Regresie logistică

Noțiunea de *regresie* se referă la o categorie de metode statistice care au ca scop estimarea relațiilor dintre una sau mai multe variabile independente și una sau mai multe variebile dependente. Metodele de regresie estimează valoarea unei variabile dependente asociate valorii unei variabile independente sau valorilor unui set de variabile independente. Astfel, scopul regresiei este de a determina o funcție a variabilelor independente care furnizează valoarea dependentă corespunzătoare. Tehnicile prin care se realizează estimarea în cadrul regresiei depind în principal de tipul datelor de intrare/ieșire din setul de date de antrenare și de tipul funcției care se estimează.

Metodele de regresie logistică se aplică în situațiile în care variabila dependentă este, în cel mai simplu caz, binară (Da/Nu, Adevărat/Fals, 0/1 etc). În cazul regresiei logistice, se caută o funcție care, pentru o valoare arbitrară a variabilei independente, generează o valoare dependentă în intervalul (0, 1) care exprimă probabilitatea de încadrare în cele două cazuri descrise de valorile inițiale ale variabilei dependente.

Exemplu:

Următorul set de date conține numărul de ore dedicat studiului și valorile de adevăr ale afirmației “Am promovat examenul.”

Nr ore de studiu	**0.5**	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**1.75**	**2**	**2.25**	**2.5**	**2.75**	**3**	**3.25**	**3.5**	**4**	**4.25**	**4.5**	**4.75**	**5**	**5.5**
Promovare	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**

Dorim să determinăm o funcție care să permită determinarea probabilității de promovare a examenului pentru orice valoare a numărului de ore de studiu. Această funcție are expresia din Ec. (1). Pentru anumite valori ale parametrilor w_0 și w_1, se obține graficul din Fig. 1, unde se reprezintă și datele de antrenare.

$$\hat{y} = \frac{1}{1 + e^{-(w_0 + w_1 x)}} \tag{1}$$

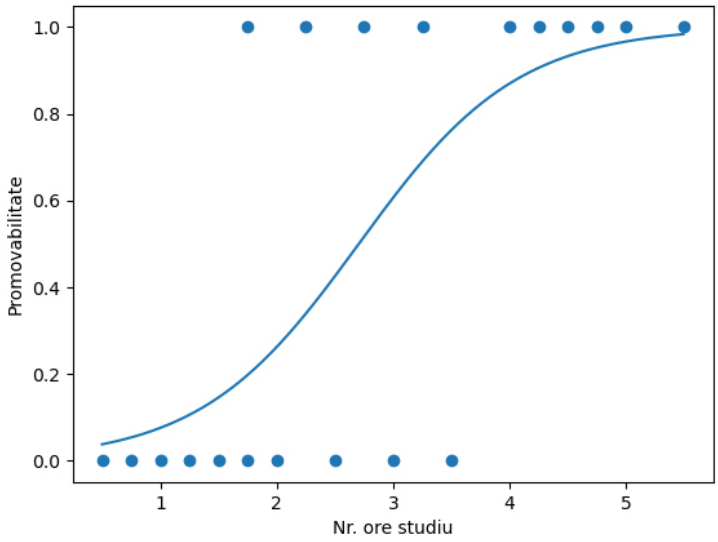
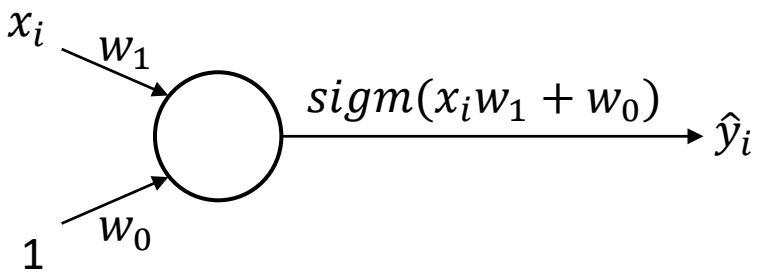


Fig. 1. Funcția care rezultă prin aplicarea regresiei logistice pe datele de antrenare, pentru w_0 = -3.966, w_1 = 1.467

Scopul este de a determina valorile coeficienților w_0 și w_1, astfel încât funcția din Ec. (1) să genereze erori cât mai mici pentru datele de antrenare. Dorim o pereche de valori ale coeficienților care să minimizeze diferențele dintre probabilitățile obținute cu funcția din Ec. (1) și valorile de adevăr din setul de date, pentru toate valorile variabilei independente (ale numărului de ore de studiu). Spre deosebire de cazul regresiei liniare, în situația dată nu există o formulă analitică de calcul a acestor coeficienți, valorile lor trebuind estimate printr-o metodă de optimizare.

Definim un perceptron de forma:



Unde x_i sunt valorile variabilei independente (orele de studiu), w_0 și w_1 sunt ponderile intrărilor perceptronului, iar funcția de activare este funcția sigmoid (Ec. (2)).

$$sigm(x) = \frac{1}{1+e^{-x}} \tag{2}$$

Antrenarea perceptronului se realizează astfel:

Pentru determinarea erorii, vom folosi funcția *cross entropy* (Ec. (3)). Pe parcursul mai multor iterații, vom determina valoarea acestei funcții (eroarea de la fiecare iterație), vom determina gradienții erorii în raport cu cele două ponderi w_0 și w_1 (Ec. (4)) și vom actualiza ponderile conform cu Ec. (5), în ideea de a reduce eroarea la fiecare iterație.

$$CE = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \hat{y}_i + (1-y_i)\log\left(1-\hat{y}_i\right) \tag{3}$$

$$\begin{aligned} \frac{\partial CE}{\partial w_0} &= \sum_{i=1}^{N} \hat{y}_i - y_i \\ \frac{\partial CE}{\partial w_1} &= \sum_{i=1}^{N} \left(\hat{y}_i - y_i\right) x_i \end{aligned} \tag{4}$$

...unde

x_i sunt valorile variabilei independente (nr. ore studiu)

y_i sunt valorile variabilei dependente (valorile de 0 sau 1 ale promovării)

$\hat{y}_i$ sunt valorile predicțiilor care se obțin pentru x_i (Ec. (1))

- se inițializează ponderile w_0 și w_1 cu valori aleatoare dintr-un interval restrâns, de ex [-0.5, 0.5]
- se stabilește un număr de iterații (“epoci”) (o valoare mare, de ex. 1000). În cadrul fiecărei epoci are loc o etapă de calcul a ieșirii $\hat{y}_i$ și de reglare a ponderilor.
- pentru fiecare epocă:
    - se determină eroarea (Ec. (3))
    - se determină valoarea de la ieșirea perceptronului
    - se determină componentele gradientului (Ec. (4))
    - se actualizează ponderile (Ec. (5)). Rata de învățare α are de obicei o valoare subunitară mică (se poate încerca inițial α = 0.01 și apoi, dacă este cazul, se ajustează acest parametru în ideea de a obține o eroare cât mai mică)
    - se determină din nou eroarea și se verifică dacă aceasta se apropie suficient de mult de valoarea determinată la început (i.e. dacă modulul diferenței erorilor este mai mic decât o constantă subunitară ε de valoare mică (de ex. 0.001)).

Dacă erorile sunt suficient de apropiate ca valoare, se consideră algoritmul încheiat și se consideră corecte valorile curente ale w_0 și w_1.

$$w_0 = w_0 - \alpha \frac{\partial CE}{\partial w_0}$$
$$w_1 = w_1 - \alpha \frac{\partial CE}{\partial w_1} \quad (5)$$

$$MSE = \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \hat{y}_i \right)^2 \quad (6)$$

Aplicații:

1. Implementați regresia logistică pentru datele din exemplu:
   - Determinați cei doi coeficienții și funcția obținută (pentru verificare, un set de valori utile este w_0 = -3.966, w_1 = 1.467)
     - Pentru afișarea graficelor în Python, vezi laboratorul anterior
   - Determinați probabilitatea de promovare a examenului pentru o valoare oarecare a numărului de ore de studiu
   - Afisați un grafic care să ilustreze evoluția erorii pe parcursul epocilor.

2. Implementați regresia logistică folosind ca eroare funcția MSE (Ec. (6)). Determinați gradienții aproximându-i prin diferențe finite (vezi laboratorul anterior). Ajustați parametrii din faza de antrenare astfel încât să obțineți valori cât mai apropiate de cele de la punctul 1).