Optimizari
Laborator 3: Metoda Newton

1 Problema UNLP - Metoda Newton

Fie problema de optimizare fara constrangeri:

$$x^* = \arg\min_{x \in \mathbb{R}^n} F(x) \tag{1}$$

O metoda clasica de rezolvare o reprezinta metoda Newton. Aceasta este o metoda de ordinul 2, i.e foloseste informatia de gradient si hessiana. Pseudocodul metodei Newton:

Algoritmul Newton

Date de intrare : $\boldsymbol{x}_0\,(t=0)$ punctul initial, pasul $\alpha_t > 0$

1. Atata timp cat `criteriu(`x_t`)` $\geq \epsilon$:

   1.1 $\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \alpha_t(\nabla^2 F(\boldsymbol{x}_t))^{-1}\nabla F(\boldsymbol{x}_t)$

   1.2 $t = t + 1$

2.Returneaza x_{t+1}

unde `criteriu(`x_t`)` se numeste criteriu de oprire si poate fi:
$\|\nabla F(x_t)\|$ sau $|F(x_t) - F^*|$ sau $\|x_{t+1} - x_t\|$

Pentru problema (1), criteriul de oprire ideal este norma gradientului, intrucat din conditiile de optimalitate de ordinul 1 se stie ca $\nabla F(x^*) = 0$.

2 Regresia logistica. Formularea problemei

Regresia logistică (*Logistic regression*) este un algoritm important de învățare automată. Scopul este modelarea probabilității ca o variabila aleatoare Y să fie 0 sau 1 prin prisma datelor experimentale. Pentru aceasta se introduce mai intai o *funcția de activare sigmoid* de forma:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

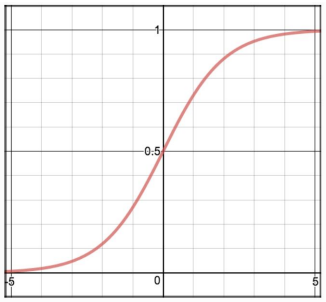


Figure 1: Functia sigmoid standard $\sigma(z)$ cu $z \in [-6, 6]$ si $\sigma(z) \in [0, 1]$.

Ea face maparea intre valoarea de predicție (o valoare reala) si probabilitate (o valoare între 0 și 1), a se observa Figura 1.
Luăm în considerare ca model o funcție liniar generalizată parametrizată în w:

$$h_w(X) = \sigma(w^T X) = \frac{1}{1 + e^{-w^T X}} = \mathrm{P}(Y = 1|X, w)$$

unde $P(Y = 1|X, w)$ este probabilitatea condiționată. Ținând cont de teoria fundamentală a probabilitatii (i.e. suma probabilitatilor este 1, $\sum P(Y|X, w) = 1$), avem ca:

$$\mathrm{P}(Y = 0|X, w) = 1 - h_w(X).$$

Acum calculăm funcția de probabilitate (likelihood) presupunând că toate observațiile sunt independente, Bernoulli distribuite:

$$\begin{aligned} L(w|y, x) = \mathrm{P}(Y|X, w) &= \prod_{i=1}^{N} \mathrm{P}\left(y_i|x_i, w\right) \\ &= \prod_{i=1}^{N} h_w\left(x_i\right)^{y_i} \left(1 - h_w\left(x_i\right)\right)^{(1-y_i)}. \end{aligned}$$

De obicei, logaritm din functia likelihood este maximizat pentru a afla coeficientii de regresie w:

$$\begin{aligned} \frac{1}{N} \log L(w|y, x) &= \frac{1}{N} \sum_{i=1}^{N} \log \mathrm{P}\left(y_i|x_i, w\right) \\ &= \frac{1}{N} \sum_{i=1}^{N} \left[y^{(i)} \log\left(h_w\left(x^{(i)}\right)\right) + \left(1 - y^{(i)}\right) \log\left(1 - h_w\left(x^{(i)}\right)\right)\right] \end{aligned} \tag{2}$$

Numeric, parametrii de regresie w se gasesc cu tehnci de optimizare, cum ar fi metoda gradient sau metoda Newton (cand numarul de date N este mic) si cu algoritmi stohastici (de ex: stohastic gradient), in contextul de big data. In cele ce urmeaza aplicam metodele gradient si Newton pentru gasirea parametrilor de regresie w, i.e. maximizarea functie likelihood.

3 Aplicatie: Caine sau pisica ?

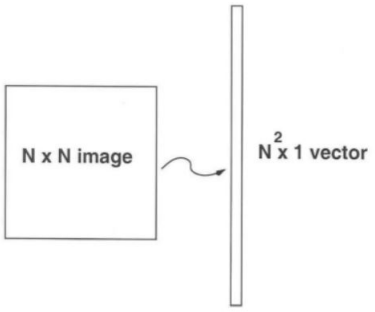
Scop: Gasirea unui clasificator binar folosind regresia logistica care sa clasifice daca intr-o poza se gaseste o pisica sau un caine.

3.1 Baza de date si preprocesare

Baza de date de antrenare [2] utilizata contine 105 poze cu pisici si 102 poze cu caini. Baza de test contine 26 poze cu pisici, respectiv 25 poze cu caini. Fiecare poza are asociata o eticheta, 1 pentru pisici si 0 pentru caine. Astfel vectorul $y \in \mathbb{R}^{207}$ este vectorul de etichete.

Figure 2: Imagini din setul de antrenare

Imaginile au diferite dimensiuni, motiv pentru care au fost redimensionate la 227×227. Dupa pozele au fost convertite in alb negru, transformate intr-un vector si salvate in variabila data. In final se obtine o matrice de dimensiune $X \in \mathbb{R}^{207 \times 51529}$. Pe aceasta



baza de date vom aplica o metoda de reducere a dimensiunii numita PCA (*engl. Principal component analysis* [3]) care sa retina cea mai importanta informatie. Vom reduce X la 45×207, adica $N = 207$ si $n = 45$. Fiecare imagine are o eticheta asociata: 1 pentru pisica si 0 pentru caine. Procedam in aceasi maniera si pentru baza de date de testare.

3.2 Metode de optimizare

Vom rescrie problema de maximizare a functiei likelihood ca o problema de minimizare folosind urmatoarea relatie:

$$\max_w F(w) = -\min_w -F(w).$$

Astfel rescriem 2 dupa cum urmeaza:

$$\begin{aligned} F(w) &= -\frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}\log\left(h_w\left(x^{(i)}\right)\right) + \left(1-y^{(i)}\right)\log\left(1-h_w\left(x^{(i)}\right)\right)\right] \\ &= \frac{1}{N}\cdot\left(-y^T\log(h) - (1-y)^T\log(1-h)\right), \end{aligned}$$

unde $y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \in \mathbb{R}^N$, $X = \begin{bmatrix} x_1^T \\ x_2^T \\ \cdots \\ x_N^T \end{bmatrix}^T \in \mathbb{R}^{n\times N}$ si $h = \frac{1}{1+e^{-w^TX}} = \begin{bmatrix} \frac{1}{1+e^{-x_1^Tw}} \\ \frac{1}{1+e^{-x_2^Tw}} \\ \vdots \\ \frac{1}{1+e^{-x_N^Tw}} \end{bmatrix} \in \mathbb{R}^N$.

Observatii:

- Înmulțirea cu y și $(1-y)$ în ecuția de mai sus este un truc care ne permite sa folosim aceeași ecuție pentru a rezolva atât cazurile $y = 1$ cat și pe cele pentru care $y = 0$. Dacă $y = 0$, prima parte se anulează. Dacă $y = 1$, a doua parte se anulează. În ambele cazuri efectuăm doar operația care trebuie.

Vom minimiza functia de cost utilizand metoda Newton. Vom alegem ca si criteriu de oprire norma gradientului sa fie mai mica decat un $\epsilon = 10^{-8}$. De asemenea, este o practica buna, sa impunem si un numar maxim de iteratii (maxIter = 10000).

1. Iteratia Metodei Newton:

$$w^{k+1} = w^k - \alpha^k\left(\nabla^2 F\left(w^k\right)\right)^{-1}\nabla F\left(w^k\right)$$

   - Pas constant: $\alpha^k = 1$
   - Pas ideal: $\alpha^k = \arg\min_{\alpha>0} F\left(w^k - \alpha\left(\nabla^2 F\left(w^k\right)\right)^{-1}\nabla F\left(w^k\right)\right)$.
     Hint: Folositi comanda fminbnd pentru a calcula pasul ideal.

Calculam mai departe gradientul si Hessiana lui F, vezi [1] pentru mai multe detalii:

- Derivata partiala a lui $F(w)$ in raport cu w_j:

$$\frac{\partial F(w)}{\partial w_j} = \frac{1}{N}\sum_{i=1}^{N}\left(h_w\left(x_i\right) - y_i\right)x_i^j \quad \Rightarrow \quad \nabla F(w) = \frac{1}{N}(h-y)^TX^T.$$

- Hessiana:

$$\begin{aligned} \nabla^2 F(w) &= \frac{1}{N}\sum_{i=1}^{N} h_w(x_i)\left(1-h_w(x_i)\right)x_ix_i^T = \frac{1}{N}\sum_{i=1}^{N}\frac{e^{-w^Tx_i}}{\left(1+e^{-w^Tx_i}\right)^2}x_ix_i^T \\ &= \frac{1}{N}XQ(w)X^T, \end{aligned}$$

  unde $Q(w) = \text{diag}([q_1(w), ..., q_N(w)])$ si $q_i(w) = h_w(x_i)(1-h_w(x_i))$.

3.3 Pragul de decizie

Funcția noastră actuală de predicție returnează un scor de probabilitate între 0 și 1. Pentru a face o mapare a acesteia într-o clasă discretă (pisica/caine), selectăm o valoare de prag peste care vom clasifica valorile în clasa promovat și sub care clasificăm valorile în clasa picat. Pentru exemplul nostru valoarea de prag va fi 0.5:

$$\begin{aligned} p \geq 0.5, &\quad \text{y} = 1 \\ p < 0.5, &\quad \text{y} = 0. \end{aligned}$$

3.4 Predictia

O funcție de predicție în regresia logistică întoarce probabilitatea observației noastre. Pe măsură ce probabilitatea se apropie de 1, modelul nostru este mai sigur că observația se află în clasa 1. Numim această clasă 1 și notația sa este P (clasa = 1):

$$P(\text{class} = 1) = \frac{1}{1 + e^{-w^T x}}.$$

4 Tema (3 puncte)

1. (3p) Implementati Metoda Newton atat pentru pas constant cat si pentru cel ideal. Creati graficul metoda Newton cu pas ideal versus pas constant, in care sa afisati evolutia criteriului, i.e norma gradientului.

(bonus) Testati clasificatorul calculat si construiti matricea de confuzie (Hint: Puteti folosi comanda *confusionmat*). Cat este acuratetea?

5 Anexa - codul Matlab

5.1 Functia Sigmoid

```
% Functia sigmoid = 1/(1+exp(-z))
% Outputul g este o valoare intre 0 si1
function g = sigmoid(z)
g = 1.0 ./ (1.0 + exp(-z));
```

References

[1] http://hua-zhou.github.io/teaching/biostatm280-2017spring/slides/18-newton/newton.html

[2] https://www.kaggle.com/competitions/dogs-vs-cats/code

[3] https://www.youtube.com/watch?v=FgakZw6K1QQ&ab_channel=StatQuestwithJoshStarmer