2. Logistička regresija je diskriminativan linearan klasifikacijski model koji ima probabilističku interpretaciju.

(a) Izvedite izraz za pogrešku unakrsne entropije krenuvši od log-izglednosti na skupu za učenje.

Kao i kod generativnih modela, funkciju pogreške možemo definirati kao negativnu log-izglednost na skupu za učenje: $E(h|\mathcal{D}) = E(\tilde{\mathbf{w}}|\mathcal{D}) = -\ln \mathcal{L}(\mathcal{D}|\tilde{\mathbf{w}})$

Za primjer x, oznaka y je zapravo Bernoullijeva slučajna varijabla, te vrijedi P(y=1|x) = P(C_1|x) = h(x). Razdioba te varijable je: $P(y|\mathbf{x}) = h(\mathbf{x})^y(1-h(\mathbf{x}))^{1-y}$

Funkcija log-izglednosti parametra $\tilde{w}$:

$$\mathcal{L}(\tilde{\mathbf{w}}|\mathcal{D}) = \ln P(\mathcal{D}|\tilde{\mathbf{w}}) = \ln \prod_{i=1}^{N} P(y^{(i)}|\mathbf{x}^{(i)}) = \ln \prod_{i=1}^{N} h(\mathbf{x}^{(i)})^{y^{(i)}}(1-h(\mathbf{x}^{(i)})^{1-y^{(i)}}$$

Funkcija pogreške definirana kao negativna log-izglednost:

$$E(\tilde{\mathbf{w}}|\mathcal{D}) = -\mathcal{L}(\tilde{\mathbf{w}}|\mathcal{D}) = -\sum_{i=1}^{N} \left\{ y^{(i)} \ln h(\mathbf{x}^{(i)}) + (1-y^{(i)}) \ln(1-h(\mathbf{x}^{(i)})) \right\}$$

(b) Po čemu možete zaključiti da je ovo diskriminativan model?

Možemo zaključiti da je logistička regresija diskriminativan model po tome što ona izravno modelira aposteriornu vjerojatnost P(C_j|**x**), dok generativni modeli tu vjerojatnost modeliraju posredno preko zajedničke gustoće P(**x**, C_j).

(c) Kako konvergencija algoritma gradijentnog spusta ovisi o parametru η? Kakav utjecaj ima broj primjera N na parametar η? Kako bismo mogli definirati funkciju pogreške, a da poništimo takav utjecaj?

Što je veći *η* to će gradijentni spust brže konvergirati, no treba biti oprezan. Ako je *η* premalen, algoritam će sporo konvergirati, no s druge strane, ako je *η* prevelik, algoritam će oscilirati ili čak i divergirati.

Budući da je $\nabla E(\tilde{\mathbf{w}}) = \sum_{i=1}^{N} \underbrace{\left(h(\mathbf{x}^{(i)}) - y^{(i)}\right)\tilde{\mathbf{x}}^{(i)}}_{\nabla E_i(\tilde{\mathbf{w}})}$, a promjena $\tilde{\mathbf{w}} \leftarrow \tilde{\mathbf{w}} - \eta \nabla E$,

očita je veća promjena težina u gradijentnom spustu sa većim bojem primjera *N*. Stoga bi trebalo korigirati *η* u ovisnosti o *N*.

Da bismo poništili takav utjecaj, mogli bismo definirati funkciju pogreške kao očekivanje funkcije gubitka $L(h(\mathbf{x}), y) = -y \ln h(\mathbf{x}) - (1-y)\ln(1-h(\mathbf{x}))$.

(d) Neka je skup primjera za učenje

$$\mathcal{D} = \{((1,1),1),((-1,-1),0)\}.$$

Koliko iznosi pogreška unakrsne entropije, a koliko vektor gradijenta $\nabla E(\tilde{\mathbf{w}})$ za $\tilde{\mathbf{w}} = (0,0,0)^{\mathrm{T}}$? Hoće li algoritam gradijentnog spusta konvergirati? Obrazložite odgovor.

$E(\tilde{\mathbf{w}}|\mathcal{D})$ = 1.3863 $\quad\quad$ $\nabla E(\tilde{\mathbf{w}}) = \begin{bmatrix} 0 \\ -1 \\ -1 \end{bmatrix}$

Ne znam za vas, ali meni je izkonvergirao. Sve ovisi o tome kakav eta se uzme, ako je eta prevelik, težine će samo rasti i rasti, a ako je 'dobar', stvar će izkonvergirati prije ili kasnije.

(e) Hoće li algoritam gradijentnog spusta u situaciji iz zadatka 2d konvergirati ako koristimo regulariziranu funkciju pogreške? Zašto?

Hoće, jer kad dodamo regularizaciju, sprječavamo prenaučenost u modelu, tj. sprječavamo da težine poprime velike vrijednosti, jer time povečavaju pogrešku. Sprječavamo tako da kažnjavamo veće težine sa većom vrijednosti funkcije pogreške.

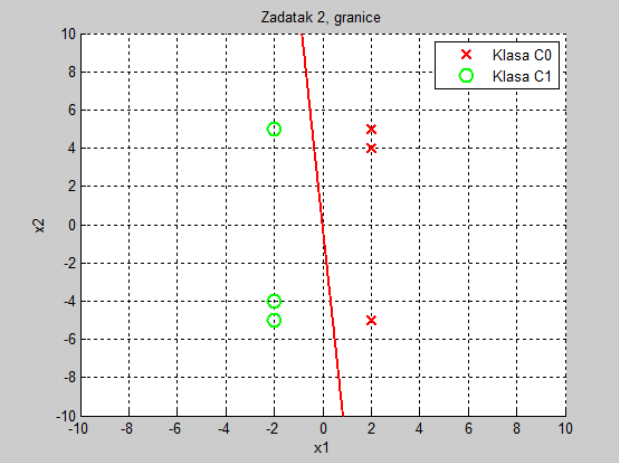
(f) U općenitom slučaju, ako povećamo vrijednost parametra regularizacije λ, hoće li se vrijednost empirijske pogreške povećati ili smanjiti? Obrazložite odgovor.

U općenitom slučaju, povećanjem parametra regularizacije λ povećavamo vrijednost empirijske pogreške. Zašto? Povećanjem regularizacije jače kažnjavamo složenije modele, tj. povećanjem regularizacije smanjujemo složenost modela. A svi mi znamo da se složeniji modeli bolje prilagođavaju skupu za učenje i time imaju manju empirijsku pogrešku.

(g) Neka je skup primjera za učenje

$$\mathcal{D} = \{((2,5),0),((2,4),0),((2,-5),0),((-2,-5),1),((-2,-4),1),((-2,5),1))\}.$$

Skicirajte ovaj skup i granicu koju dobivamo modelom logističke regresije (granicu možete lako nacrtati ako se prisjetite koja je veza između logističke regresije i generativnog modela). Komentirajte: kako je moguće da logistička regresija ne uspijeva ispravno klasificirati linearno odvojive primjere?



Slika 1. *Primjeri i granica koja odvaja klase*

Slično kao i u d) podzadatku, izkonvergirao mi je zadatak. Nije savršena granica, ali da se namjestiti uz bolju etu.

Dakle, ja nemam problema sa ovim zadacima, ako mi je eta dobar, dobit ću rješenje, nije mi se događalo da gradijentni spust ne konvergira za linearno odvojive razrede. Ali spomenut ću samo to da “postoji” mogućnost da algoritam ne konvergira na linearno razdvojivim razredima.

Naime, ako su primjeri za učenje linearno odvojivi, gradijent pogreške nikada neće biti jednak nuli (tj. funkcija pogreške nema minimuma). Posljedično, gradijentni spust neće konvergirati i težine će rasti prema beskonačnosti. Sigmoida će time postajati sve strmija, njezini će izlazi biti sve bliži vrijednostima 0 i 1, pa se tako gubi blagi prijelaz izmedu klasa.

Mada opet velim, to se meni nije događalo i osobno smatram da se ovaj problem neće pojaviti uz dobru stopu učenja.