Resolução do Exercício 2: Verossimilhança, Entropia Cruzada e Regressão Logística

(a)

Para a Regressão Logística com alvos $t_n \in \{0, 1\}$, a probabilidade de observar um alvo t_n dado o input $\boldsymbol{\phi}_n$ e os pesos $\mathbf{w}$ é modelada como uma distribuição de Bernoulli com parâmetro

$$y_n = \sigma(\mathbf{w}^T \boldsymbol{\phi}_n) = P(t_n = 1 | \boldsymbol{\phi}_n; \mathbf{w}).$$

Assim, a função de verossimilhança (likelihood) para um conjunto de N observações independentes $\{\boldsymbol{\phi}_n, t_n\}_{n=1}^N$ é dada por:

$$p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^{N} p(t_n|\boldsymbol{\phi}_n; \mathbf{w}) = \prod_{n=1}^{N} y_n^{t_n}(1 - y_n)^{1-t_n},$$

onde $\mathbf{t} = (t_1, \ldots, t_N)^T$. Para facilitar a otimização, geralmente trabalhamos com o logaritmo natural da verossimilhança (log-likelihood):

$$\ln p(\mathbf{t}|\mathbf{w}) = \sum_{n=1}^{N} \ln \left(y_n^{t_n}(1 - y_n)^{1-t_n} \right).$$

Usando as propriedades do logaritmo:

$$\ln p(\mathbf{t}|\mathbf{w}) = \sum_{n=1}^{N} \left[t_n \ln y_n + (1 - t_n) \ln(1 - y_n) \right].$$

A função de erro de entropia cruzada é definida como:

$$E(\mathbf{w}) = -\sum_{n=1}^{N} \left[t_n \ln y_n + (1 - t_n) \ln(1 - y_n) \right].$$

Comparando as duas equações, vemos que $E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{w})$. Portanto, maximizar a verossimilhança é equivalente a minimizar a entropia cruzada.

(b)

Lembrando que $y_n = \sigma(\mathbf{a}_n)$, com $\mathbf{a}_n = \mathbf{w}^T \boldsymbol{\phi}_n$, e que

$$\frac{d\sigma(a)}{da} = \sigma(a)(1 - \sigma(a)),$$

a função de erro é:

$$E(\mathbf{w}) = -\sum_{n=1}^{N} \left[t_n \ln \sigma(\mathbf{w}^T \boldsymbol{\phi}_n) + (1 - t_n) \ln(1 - \sigma(\mathbf{w}^T \boldsymbol{\phi}_n)) \right].$$

O gradiente de $E(\mathbf{w})$ é:

$$\begin{aligned}
\nabla E(\mathbf{w}) &= -\sum_{n=1}^{N} \left[t_n \frac{1}{y_n} y_n(1 - y_n)\boldsymbol{\phi}_n + (1 - t_n) \frac{1}{1 - y_n} (-y_n(1 - y_n))\boldsymbol{\phi}_n \right] \\
&= -\sum_{n=1}^{N} \left[t_n(1 - y_n) - (1 - t_n)y_n \right] \boldsymbol{\phi}_n \\
&= \sum_{n=1}^{N} (y_n - t_n)\boldsymbol{\phi}_n.
\end{aligned}$$

Essa forma do gradiente é conveniente para métodos de otimização como o gradiente descendente:

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} - \eta \nabla E(\mathbf{w}^{(k)}),$$

com η sendo a taxa de aprendizado. Isso também é útil para métodos estocásticos como o SGD.

(c)

O overfitting ocorre quando os dados são linearmente separáveis. Nesse caso, existe um $\mathbf{w}$ tal que $\mathbf{w}^T \boldsymbol{\phi}_n$ separa perfeitamente as classes, fazendo com que $\|\mathbf{w}\| \to \infty$.

A função sigmoide se torna uma função degrau de Heaviside, e o modelo se torna excessivamente confiante, generalizando mal.

Para evitar isso, usa-se regularização L_2:

$$E_{\text{regularizado}}(\mathbf{w}) = -\sum_{n=1}^{N} \left[t_n \ln y_n + (1 - t_n) \ln(1 - y_n) \right] + \frac{\lambda}{2} \|\mathbf{w}\|^2,$$

onde $\lambda > 0$ controla a força da regularização. O termo $\frac{\lambda}{2}\|\mathbf{w}\|^2$ penaliza pesos grandes, resultando em fronteiras de decisão mais suaves e melhores propriedades de generalização.

Nota: O NotebookLM pode gerar respostas incorretas. Verifique sempre o conteúdo.