a tanításhoz.



3. ábra. Pontosság a CIFAR-10 adathalmazon annak függvényében, hogy tanításkor hogyan választjuk meg epszilon értékét

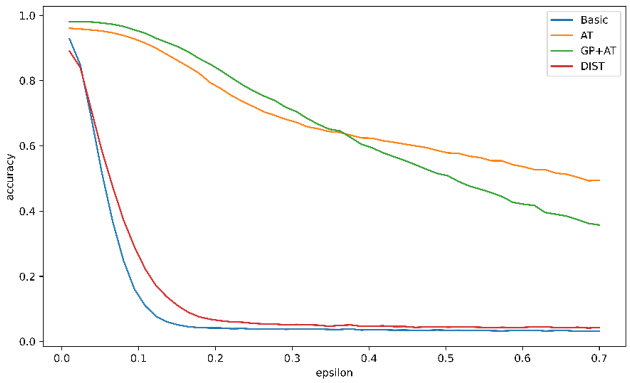
4.3 Kiértékelési eredmények

A teszteléskor a 3. táblázatban látható módszereket hasonlítottuk össze a saját kombinált (GP+AT) módszerrel. A táblázatba beírtuk a használt hibafüggvényeket, valamint a rövidített neveket is, amelyeket az 4. és 5. ábrán látható görbék jelmagyarázatainál használtunk.

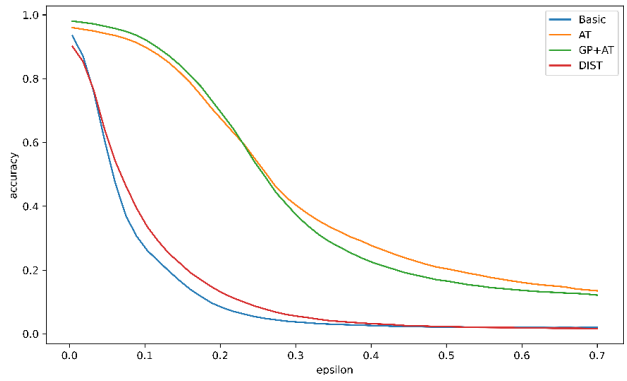
3. táblázat. Összehasonlított módszerek a modellek építésére

Név	Módszer a modellnél	Hibafüggvény
Basic	sima konvolúciós hálózat	keresztentrópia
DIST	sima konvolúciós hálózat	defenzív desztilláció
AT	sima konvolúciós hálózat	adversarial loss
GP+AT	Gauss piramis + konv.	adversarial loss

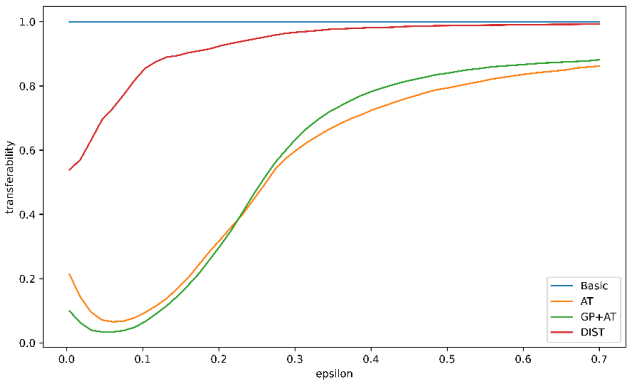
Mind a transzferabilitás, mind a pontosság szerinti ábrákról (4. ábrán a MNIST, 5. ábrán pedig a CIFAR-10 adathalmazon) az olvasható le, hogy a Gauss Piramis használata Adversarial tanítással (GP+AT) kombinálva kis ε értékek mellett növeli a hálózat robusztusságát (akár 5-8% transzferabilitásbeli csökkenés érhető el). A CIFAR-10 adathalmazon egyértelműen látszik a javasolt GP+AT módszerünk előnye minden ε értéknél. A Desztilláció, mint védekezési módszer a mérések alapján nem bizonyult hatékonynak (kicsivel sikerült csak robusztusabb modellt tanítani vele, mint a Basic példában). Az Adversarial tanítás egyedüli védekezési módszerként is jól teljesít, de az általunk javasolt Gauss Piramissal még robusztusabb eredményeket ad.



a) PGD pontosság

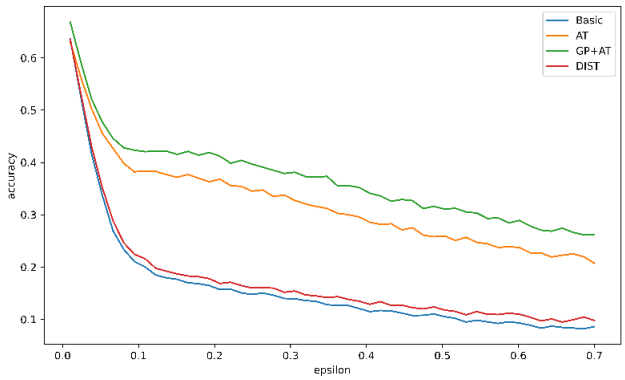


b) FGSM pontosság

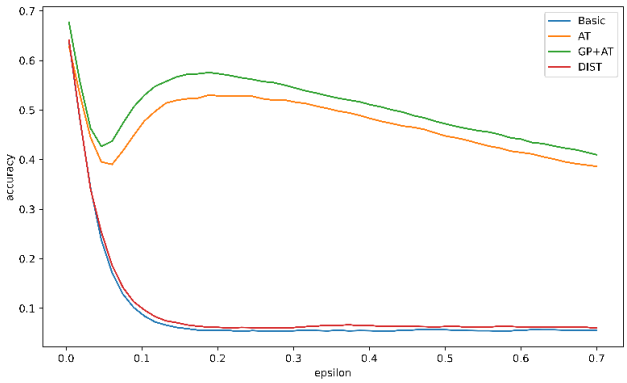


c) FGSM transzferabilitás

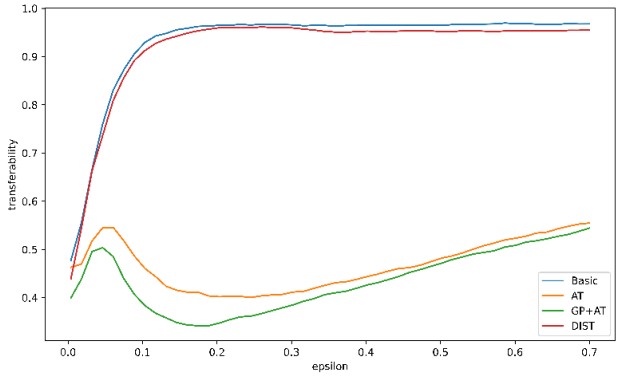
4. ábra. Pontosság FGSM, PGD támadással szemben és FGSM transzferabilitás a MNIST adathalmazon



a) PGD pontosság



b) FGSM pontosság



c) FGSM transzferabilitás

5. ábra. Pontosság FGSM, PGD támadással szemben és FGSM transzferabilitás a CIFAR-10 adathalmazon

Ezen kívül végeztünk egy olyan páros vizsgálatsorozatot is, ahol a támadó nem csak egy egyszerű helyettesítő modellt készít, hanem egy olyan modellt, amely feltételezi, hogy valamilyen védekező módszert használtak a célmodell elkészítésekor (tanításakor). Ha ezen a helyettesítő modellen kísérletezi ki a támadó azokat az adversarial mintákat, amelyek a helyettesítő modellt megtévesztik, akkor ezeknek a bevetése a célmodell ellen erősebb támadást eredményezhet. A páros vizsgálatsorozatot eredménye a 4. táblázatban látható, ahol az oszlopok fejléceiben a támadott (azaz a védekező modellek), a sorok fejléceiben pedig a támadó modellek olvashatók.

A robusztusság vizsgálatánál úgy sorrendezhetjük a védekező módszereket, hogy egy adott támadás mellett növekvő sorrendbe rendezzük a módszereket transzferabilitás szerint (minél kisebb a transzferabilitás, annál robusztusabb a modell az adott támadással szemben). A táblázatból kiolvasható, hogy a GP+AT módszernek a legkisebb minden támadási modellel szemben a transzferabilitása (kivéve saját magánál), amiből azt a következtetést vonhatjuk le, hogy ez a legrobusztusabb módszer mind közül a vizsgált körülményekben.

4. táblázat. MNIST páronkénti transzferabilitás – az oszlopok a támadott, a sorok a támadó modellek ($\varepsilon = 0{,}15$).

Támadó/Cél	Basic	AT	GP+AT	DIST
Basic	0,950	0,369	**0,349**	0,936
AT	0,827	0,761	**0,726**	0,813
GP+AT	0,660	0,832	0,784	**0,653**
DIST	0,989	0,281	**0,238**	0,991

5 Összefoglalás

A cikkünkben egy új védekezési módszert mutattunk be a képosztályozó modelleket támadó irányított zaj alapú támadások ellen. A javasolt módszerünk Gauss képpiramisokon, az azt használó mély neurális hálókon és adversarial tanításon alapul. Két adathalmazon mértük a módszerünk jóságát és összehasonlítottuk más védekezési módszerekkel. A futási eredmények azt mutatták, hogy a Gauss piramisok és adversarial tanítás kombinációja (GP+AT) érte el a legjobb eredményt a pontosság szempontjából. Továbbá megvizsgáltuk a modelleket transzferabilitás szempontjából is, ami a black-box típusú támadások elleni ellenállóképességre utal. A GP+AT módszerünknek volt a legkisebb transzferabilitása minden támadási modellel szemben, amiből azt a következtetést vontuk le, hogy ez a legrobusztusabb módszer mind közül a vizsgált körülmények esetében. A Gauss piramis, mint védekezési módszer hatásossága megnőhet nagyobb felbontású képeken, hiszen a piramis kisebb rétegei is (ahol a támadó zaj hatása már jelentősen csökken) tartalmazhatnak annyi információt, amiből nagy pontosságú hálózat elérhető.

Köszönetnyilvánítás

A kutatás az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg (EFOP-3.6.2-16-2017-00013, Innovatív Informatikai és Infokommunikációs Megoldásokat Megalapozó Tematikus Kutatási Együttműködések).

Irodalomjegyzék

[1] Altrichter M., Horváth G., Pataki B., Strausz Gy., Takács G., Valyon J. (2006). Neurális Hálózatok

[2] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.

[3] Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

[4] Hosseini, H., Chen, Y., Kannan, S., Zhang, B., & Poovendran, R. (2017). Blocking transferability of adversarial examples in black-box learning systems. arXiv preprint arXiv:1703.04318.

[5] Krizhevsky, A., & Hinton, G. (2009). Learning multiple layers of features from tiny images.

[6] Kurakin, A., Goodfellow, I., & Bengio, S. (2017). Adversarial machine learning at scale. ICLR 2017, arXiv preprint arXiv:1611.01236.

[7] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[8] Li, S., Hao, Q., Kang, X., & Benediktsson, J. A. (2018). Gaussian pyramid based multiscale feature fusion for hyperspectral image classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 11(9), 3312-3324.

[9] Liu, S., Wu, H., Lee, H. Y., & Meng, H. (2019). Adversarial attacks on spoofing countermeasures of automatic speaker verification. In 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) (pp. 312-319). IEEE.

[10] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.

[11] Meng, L., Lin, C. T., Jung, T. P., & Wu, D. (2019). White-box target attack for EEG-based BCI regression problems. In International conference on neural information processing (pp. 476-488). Springer, Cham.

[12] Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., & Swami, A. (2017). Practical black-box attacks against machine learning. In Proceedings of the 2017 ACM on Asia conference on computer and communications security, pp. 506-519.

[13] Papernot, N., McDaniel, P., Wu, X., Jha, S., & Swami, A. (2016). Distillation as a defense to adversarial perturbations against deep neural networks. In 2016 IEEE symposium on security and privacy (SP) (pp. 582-597). IEEE.

[14] Vapnik, V., & Chervonenkis, A. (1991). The necessary and sufficient conditions for consistency in the empirical risk minimization method. Pattern Recognition and Image Analysis, 1(3), 283-305.