Raspoznavanje uzoraka – 1.MI 2o1o/2o11

Q: jel netko možda rješavao onaj ispit iz 2008. koji je na materijalima? muči me drugi zadatak, 2 razreda, a treba koristiti algoritam perceptrona za više od 2 razreda? jel zna netko kako to isprivno napraviti jer imam 2 uzorka u jednom razredu i neznam kaj s tim napravit.

A: <u>(krezubica)</u>
Evo algoritma perceptrona za vise od 2 razreda: (to je moj algoritam iz biljeznice iz URU prosla godina, meni je moj zapis citljiv, nadam se da ce i vama ... :))
istodobno ugadjamo sve tezinske koeficjente
uzorak se razvrstava u razred ako je di(x(k)) > dj(x(k)) za sve i != j
uzorke samo numeriras redom... uzorci prvog razreda su x1, x2... xn1, gdje je n1 broj uzoraka prvog razreda, pa su uzorci drugog x(n1 + 1) , x(n1 +2), ..x(n1 + n2) , gdje je n2 broj uzoraka drugog razreda i tako ... nadam se da se kuzi kaj mislim :)

k-ti korak postupka:

radimo sa:
x(k) element wi , uzorak iz skupa za ucenje (uzimaju se svi uzorci redom, kruzno)
di - decizijskom funkcijom za wi
di(x(k)) - vrijednost decizijske funkcije za wi za k-ti korak/uzorak

za svaki dj(x) element {d1(x), d2(x)... dM(x)} gdje je M broj razreda (i decizijskih f-ji takodjer)

racunamo: {
dj(x(k)) = Wj(T)(k)*x(k) , gdje oznaka (T) oznacava transponirano
ako je di(k(x)) > dj(x(k))
- tezinski vektor se ne popravlja,
Wj(k+1) = Wj(k)

ako je di(x(k)) <= dj(x(k))
- popravlja se vrijednost
Wj(k+1) = Wj(k) - cx(k)

}
ako smo popravili bilo koji Wj(k):
- korekcija tezinskog faktora Wi
Wi(k+1) = Wi(k) + cx(k)

inace:
-nema korekcije
Wi(k+1) = Wi(k)

Q: U tom drugom zadatku, drugi dio pitanja je da usporedimo da li bi dobili isti rezultat postupkom učenja za slučaj 2 razreda, i zašto je tako. Pa kako da usporedimo kad u prvom slučaju imamo 2 decizijske fje a u drugom samo jednu? Ili je to možda dovoljno odgovorit, da je razlika u broju decizijskih fja?

EDIT: evo sadm sa riješio, ispada da se ova 2 decizijska pravca koja se dobiju općenitim postupkom poklapaju, a i njih dva zajedno se poklapaju s ovim koji se dobiju postupkom za 2 klase. Zna netko kako bi to onda točno trebalo objasnit na ispitu?

A: (krezubica)
pa eto, upravo to sto si napisao.samo jos dodas zasto se poklapaju -- a to se vidi iz postupka. kad god radis korekciju na jednom, radis (suprotnu) korekciju na drugom, dakle jasno da se moraju poklapat. i onda usput komentiras da se na isti nacin mi u svakom koraku obicnog algoritma radila korekcija (tj. isto kad i ovdje), pa dakle, vidite, isto je :)

Q: Kod perceptrona s djelomicnom korekcijom da li se takoder uzima najmanje cijelo vece od |wT * x| / (xT * x) kao i kod onog s apsolutnom korekcijom? I ako ne zasto vrijedi pri dnu 15. slajda 3. prezentacije: za lambda > 1 uzorak se ispravno klasificira nakon svakog ugadanja? Jer ako je wT * x = 0, onda se bez obzira na lambda nista ne dogada.

A: (krezubica)
tu se, isto kao i u 2) slucaju , zahtjeva da w(0) != 0 , tj. da pocetna vrijednost pa tako ni umnozak nisu 0. e, sad, kaze 0 < lambda < 2 opcenito, s tim da za 1 < lambda < 2 dobivamo slucaj 2) tj. apsolutnu korekciju. to je zato sto funkcija najvece cijelo u NAJGOREM slucaju (kad "najmanje povecava") vraca upravo svoj argument:
ciel (1.0) = 1.0
ciel (2.0) = 2.0
a u najboljem, (skoro) argument +1:
ciel (1.1) = 2.0
ciel (1.0000001) = 2.0
ciel (3.00001) = 4.0

tj. ciel se moze napisati kao: f(x) = lambda*x; x < lambda*x < x + 1

sad, posto ovo u zagradi (pretpostavljam) u najgorem slucaju moze biti 1, tj. najmanji argument funkcije ciel (u ovoj primjeni) je upravo 1, dobivamo:
x=1:
1 < lambda*! < 2, dakle, zato se za lambda > 1 svodi na drugi slucaju, a za 0 < lambda < 1 samo konvergira

(smiley)
Cini mi se da u 2. slucaju moze biti w(0) = 0. Bar nigdje nisam vidio da pise da ne moze. Zar ne moze umnozak biti 0 i ako je w(0) != 0?

(krezubica)

ta cjela razrada / podjela ti je izvedena gradijentnim postupkom na stranicama 12.-13. treceg predavanja (ko sto primjecujes forumla je ista) i sva rasprava koja tu vrijedi za lambda vrijedi i kod ona 3 slucaja

(Senkyen)

→ !!

asistent je na auditornima za slučaj apsolutne korekcije rekao da ne uzmemo baš funckiju ceil već da gledamo baš prvi cijeli broj strogo veći od onog izraza, tj. ako izraz ispadne npr. 0 mi ćemo uzeti vrijednost 1 za C, ako ispadne izraz 1 mi ćemo uzeti 2 itd. tako da w(1) može biti nul vektor i sve će raditi ok.

za djelomične korekcije asistent nije ništa zaokruživao, pa mi je to malo čudno jer je onda često korekcija nula (iako nije uzorak dobro klasificiran) i tu si ne možemo pomoći da maknemo decizijsku funkciju koja nam baš prolazi kroz taj uzorak, ali algoritam doslovno tako piše, pa ko sam ja da se bunim ;)

1.MI 2oo8.

(DeathClaw)

→može netko provijeriti rez iz 1.mi 2008 ()

dobio sam

{0 za w^T*x >=0

w(k+1) = w(k) + c { -2 * x *|w^T * x|^3 / (x^T * x) za w^T * x <0

(PAZI:

-- jel tu uvrštavamo u w(k+1)=w(k)+c{...} ili w(k+1)=w(k)-c{...}?

-- da...mislim da minus treba biti...)

2. zad sam dobio w1 = [2 -1 2] ; w2 = [-2 1 -2]

→ !!

2. zad:

omega1 = {x1 , x2}

omega2 = {x3}

x1 = [0 0 1]; x2 = [111]; x3 = [-1 2 1]

w1(1) = w2(1) = w(3) = 1; c =1

1) uzemo uzorak x1

d1 = w1^T * x1 = 0

d2 = w2^T * x1 = 0

d1 <= d2 -> korekcija, w1 povečaj, w2 smanji

w1(2) = w1(2) + c * x1 = [0 0 1]^T

w2(2) = w2(2) - c * x1 = [0 0 -1^T]

2) uzmemo uzorak x2

d1 = 1; d2 = -1 d1 > d2 -> nema korekcije

w1(3) = w1(2)
w2(3) = w2(2)

3) uzmemo uzorak x3 (pazi x3 € omega2)
d1 = 1
d2 = -1 d2 < d1 -> korekcija -> w2 povečaj w1 smanji

w1(4) = w1(3) - 1 * x3 = [1 - 2 0]^T
w2(4) = w2(3) + 1 * x3 = [-1 2 0]^T

----
prošli smo sve uzorke, bilo je korekcija, nastavljamo
x4 = x1; x5 = x2; x6 = x3; x4, x5 € omega1, x6 € omega2

4) uzmemo x4
d1 = 0; d2 = 0 -> d1<= d2 -> korekcija -> povečaj w1, smanji w2
w1(5) = w1(4) + x4 = [1 -2 1]^T
w2(5) = w2(4) - x4 = [-1 2 1]^T

5) uzmemo x5
d1 = 0; d2 = 0; d1 <= d2 -> korekcija; w1 povećaj, w2 smanji
w1(6) = [2 -1 2]
w2(6) = [-2 1 -2]

6) uzmemo x6 (pazi x6 € omega2)
d1 = -2; d2 = 2 -> d2 > d2 -> nema korekcije
w1(7) = w1(6)
w2(7) = w2(6)

----
x7 = x1; x8=x2; x9 = x3

7) x7
d1 = 2; d2 = -2 -> d1 > d2 -> nema korekcije

8)x8
d1 = 3; d2 = -3 -> d1 > d2 -> nema korekcije

----
prošli smo kroz sve primjere bez korekcije: KRAJ
uzimamo zadnje izračunate w1 i w2 a to su w1 = [2 -1 2] ; w2 = [-2 1 -2]