Katedra za Signale i sisteme
Elektrotehnički fakultet
Univerzitet u Beogradu

Mašinsko učenje

Treći domaći zadatak

Metod nosećih vektora

Student:
Aleksandar Đorđević 2023/3049

Mentor:
dr. Predrag Tadić

1 Uvod

Osnovna ideja metoda nosećih vektora (*support vector machine*) jeste maksimizacija margine odnosno za linearno separabilne klase treba provući pravu između njih tako da je ona najviše moguće udaljene od najbiližih primera. Uvođenjem šarka gubitaka moguće je rešiti i probleme koji nisu linearno separabilni, a uvođenjem nelinearnih kernela moguće je dobiti i nelinearne klasfikacione krive. Klasifikator je moguće projektovati na dva načina rešavanjem **primalnog** odnosno **dualnog** problema.
Pre projektovanja klasifikatora podaci su standardizovani i podeljeni na trening i test skup.

2 Primalni problem

Funkcionalna margina se definiše kao:

$$\hat{\gamma}^{(i)} = y^{(i)}(\omega^T x^{(i)} + b) \tag{1}$$

Ukoliko je ona pozitivna primer je ispravno klasifikovan tako da se uslov da svi primeri budu ispravno klasifikovani (što je moguće za linearno separabilne probleme) može zapisati kao $\hat{\gamma}^{(i)} > 0, \forall i$. Takođe, lako se pokazuje da ukoliko je uspešno zadovoljen uslov funkcionalna margina se može proizvoljno povećati prostim skaliranjem klasifikacione prave te se gorenavedeni uslov obično piše kao $\hat{\gamma}^{(i)} \geq 1, \forall i$.
Kako za geometrijsku marginu važi $\gamma = \frac{\hat{\gamma}}{||\omega||}$ uslovnom minimizacijom norme vektora ω se može maksimizovati geometrijska margina pod uslovom perfektne klasifikacije. Ovo je problem linearnog programiranja čije rešenje često može biti nestabilno te se norma kvadrira kako bi se problem mogao rešiti kvadratnim programiranjem.
Ovako definisan problem ne dozvoljava pogrešno klasifikovane primere te je u praksi gotovo neupotrebljiv. Kako bi omogućili i pogrešnu klasifikaciju pojedenih primera uvode se šarka gubici koji dozvoljavaju funkcionalnoj margini da postane negativna. Koeficijent koji penalizuje šarka gubitke se obeležava sa C i predstavlja hiper parametar modela. Problem koji je rešavan u ovom zadatku izgleda ovako:

$$\min_{\omega,b,\xi}(\frac{1}{2}||\omega||^2 + C\sum_{i=1}^{m}\xi_i) \tag{2}$$

Pod uslovima:

$$\hat{\gamma}^{(i)} \geq 1 - \xi_i, \xi_i \geq 0 \tag{3}$$

Kako je već navedeno ovo je problem kvadratnog programiranja i rešen je pomoću *Python* biblioteke **cvxopt**. Solver ove biblioteke minimizuje funkciju $\frac{1}{2}x^T P x + q^T x$ po x uz opciona ograničenja $Ax = b$ i $Gx \leq h$.
U vektor x smeštamo sve promenljive po kojima optimizujemo to jest b, ω i ξ.

$$x = [b \ \ \omega_1 \ldots \omega_n \ \ \xi_1 \ldots \xi_m]^T \tag{4}$$

Za ovako definisan vektor x ostale matrice izgledaju ovako:

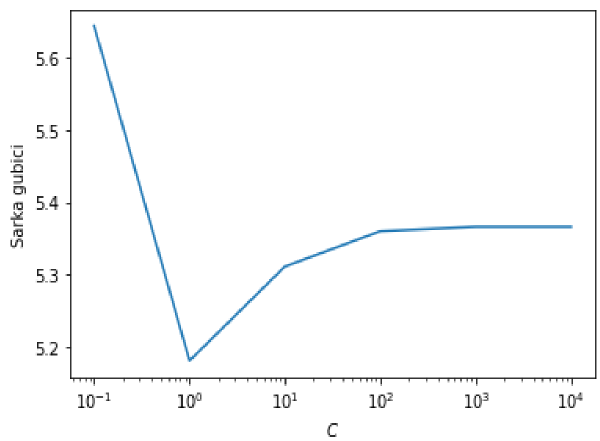
$$P = \begin{bmatrix} 0 & \ldots & \ldots & \ldots & 0 \\ 0 & 1 & \ldots & \ldots & 0 \\ \vdots & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, q = [\underbrace{0 \ldots 0}_{n+1}\underbrace{C \ldots C}_{m}]^T \tag{5}$$

$$G = -\begin{bmatrix} y^{(1)} & y^{(1)}x_1^{(1)} & \dots & y^{(1)}x_n^{(1)} & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ y^{(m)} & y^{(m)}x_1^{(m)} & \dots & y^{(m)}x_n^{(m)} & 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 1 \end{bmatrix}, h = -\,[\underbrace{1\dots1}_{m}\underbrace{0\dots0}_{m}]^T \tag{6}$$

Rešavanjem se dobijaju vrednosti za optimalno b, ω i ξ. Primer se klasifikuje u klasi ±1 na osnovu sledeće hipoteze:

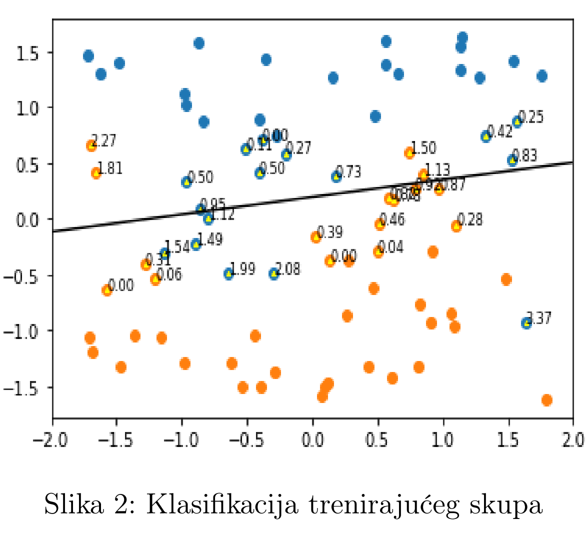
$$h(x) = sign(\omega^T x + b) \tag{7}$$

Hiper-parametar C je određen pomoću k-fold krosvalidacije za $k = 6$ i $C \in \{0.1, 1, 10, 100, 1000, 10000\}$ minimizaciojom sume šarka gubitaka. Na osnovu sledećeg grafika dobijena je vrednost $C = 1$.



Slika 1: Zavisnost sume šarka gubitaka od parametra C

Noseći vektori su oni čija je funkcionalna margina manja ili jednaka 1 (zbog numeričke greške uzeta je tolerancija od 10^{-6}). Na sledećoj slici su prikazani odabirci iz trenirajućeg skupa (noseći vektori su označeni žutim trouglovima i navedeni su njihovi šarka gubici) zajedno sa separacionom krivom.



Slika 2: Klasifikacija trenirajućeg skupa

Na trenirajućem skupu je ostvarena tačnost od 87.5%, dok je tačnost na testirajućem skupu 95%.

3 Dualni problem

Dualni problem je ekvivalentan primalnom samo pod određenim uslovima definisanim *Karush-Kuhn-Tucker*-ovom teoremom. Kada su oni ispunjeni dualni problem je praktičniji zbog lakog uvođenja nelinearnosti pomoću kernel trika. Naime, metod nosećih vektora može dati isključivo linearan klasifikator za data obeležja, međutim ukoliko prebacimo problem u više dimenzija (dodavanjem izvedenih obeležja) to projektujemo klasifikacionu hiper-ravan povratkom u originalan prostor, u opštem slučaju, dobijamo nelinearnu klasifikacionu krivu. Kernel trik je značajan jer nam omogučava da efektivno dodamo nova obeležja bez njihovog direktnog izračunavanja, većsamo računanjem kernel funkcije. U ovom zadatku je korišćen polinomijalni kernel:

$$K(x1, x2) = (< x1, x2 > +c)^d \tag{8}$$

Uzeto je $c = 1$ i $d = 6$. U dualnom problemu se vrši uslovna maksimizacija po koeficijentima α.

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y^{(i)} y^{(j)} K(x^{(i)}, x^{(j)}) \tag{9}$$

Uz uslove:

$$\sum_{i}^{m} \alpha_i y^{(i)} = 0, 0 \le \alpha_i \le C \tag{10}$$

Ovo je isto problem kvadratnog programiranja te je rešen isto pomoću cvxopt biblioteke. Kako je u pitanju maksimizacija vektori P i q su pomnoženi sa -1.
Ako za promenljivu po kojoj optimizujemo usvojimo $x = \begin{bmatrix} \alpha_1 & \dots & \alpha_m \end{bmatrix}^T$ ostale matrice izgledaju ovako:

$$P = \left[y^{(i)} y^{(j)} K(x^{(i)}, x^{(j)}) \right], i, j \in [1, m], q = -[\underbrace{1 \dots 1}_{m}]^T \tag{11}$$

$$G = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \\ -1 & 0 & \dots & 0 \\ 0 & -1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -1 \end{bmatrix}, h = [\underbrace{C \dots C}_{m} \underbrace{0 \dots 0}_{m}]^T \tag{12}$$

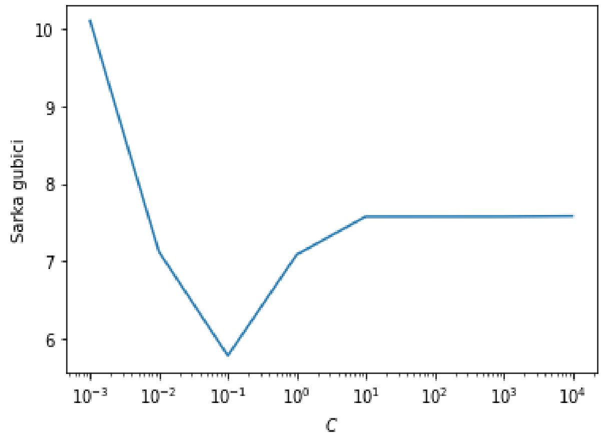
$$A = y, b = 0 \tag{13}$$

Rešavanjem ovog problema dobijaju se optimalne vrednosti za α, dok se b dobija iz sledeće formule:

$$b = y^{sv} - \sum_{i}^{m} \alpha_i y^{(i)} K(x^{(i)}, x^{sv}) \tag{14}$$

Noseći vektor (svi oni sa nenultim α uz toleranciju) se bira tako da mu funkcionalna margina bude 1 (opet uz toleranciju zbog numeričke greške).
Sada se klasifikacija vrši pomoću sledeće hipoteze:

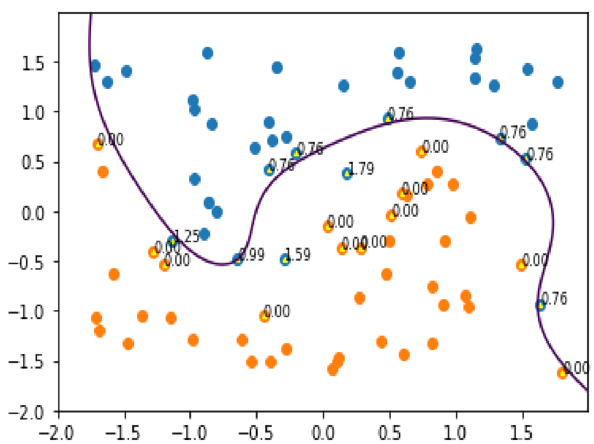
$$h(x) = sign(\sum_{i=1}^{m} \alpha_i y^{(i)} K(x^{(i)}, x) + b) \tag{15}$$



Slika 3: Zavisnost sume šarka gubitaka od parametra C

Izbor C je izvršen k-fold krosvalidacijom sa $k = 6$ i $C \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000\}$. Na osnovu sledećeg grafika dobijeno je $C = 0.1$.

Na sledećoj slici prikazani su odabirci iz trenirajućeg skupa (noseći vektori su obeleženi žutim trouglovima i istaknuti su njihovi šarka gubici) uz separacionu krivu.



Slika 4: Klasifikacija trenirajućeg skupa

Tačnost na trenirajućem skupu iznosi 96.25%, a na testirajućem 95%.