7. Logistička regresija II

Strojno učenje 1, UNIZG FER, ak. god. 2022./2023.

Jan Šnajder, vježbe, v1.10

1 Zadatci za učenje

1. [***Svrha: Znati izvesti algoritam multinomijalne logističke regresije.***]

   (a) Definirajte funkciju *softmax*. Izračunajte softmax($\boldsymbol{\alpha}$) za ulazni vektor $\boldsymbol{\alpha} = (2, 8, 1, 5)$. Koja su dva efekta funkcije softmax?

   (b) Definirajte model multinomijalne logističke regresije.

   (c) Izvedite pogrešku modela multinomijalne logističke regresije kao negativan logaritam vjerojatnosti oznaka koje model dodjeljuje primjerima iz skupa označenih primjera.

2. [***Svrha: Znati izvesti algoritam LMS poopćenih linearnih modela. Razumjeti prednosti tog algoritma.***]

   (a) Izvedite pravilo za ažuriranje težina algoritma LMS (engl. *least mean squares*) kao gradijent funkcije gubitka, i to za (i) model linearne regresije i (ii) model logističke regresije.

   (b) Objasnite prednost algortima LMS (odnosno stohastičkog gradijentnog spusta) nad grupnim (*batch*) gradijentnim spustom.

3. [***Svrha: Uočiti zajedničkosti poopćenih linearnih modela.***]

   (a) Opišite veze između (i) modela linearne regresije, logističke regresije i multinomijalne logističke regresije, (ii) distribucija zavisne varijable y i (iii) aktivacijskih funkcija f. Što je zajedničko svim distribucijama s kojima smo dosada radili?

   (b) Objasnite riječima ovaj izraz:

$$\mathbf{w}^* = \operatorname*{argmax}_{\mathbf{w}} \ln P(\mathcal{D}|\mathbf{w}) = \operatorname*{argmin}_{\mathbf{w}} E(\mathbf{w}|\mathcal{D})$$

4. [***Svrha: Razumjeti motivaciju za adaptivne bazne funkcije i vezu između poopćenih linearnih modela i modela neuronske mreže.***]

   (a) Objasnite što su bazne funkcije i koji je problem s fiksnim baznim funkcijama.

   (b) Definirajte poopćeni linearnim model s proizvoljnom aktivacijskom funkcijom f koji kao bazne funkcije koristi poopćene linearne modele s istom takvom aktivacijskom funkcijom. Načinite skicu takvog modela, odnosno dvoslojne neuronske mreže. Na skici naznačite komponente ulaznog vektora, težine modela, i bazne funkcije.

   (c) Koja je prednost ovakvog modela u odnosu na (i) poopćeni model bez baznih funkcija i (ii) poopćeni model s fiksnim baznim funkcijama? Koji je nedostatak takvog modela u odnosu na poopćeni model s fiksnim baznim funkcijama?

2 Zadatci s ispita

1. (T) Kod logističke regresije optimizaciju tipično provodimo gradijentnim spustom. Međutim, kod linearne regresije optimizaciju smo provodili izračunom pseudoinverza matrice dizajna. **Zašto**

optimizaciju kod logističke regresije također ne provodimo izračunom pseudoinverza matrice dizajna?

A Optimizaciju parametara linearne regresije također možemo napraviti gradijentnim spustom po empirijskoj pogrešci, ali to ne radimo jer izračun pseudoinverza ima manju računalnu složenost

B Maksimizacije logaritma vjerojatnosti ispravnih oznaka logističke regresije ne daje izraz u zatvorenoj formi koji sadržava pseudoinverz matrice dizajna

C Zbog nelinearnosti logističke funkcije, kod logističke regresije izračun pseudoinverza matrice dizajna nije moguće napraviti u zatvorenoj formi

D Optimizaciju možemo provesti izračunom pseudoinverza matrice dizajna, međutim, za razliku od gradijentnog spusta, taj postupak ne funkcionira kada je matrica dizajna singularna

2. (T) Kod logističke regresije za optimizaciju parametara tipično koristimo gradijentni spust ili Newtonov optimizacijski postupak. **Što su prednosti, a što nedostatci gradijentnog spusta u odnosu na Newtonov postupak, i to konkretno kod logističke regresije?**

A Za razliku od Newtonovog postupka, gradijentni spust može se koristiti za "online" (pojedinačno) učenje, no može krivudati i zato sporije konvergirati od Newtonovog postupka

B Za razliku od Newtonovog postupka, gradijentni spust može se koristiti za L2-regulariziranu logističku regresiju, no ako je stopa učenja prevelika, postupak može divergirati, dok Newtonov postupak nema taj problem

C Newtonov postupak brže konvergira, ali se može koristiti samo za konveksnu funkciju pogreške, dok gradijentni spust nema tog ograničenja, ali može zaglaviti u lokalnom optimumu

D Gradijentni spust znatno je računalno jednostavniji od Newtonovog postupka, no za razliku od Newtonovog postupka kod L2-regularizirane regresije ne konvergira ako primjeri nisu linearno odvojivi

3. (T) Kod Newtonovog postupka optimizacije za logističku regresiju ažuriranje težina provodi se prema sljedećem pravilu:

$$\mathbf{w} \leftarrow \mathbf{w} - \mathbf{H}^{-1}\nabla E(\mathbf{w}|\mathcal{D})$$

Očito, za provođenje ovog postupka potrebno je računati inverz Hesseove matrice, tj. matrice parcijalnih derivacija $\mathbf{H}$. Općenito, operacija invertiranja matrice nije uvijek izvediva, a čak i kada jest izvediva, rješenje nije uvijek numerički stabilno. **Kod logističke regresije, koji je nužan i dovoljan uvjet za izvedivost i numeričku stabilnost Newtonovog optimizacijskog postupka?**

A Značajke moraju biti linearno zavisne

B U podatcima ne smije biti multikolinearnosti

C Broj primjera mora biti veći od broja značajki plus jedan

D Funkcija pogreške mora biti konveksna

4. (T) Svi poopćeni linearni modeli mogu se trenirati u "online" (pojedinačnom) načinu, primjenom algoritma LMS. To vrijedi i za algoritam linearne regresije, za koji smo prvotno optimizaciju provodili računajući pseudoinverz matrice dizajna. Jedna od prednosti algoritma LMS u odnosu na izračun pseudoinverza kod linearne regresije je manja računalna složenost LMS-a. Neka E označava broj epoha, N je broj primjera, a m broj značajki u prostoru značajki. **Koja je (vremenska) računalna složenost algoritma LMS, primijenjenog na linearnu regresiju?**

A $\mathcal{O}(ENm)$ B $\mathcal{O}(E(N+m))$ C $\mathcal{O}(EN^2m)$ D $\mathcal{O}(ENm^2)$

5. (T) Problem višeklasne ($K > 2$) klasifikacije logističkom regresijom možemo riješiti na više načina. Možemo primijeniti (1) multinomijalnu logističku regresiju (MLR), (2) binarnu logističku regresiju

sa shemom OVO (BLR-OVO) ili (3) binarnu logističku regresiju sa shemom OVR (BLR-OVR). **Koja je prednost MLR nad BLR-OVO i BLR-OVR?**

A MLR ima više parametara od BLR-OVR, ali nije osjetljiva na neuravnoteženost broja primjera po klasama

B Za razliku od BLR-OVR i BLR-OVO, kod MLR ne postoje područja u ulaznom prostoru za koje klasifikacijska odluka nije određena

C MLR i BLR-OVR imaju manje parametara od BLR-OVO, no jedino za MLR vrijedi $\sum_k P(y = k|\mathbf{x}) = 1$

D Za razliku od BLR-OVO i BLR-OVR, MLR koristi funkciju softmax, pa minimizacija L1-regularizirane pogreške ima rješenje u zatvorenoj formi

6. (N) Raspolažemo označenim skupom primjera iz triju klasa ($K = 3$) u trodimenzijskome ulaznom prostoru ($n = 3$). Na tom skupu treniramo model multinomijalne logističke regresije. Treniranje provodimo gradijentnim spustom. U nekoj od iteracija gradijentnog spusta, matrica težina je sljedeća (stupci odgovaraju težinama za pojedine klase):

$$\mathbf{W} = \begin{pmatrix} 1 & 1 & 1 \\ 2 & 0 & -2 \\ 3 & -4 & 6 \\ -3 & 0 & 2 \end{pmatrix}$$

Jedan od primjera u skupu za učenje je primjer $\mathbf{x} = (3, 2, -1)$ s oznakom $\mathbf{y} = (0, 1, 0)$. **Koliko iznosi gubitak unakrsne entropije koji u ovoj iteraciji optimizacijskog postupka nanosi dotični primjer?**

A 7 B 11 C 23 D 35

7. (P) Poopćeni linearni modeli mogu koristiti adaptivne bazne funkcije. Prednost toga je da ne moramo ručno definirati preslikavanje $\boldsymbol{\phi}$ u prostor značajki, već se to preslikavanje može naučiti na temelju podataka. Rasplažemo podatcima iz $K = 3$ klase u 10-dimenzijskome ulaznom prostoru. Za taj višeklasni problem koristimo multinomijalnu logističku regresiju, ali s adaptivnim baznim funkcijama. Svaka adaptivna bazna funkcija ϕ_j parametrizirana je kao skalarni produkt vektora značajki i vektora primjera, kao što smo radili na predavanjima. Naš model definirali smo ovako:

$$h_k(\mathbf{x}) = \text{softmax}_k\Big(\sum_{j=0}^{3} w_{j,k}\phi_j(\mathbf{x})\Big)$$

Ovime je definirana hipoteza za klasu k. Svaka klasa ima svoju hipotezu h_k. Svaka klasa ima i svoje težine $w_{j,k}$. Međutim, bazne funkcije ϕ_j zajedničke su za sve klase (dakle, ti parametri su dijeljeni između klasa). **Koliko ukupno parametara ima ovaj model?**

A 45 B 49 C 136 D 142