Regressão Logística

10 de julho de 2023

1 O que é regressão Logística?

A regressão logística é um método estatístico usado para prever uma variável dependente binária, ou seja, uma variável que tem apenas duas possíveis categorias de saída, com base em uma ou mais variáveis independentes.

Por exemplo, suponha que você esteja tentando prever se um e-mail é spam (1) ou não-spam (0), com base na frequência de certas palavras. Nesse caso, a variável dependente é binária (spam ou não-spam) e as variáveis independentes são as frequências das palavras.

Vamos começar com a regressão linear. Em regressão linear, você pode fazer uma previsão simplesmente tomando uma combinação linear das características. Em notação matemática, isso pode ser escrito como:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n \tag{1}$$

Onde:

- y é a variável dependente (a saída que você está tentando prever),
- $x_1, x_2, ..., x_n$ são as variáveis independentes,
- $\beta_0, \beta_1, ..., \beta_n$ são os parâmetros do modelo que você precisa estimar

No entanto, para a regressão logística, queremos prever uma probabilidade que deve estar entre 0 e 1. Para garantir que isso aconteça, passamos a combinação linear das características através da função logística (ou função sigmóide), que mapeia qualquer número real para o intervalo (0, 1). A equação de regressão logística, portanto, se torna:

$$p(Y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n)}} \tag{2}$$

Isso nos dá a probabilidade de Y ser igual a 1, dadas as variáveis independentes. Onde e é a base dos logaritmos naturais.

Para treinar o modelo (ou seja, estimar os parâmetros β), usamos o método da máxima verossimilhança. A função de verossimilhança, neste caso, é:

$$L(\beta) = \prod_{i=1}^{n} [p(y_i = 1)]^{y_i} [1 - p(y_i = 1)]^{1 - y_i} \tag{3}$$

Contudo, esta função objetivo pode sobre com problemas numéricos (*underflow*). Assim, é mais comum utilizar a log-verossimilhança:

$$\log L(\beta) = \sum_{i=1}^{n} y_i \log[p(y_i = 1)] + (1 - y_i) \log[1 - p(y_i = 1)] \tag{4}$$

No caso da regressão logística podemos escrever a log-verossimilhança, assim:

$$\log L(\beta) = \sum_{i=1}^{n} y_i \log\left(\frac{1}{1+e^{-(\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}+\ldots+\beta_n x_{in})}}\right)+(1-y_i)\log\left(1-\frac{1}{1+e^{-(\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}+\ldots+\beta_n x_{in})}}\right) \tag{5}$$

2 Por quê utilizamos a verossimilhança e não o erro quadrado médio?

Ótima pergunta! Em muitos modelos estatísticos, usamos a minimização do erro quadrado médio (MSE, do inglês "Mean Squared Error") como critério de aprendizado. No entanto, na regressão logística, usamos a maximização da verossimilhança. Aqui estão algumas razões para isso:

1. A natureza da variável dependente: Na regressão logística, a variável dependente é binária, e a função sigmoide garante que a saída do modelo está entre 0 e 1, que podemos tratar como a probabilidade de a classe ser 1. O método de máxima verossimilhança é apropriado quando estamos modelando probabilidades.

2. A não-normalidade dos resíduos: Uma das suposições da regressão linear (que usa o MSE) é que os erros (ou resíduos) são normalmente distribuídos. No entanto, na regressão logística, essa suposição não é válida, pois estamos modelando probabilidades e a saída é binária.

3. Robustez a outliers: O uso do MSE pode ser muito sensível a outliers, pois os erros são elevados ao quadrado. Isso pode distorcer o modelo se houver outliers. A maximização da verossimilhança é menos sensível a outliers.

4. Interpretação probabilística: A maximização da verossimilhança tem uma interpretação probabilística clara: estamos escolhendo os parâmetros que maximizam a probabilidade dos dados observados.

Assim, embora o MSE seja muito útil em muitos cenários, especialmente na regressão linear, não é o critério de aprendizado mais apropriado para a regressão logística.

3 Por quê utilizamos a log-verossimilhança no lugar da verossimilhança?

Usar a log-verossimilhança em vez da verossimilhança diretamente tem várias vantagens:

1. Operação de soma em vez de produto: A log-verossimilhança transforma a operação de produto da verossimilhança em uma soma, pois o logaritmo do produto é a soma dos logaritmos. Isto é matematicamente mais conveniente para manipulação e simplifica os cálculos.

2. Estabilidade numérica: Quando trabalhamos com uma grande quantidade de dados, a verossimilhança pode se tornar extremamente pequena, ao ponto de que pode haver underflow (quando números muito pequenos são arredondados para zero). Tirar o logaritmo de uma pequena probabilidade dá um número negativo, o que evita esse problema de underflow.

4 Como encontramos os parâmetros?

A regressão logística é comumente formulada como um problema de otimização, onde buscamos maximizar a função de log-verossimilhança.

$$\hat{\beta} = \underset{\beta}{\operatorname{argmax}} \{\log L(\beta)\} \tag{6}$$

Aqui, $\hat{\beta}$ é a estimativa dos parâmetros do modelo que maximizam a função de log-verossimilhança, $\log L(\beta)$. Lembrando que nossa função de log-verossimilhança é:

$$\log L(\beta) = \sum_{i=1}^{n} y_i \log\left(\frac{1}{1+e^{-(\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}+\ldots+\beta_n x_{in})}}\right) + (1-y_i)\log\left(1-\frac{1}{1+e^{-(\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}+\ldots+\beta_n x_{in})}}\right) \tag{7}$$

Na prática, os parâmetros são frequentemente encontrados através de um algoritmo iterativo, como o gradiente ascendente. Neste caso, precisamos calcular as derivadas parciais da log-verossimilhança em relação aos pesos.

4.1 Cálculo da derivadas parciais

Por conveniência, vamos definir $p_i = \frac{1}{1+e^{-(\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}+\ldots+\beta_n x_{in})}}$. A derivada de $\log p_i$ em relação a β_j é dada por:

$$\frac{\partial \log p_i}{\partial \beta_j} = \frac{1}{p_i}\frac{\partial p_i}{\partial \beta_j} = \frac{1}{p_i} p_i(1-p_i)x_{ij} = (1-p_i)x_{ij} \tag{8}$$

onde usamos o fato de que a derivada de p_i em relação a β_j é $p_i(1-p_i)x_{ij}$. Agora, vamos calcular a derivada de $\log(1-p_i)$ em relação a β_j:

$$\frac{\partial \log(1-p_i)}{\partial \beta_j} = \frac{1}{1-p_i}\frac{\partial (1-p_i)}{\partial \beta_j} = \frac{1}{1-p_i}(-p_i)p_i x_{ij} = -p_i x_{ij} \tag{9}$$

Combinando essas duas partes, podemos calcular a derivada parcial da log-verossimilhança em relação a β_j:

$$\frac{\partial \log L(\beta)}{\partial \beta_j} = \sum_{i=1}^{n} y_i \frac{\partial \log p_i}{\partial \beta_j} + (1-y_i)\frac{\partial \log(1-p_i)}{\partial \beta_j} = \sum_{i=1}^{n} y_i(1-p_i)x_{ij} - (1-y_i)p_i x_{ij} = \sum_{i=1}^{n}(y_i - p_i)x_{ij} \tag{10}$$

Isto é:

$$\frac{\partial \log L(\beta)}{\partial \beta_j} = \sum_{i=1}^{n}\left(y_i - \frac{1}{1+e^{-(\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}+\ldots+\beta_n x_{in})}}\right) x_{ij} \tag{11}$$

4.2 Algoritmo do gradiente ascendente

Algorithm 1 Algoritmo de Subida do Gradiente

1: Inicialize os parâmetros β com algum valor inicial.
2: Fixe a taxa de aprendizado η e o critério de parada ϵ.
3: **repeat**
4: Calcule o gradiente da função objetivo:

$$\nabla \log L(\beta) = \sum_{i=1}^{n} (y_i - p_i)\mathbf{x}_i$$

5: Atualize os parâmetros:

$$\beta \leftarrow \beta + \eta \nabla \log L(\beta)$$

6: **until** $||\nabla \log L(\beta)|| < \epsilon$
7: **return** β

5 Exemplo

Vamos criar um exemplo numérico simples com três observações e um recurso, além do termo de interceptação. Nosso objetivo é prever se um email é spam (1) ou não (0) com base na frequência de uma determinada palavra no email.

Aqui estão os dados:

Frequência da palavra (x_1)	Spam (y)
0.1	0
0.8	1
0.3	0

Tabela 1: Dados de exemplo

Vamos inicializar nosso vetor de parâmetros β com [0, 0]. A primeira entrada é o termo de interceptação e a segunda entrada corresponde ao recurso "Frequência da palavra". Além disso, definimos a taxa de aprendizado $\eta = 0.1$.

Então, em cada etapa do algoritmo de subida do gradiente, calculamos o gradiente e atualizamos β de acordo com as fórmulas que fornecemos anteriormente.

Por exemplo, na primeira etapa, temos que calcular p_i para cada observação, usando a fórmula $p_i = \frac{1}{1+e^{-(\beta_0+\beta_1 x_{i1})}}$. Como $\beta = [0, 0]$, temos que $p_i = 0.5$ para todas as observações.

Em seguida, calculamos o gradiente. Para a interceptação, temos:

$$p_i = \frac{1}{1 + e^{-(\beta_0+\beta_1 x_{i1})}} \tag{12}$$

$$\frac{\partial \log L(\beta)}{\partial \beta_0} = \sum_{i=1}^{n} (y_i - p_i) \tag{13}$$

Para a frequência da palavra, temos:

$$\frac{\partial \log L(\beta)}{\partial \beta_1} = \sum_{i=1}^{n} (y_i - p_i) x_{i1} \tag{14}$$

Portanto, o gradiente é [-1, 0.15].

Atualizamos β subtraindo a taxa de aprendizado vezes o gradiente de β:

$$\beta \leftarrow \beta + \eta \nabla \log L(\beta) \tag{15}$$

E repetimos esse processo até atingir o critério de parada.

Esse é um exemplo muito simplificado, e na prática você teria muitos mais dados e recursos, e provavelmente usaria uma versão estocástica ou em mini-lotes do gradiente ascendente, mas espero que isso dê uma ideia geral do processo.

6 Exercícios

1. Considere um único ponto de dados com um único recurso. Seu objetivo é prever se um email é spam (1) ou não (0) com base na frequência de uma palavra. Se a frequência da palavra for 0.3 e o email não for spam, qual é o valor da função log-verossimilhança se $\beta = [0, 0]$? Use a fórmula para a função log-verossimilhança dada anteriormente.

2. Usando o mesmo ponto de dados do Exercício 1, qual é o gradiente da função log-verossimilhança se $\beta = [0, 0]$? Use as fórmulas para as derivadas parciais dadas anteriormente.

3. Ainda referindo-se ao ponto de dados do Exercício 1, se você usar a subida do gradiente com uma taxa de aprendizado de 0.1 para atualizar β, qual será o novo valor de β?

4. Agora considere um segundo ponto de dados com frequência de palavra 0.7 e o email é spam. Calcule o valor da função log-verossimilhança e seu gradiente para esses dois pontos de dados se $\beta = [0, 0]$.

5. Ainda usando os dois pontos de dados do Exercício 4, se você usar a subida do gradiente com uma taxa de aprendizado de 0.1 para atualizar β, qual será o novo valor de β?

6. Considere agora três pontos de dados: (0.3, 0), (0.7, 1), (0.1, 0), onde o primeiro número de cada par é a frequência da palavra e o segundo número indica se o email é spam. Calcule o valor da função log-verossimilhança e seu gradiente para esses três pontos de dados se $\beta = [0, 0]$. Em seguida, use a subida do gradiente com uma taxa de aprendizado de 0.1 para atualizar β. Qual é o novo valor de β?

7 Soluções

1. **Exercício 1:**

   Para calcular a função log-verossimilhança, primeiro precisamos calcular a probabilidade prevista $p = \frac{1}{1+e^{-(\beta_0+\beta_1 x)}} = \frac{1}{1+e^{-(0+0*0.3)}} = 0.5$.

   Então a log-verossimilhança é $y\log(p)+(1-y)\log(1-p) = 0\log(0.5)+(1-0)\log(1-0.5) = -\log(2)$.

$$p = \frac{1}{1+e^{-(\beta_0+\beta_1 x)}} = \frac{1}{1+e^{-(0+0\cdot 0.3)}} = 0.5$$
$$L(\beta) = y\log(p) + (1-y)\log(1-p) = 0\log(0.5) + (1-0)\log(1-0.5) = -\log(2)$$

2. **Exercício 2:**

   Para calcular o gradiente da função log-verossimilhança, precisamos calcular suas derivadas parciais em relação a β_0 e β_1.

   Usando as fórmulas fornecidas, temos $\frac{\partial \log L(\beta)}{\partial \beta_0} = y - p = 0 - 0.5 = -0.5$ e $\frac{\partial \log L(\beta)}{\partial \beta_1} = (y-p)x = (0-0.5)0.3 = -0.15$. Portanto, o gradiente é $[-0.5, -0.15]$.

$$\frac{\partial \log L(\beta)}{\partial \beta_0} = y - p = 0 - 0.5 = -0.5$$
$$\frac{\partial \log L(\beta)}{\partial \beta_1} = (y-p)x = (0-0.5)0.3 = -0.15$$

3. **Exercício 3:**

   Para atualizar β, subtraímos a taxa de aprendizado vezes o gradiente de β. Portanto, $\beta \leftarrow \beta - \eta\nabla \log L(\beta) = [0,0] - 0.1 * [-0.5, -0.15] = [0.05, 0.015]$.

$$\beta \leftarrow \beta - \eta\nabla \log L(\beta) = [0,0] - 0.1 \cdot [-0.5, -0.15] = [0.05, 0.015]$$

4. **Exercício 4:**

   Para dois pontos de dados, calculamos a probabilidade prevista e a log-verossimilhança para cada ponto e somamos os resultados.

   Para o primeiro ponto de dados (0.3, 0), temos $p = 0.5$ e a log-verossimilhança é $-\log(2)$, como calculamos no Exercício 1.

   Para o segundo ponto de dados (0.7, 1), também temos $p = 0.5$ e a log-verossimilhança é $\log(0.5) = -\log(2)$.

   Portanto, a log-verossimilhança para os dois pontos de dados é $-\log(2) - \log(2) = -2\log(2)$.

   O gradiente também é a soma dos gradientes para cada ponto de dados. Portanto, o gradiente para β_0 é $-0.5 - 0.5 = -1$ e o gradiente para β_1 é $-0.15 - 0.35 = -0.5$. O gradiente total é $[-1, -0.5]$.

$$
\begin{aligned}
L(\beta) &= -\log(2) - \log(2) = -2\log(2) \\
\frac{\partial \log L(\beta)}{\partial \beta_0} &= -0.5 - 0.5 = -1 \\
\frac{\partial \log L(\beta)}{\partial \beta_1} &= -0.15 - 0.35 = -0.5
\end{aligned}
$$

5. **Exercício 5:**

   Para atualizar β, subtraímos a taxa de aprendizado vezes o gradiente de β. Portanto, $\beta \leftarrow \beta - \eta \nabla \log L(\beta) = [0, 0] - 0.1 * [-1, -0.5] = [0.1, 0.05]$.

$$\beta \leftarrow \beta - \eta \nabla \log L(\beta) = [0, 0] - 0.1 \cdot [-1, -0.5] = [0.1, 0.05]$$

6. **Exercício 6:**

   Para três pontos de dados, procedemos da mesma forma que no Exercício 4, somando as log-verossimilhanças e gradientes para cada ponto de dados.

   A log-verossimilhança para os três pontos de dados é $-\log(2) - \log(2) - \log(2) = -3\log(2)$.

   O gradiente para β_0 é $-0.5 - 0.5 - 0.5 = -1.5$ e o gradiente para β_1 é $-0.15 - 0.35 - 0.05 = -0.55$. O gradiente total é $[-1.5, -0.55]$.

   Para atualizar β, temos $\beta \leftarrow \beta - \eta \nabla \log L(\beta) = [0, 0] - 0.1 * [-1.5, -0.55] = [0.15, 0.055]$.

$$
\begin{aligned}
L(\beta) &= -\log(2) - \log(2) - \log(2) = -3\log(2) \\
\frac{\partial \log L(\beta)}{\partial \beta_0} &= -0.5 - 0.5 - 0.5 = -1.5 \\
\frac{\partial \log L(\beta)}{\partial \beta_1} &= -0.15 - 0.35 - 0.05 = -0.55 \\
\beta \leftarrow \beta - \eta \nabla \log L(\beta) &= [0, 0] - 0.1 \cdot [-1.5, -0.55] = [0.15, 0.055]
\end{aligned}
$$