IA353 - Redes Neurais

* Técnicas de otimização não-linear irrestrita

Neste material, vamos cobrir de maneira sucinta os aspectos básicos de diferentes técnicas de otimização não-linear que podem ser utilizadas no treinamento de redes neurais do tipo MLP.

⊛ Relação entre gradiente e hessiana

- o método de Newton explora a aproximação local da função de erro $J(\theta)$ em série de Taylor até o termo de 2ª ordem.

$$J_{quad}(\theta) = J(\theta_i) + \nabla J(\theta_i)^T \cdot (\theta - \theta_i) + \frac{1}{2}[\theta - \theta_i]^T \nabla^2 J(\theta_i)[\theta - \theta_i]$$

Qual $\Delta\theta = \theta_{i+1} - \theta_i$ nos leva a um ponto ótimo da aproximação quadrática?

$$\frac{\partial J_{quad}(\theta_{i+1})}{\partial \theta_{i+1}} = 0 \quad \text{(condição de otimalidade do ponto } \theta_{i+1}\text{)}$$

$$\underbrace{\cancel{\frac{\partial J(\theta_i)}{\partial \theta_{i+1}}}}_{0} + \nabla J(\theta_i) + \nabla^2 J(\theta_i)[\theta_{i+1} - \theta_i] = 0$$

$$\Rightarrow \theta_{i+1} - \theta_i = \Delta\theta = -(\nabla^2 J(\theta_i))^{-1} \nabla J(\theta_i)$$

Se $J(\theta)$ não for uma função quadrática, $J(\theta_{i+1})$ não necessariamente será menor que $J(\theta_i)$. Por isso, a lei de ajuste passa a incluir um parâmetro α_i p/ controlar o tamanho do passo:

$$\theta_{i+1} = \theta_i - \alpha_i [\nabla^2 J(\theta_i)]^{-1} \nabla J(\theta_i)$$

PROBLEMAS: positividade da matriz hessiana ⇒ p/ haver inversa e p/ buscarmos um ponto de mínimo

cálculo exato da matriz hessiana e de sua inversa

O vetor gradiente também pode ser aproximado por uma série de Taylor:

$$\nabla J(\theta_{i+1}) = \nabla J(\theta_i + p_i) = \nabla J(\theta_i + \alpha_i d_i)$$

↳ direção: $-\nabla^2 J(\theta)^{-1} \nabla J(\theta)$ — NEWTON

$$= \nabla J(\theta_i) + \nabla^2 J(\theta_i)[\theta_{i+1} - \theta_i]$$

$$= \nabla J(\theta_i) + \nabla^2 J(\theta_i) \cdot p_i$$

⇒ Seja $g_{i+1} = \nabla J(\theta_{i+1})$ e $g_i = \nabla J(\theta_i)$. A diferença dos vetores gradientes acaba trazendo informações sobre a matriz hessiana:

$$g_{i+1} - g_i = \nabla^2 J(\theta_i) \cdot p_i$$

Esta relação será explorada nos métodos de otimização quasi-Newton, como os algoritmos DFP e BFGS.

⊛ Método de Levenberg-Marquardt

• pode ser visto como uma solução de compromisso entre o método de Newton – que converge rapidamente próximo de um mínimo local ou global – e o método do gradiente – que tem convergência assegurada com uma escolha apropriada do tamanho do passo, mas que pode ser lenta.

Idéia: $\Delta\theta = [H + \lambda I]^{-1} \cdot g$ → vetor gradiente: $\nabla J(\theta)$

↳ matriz hessiana: $\nabla^2 J(\theta)$

λ é um parâmetro de regularização que força a matriz $H + \lambda I$ a ser positiva definida.

Outra diferença deste método está associada à forma como a matriz hessiana é calculada. Considere a função de erro $J(\theta) = \sum_{i=1}^{N} \sum_{j=1}^{n} (d_j(i) - y_j(i))^2 = \sum_{i=1}^{N.n} e_i^2$, onde e_i é o erro residual (oriunde 1 para cada saída da rede em cada instante de tempo ou p/ cada padrão de entrada).

$$(1) \quad \frac{\partial J(\theta)}{\partial \theta} = \sum_{i=1}^{N.n} 2 e_i \cdot \underbrace{\frac{\partial e_i}{\partial \theta}}_{\text{vetor}}$$

Seja $J \in \mathbb{R}^{N.n \times N_{pesos}}$ o jacobiano do funcional $J(\theta)$

$$J = \begin{bmatrix} \nabla e_1^T \\ \vdots \\ \nabla e_N^T \end{bmatrix} = \begin{bmatrix} \frac{\partial e_1}{\partial \theta_1} & \cdots & \frac{\partial e_1}{\partial \theta_{N_{pesos}}} \\ & & \\ \frac{\partial e_{N.n}}{\partial \theta_1} & \cdots & \frac{\partial e_{N.n}}{\partial \theta_{N_{pesos}}} \end{bmatrix}$$

— todas as derivadas do erro referente ao 1º padrão c/ respeito a todos os N_{pesos} pesos da rede.

— todas as derivadas do erro referente ao N-ésimo padrão (e n-ésima saída), c/ respeito a todos os N_{pesos} pesos da rede

Então, podemos escrever que $\frac{\partial J(\theta)}{\partial \theta} = 2 J^T \cdot \underline{e}$, onde $\underline{e} = [e_1 \cdots e_{N.n}]^T$.

$$(2) \quad \frac{\partial^2 J(\theta)}{\partial \theta} = \sum_{i=1}^{N.n} 2 \cdot \frac{\partial e_i}{\partial \theta} \cdot \frac{\partial e_i}{\partial \theta}^T + 2 \sum_{i=1}^{N.n} e_i \cdot \frac{\partial^2 e_i}{\partial \theta^2}$$

$$= 2 \cdot J^T J + 2 \sum_{i=1}^{N.n} e_i \cdot \nabla^2 e_i$$

Quando os erros residuais são suficientemente pequenos, a matriz hessiana pode ser aproximada pelo primeiro termo:

$$\frac{\partial^2 J(\theta)}{\partial \theta} = \nabla^2 J(\theta) \approx 2 J^T J \rightarrow$$ estimativa usando apenas as derivadas de 1ª ordem

O parâmetro λ desempenha um papel bastante importante no funcionamento do algoritmo:

- λ muito pequeno $\Rightarrow$ nos aproximamos do método de Newton
- λ muito grande $\Rightarrow$ a informação da hessiana tem pouco efeito e o comportamento se assemelha ao do método do gradiente

PROCEDIMENTO SUGERIDO:

(i) compute $J(\theta_{i-1})$

(ii) escolha um valor modesto para λ, e.g., $\lambda = 10^{-3}$

(iii) obtenha $\Delta\theta = (H + \lambda I)^{-1} g$ e calcule $J(\theta_{i-1} + \Delta\theta)$

(iv) se $J(\theta_i) \geq J(\theta_{i-1})$ – ou seja, a atualização piora o desempenho da rede –, aumente λ por um fator (e.g., 10) e retorne ao passo (iii).

(v) se $J(\theta_i) < J(\theta_{i-1})$, reduza λ por um fator (e.g., 10), efetive a alteração nos pesos ($\theta_i = \theta_{i-1} + \Delta\theta$) e avance p/ a próxima iteração

⊛ Métodos Quasi-Newton: DFP e BFGS

- trabalham com uma aproximação iterativa da inversa da matriz hessiana, de forma que: $\lim_{i \to \infty} H_i = \nabla^2 J(\theta)^{-1}$
- em cada iteração, a inversa da hessiana é aproximada pela soma de duas matrizes simétricas de rank 1, procedimento que é chamado de correção de posto 2.

DFP: $$H_{i+1} = H_i + \frac{p_i p_i^T}{p_i^T q_i} - \frac{H_i q_i q_i^T H_i}{q_i^T H_i q_i}, \quad i = 0, \dots, N_{pesos} - 1,$$

onde $p_i = \alpha_i d_i$ e $q_i = g_{i+1} - g_i$.

Inicialmente, a direção de minimização é dada pelo vetor gradiente e a matriz hessiana corresponde a uma matriz identidade. A cada N_{pesos} iterações, o algoritmo é reinicializado.

BFGS: $$H_{i+1} = H_i + \frac{p_i p_i^T}{p_i^T q_i}\left[1 + \frac{q_i^T H_i q_i}{p_i^T q_i}\right] - \frac{H_i q_i p_i^T + p_i q_i^T H_i}{p_i^T q_i}$$

Em ambos os casos, o tamanho do passo α_i em cada iteração é determinado

com o auxílio de uma busca unidimensional.

⊛ <u>Métodos baseados no gradiente conjugado</u>

• são métodos projetados para exigir menos cálculos que o método de Newton e apresentar taxas de convergência maiores que as do método do gradiente. A regra básica de atualização $\theta_{i+1} = \theta_i + \alpha_i d_i$ nos permite escrever que a solução ótima θ^*, uma vez obtida, é dada por:

$$\theta^* = \alpha_0 d_0 + \alpha_1 d_1 + \dots = \sum_i \alpha_i d_i$$

Como θ^* é um ponto no espaço $\mathbb{R}^{N_{pesos}}$, ele pode ser representado como uma combinação linear de N_{pesos} vetores que formam uma base:

$$\theta^* = \alpha_0 d_0 + \dots + \alpha_{N_{pesos}-1} \cdot d_{N_{pesos}-1} = \sum_{i=0}^{N_{pesos}-1} \alpha_i d_i , \qquad (A)$$

desde que o conjunto $\{d_0, \dots, d_{N_{pesos}-1}\}$ forme uma base de $\mathbb{R}^{N_{pesos}}$ e $\alpha = [\alpha_0 \dots \alpha_{N_{pesos}-1}]^T$ seja a representação de θ^* nesta base.

O princípio de operação do método do gradiente conjugado para obter θ^* em até N_{pesos} iterações consiste em buscar vetores $d_i \in \mathbb{R}^{N_{pesos}}$, $i = 0, \dots, N_{pesos}-1$, linearmente independentes e encontrar a representação de θ^* nesta base, ou seja, os coeficientes α_i.

<u>Definição</u>: dada uma matriz simétrica A ($N_{pesos} \times N_{pesos}$), as direções $d_i \in \mathbb{R}^{N_{pesos}}$ são A-conjugadas (ou A-ortogonais) se

$$d_j^T A d_i = 0, \quad i \neq j$$

- note que esta definição é uma generalização do conceito de ortogonalidade entre vetores (que ocorre p/ $A = I$).

- se A for simétrica e definida positiva, as direções A-conjugadas são necessariamente linearmente independentes. Consequentemente, tal conjunto $\{d_0, \dots, d_{N_{pesos}-1}\}$ forma uma base do $\mathbb{R}^{N_{pesos}}$.

• multiplicando (A) à esquerda por $d_j^T A$, obtemos

$$d_j^T A \theta^* = \sum_{i=0}^{N_{pesos}-1} \alpha_i d_j^T A d_i$$

$$= \alpha_j d_j^T A d_j$$

Então, $\alpha_j = \dfrac{d_j^T A \theta^*}{d_j^T A d_j}$, $j = 0, \dots, N_{pesos}-1$

Esta expressão ainda não fornece uma solução viável, pois os coeficientes da representa

ção de θ^* dependem do próprio vetor θ^*.

- para eliminar θ^*, duas hipóteses adicionais são feitas:
  - * o problema é quadrático $\Rightarrow J(\theta) = \frac{1}{2}\theta^T Q \theta - b^T\theta$

    No ponto de mínimo (θ^*), o gradiente deve ser nulo.

    $$\nabla J(\theta^*) = 0 \Rightarrow Q\theta^* - b = 0 \Rightarrow Q\theta^* = b$$
  - * a matriz Q é a própria matriz A

Com isto:

$$\alpha_j = \frac{d_j^T b}{d_j^T Q d_j}, \quad j = 0, \dots, N_{passos}-1$$

e

$$\theta^* = \sum_{i=0}^{N_{passos}-1} \frac{d_j^T b}{d_j^T Q d_j} \cdot d_j$$

Como a solução θ^* deve ser obtida iterativamente a partir de uma condição inicial θ_0, podemos considerar que

$$\theta^* = \theta_0 + \alpha_0^* d_0 + \alpha_1^* d_1 + \dots + \alpha^*_{N_{passos}-1} d_{N_{passos}-1}$$

sendo os coeficientes dados por

$$\alpha_j^* = \frac{d_j^T Q(\theta^* - \theta_0)}{d_j^T Q d_j}, \quad j = 0, \dots, N_{passos}-1$$

Na j-ésima iteração ($0 < j < N_{passos}-1$), θ_j assume a forma

$$\theta_j = \theta_0 + \alpha_0^* d_0 + \dots + \alpha_{j-1}^* d_{j-1}$$

Pré-multiplicando à esquerda por $d_j^T Q$,

$$d_j^T Q \theta_j = d_j^T Q \theta_0 + \underbrace{\cancel{\alpha_0^* d_j^T Q d_0}}_{=0 \text{ pela } Q\text{-ortogonalidade}}^{\,0} + \cancel{\dots}^{\,0}$$

$$\Downarrow$$

Usando esta igualdade na equação de α_j^*:

$$\alpha_j^* = \frac{d_j^T Q(\theta^* - \theta_j)}{d_j^T Q d_j}, \quad j = 0, \dots, N_{passos}-1$$

Como $\nabla J(\theta_j) = Q\theta_j - b$ e $Q\theta^* = b$, é possível perceber que

$$\alpha_j^* = -\frac{d_j^T \nabla J(\theta_j)}{d_j^T Q d_j}, \quad j = 0, \dots, N_{passos}-1$$

Assim, a lei de ajuste do método das direções conjugadas é dada por:

$$\theta_{i+1} = \theta_i - \frac{d_i^T \nabla J(\theta_i)}{d_i^T Q d_i} \cdot d_i \qquad (B)$$

alcançando a solução ótima em N_{pesos} iterações se o problema for quadrático.

⇒ Antes de aplicar a regra de atualização em (B), é necessário obter as direções conjugadas $d_i \in \mathbb{R}^{N_{pesos}}$, $i = 0, \dots, N_{pesos}-1$. Uma maneira de obtê-las é dada a seguir:

$$\begin{cases} d_0 = -\nabla J(\theta_0) \\ d_{i+1} = -\nabla J(\theta_{i+1}) + \beta_i d_i, \end{cases}$$

com $\beta_i = \dfrac{\nabla J(\theta_{i+1})^T Q d_i}{d_i^T Q d_i}$

Quando temos um problema não-quadrático, a matriz Q será aproximada pela hessiana calculada sobre o ponto θ_i (lembrar da expansão em série de Taylor), ou seja, $Q = \nabla^2 J(\theta_i)$.

Assim, o <u>método do gradiente conjugado</u> pode ser resumido da seguinte forma:

- calcule $\nabla J(\theta_0)$ e $d_0 = -\nabla J(\theta_0)$
- para $i = 0, \dots, N_{pesos}-1$, faça:
  - (a) calcule $\nabla^2 J(\theta_i)$
  - (b) $\theta_{i+1} = \theta_i + \alpha_i d_i$, com $\alpha_i = \dfrac{-d_i^T \nabla J(\theta_i)}{d_i^T \nabla^2 J(\theta_i) d_i}$
  - (c) calcule $\nabla^2 J(\theta_{i+1})$
  - (d) se $i < N_{pesos}-1$, $d_{i+1} = -\nabla J(\theta_{i+1}) + \beta_i d_i$, onde $\beta_i = \dfrac{\nabla J(\theta_{i+1})^T \nabla^2 J(\theta_i) d_i}{d_i^T \nabla^2 J(\theta_i) d_i}$

<u>Vantagens</u>: (i) não há busca unidimensional do passo

(ii) não há inversão da matriz hessiana

<u>Desvantagens</u>: (i) cálculo e armazenamento de $\nabla^2 J(\cdot)$ a cada iteração

(ii) como o ajuste de θ considera informações da função até a 2ª ordem (e a mesma não necessariamente é quadrática), não há garantia de que o passo será minimizante.

(iii) o algoritmo não pode ser considerado globalmente convergente, pois não existe garantia de a matriz $\nabla^2 J(\cdot)$ ser definida positiva em todo o espaço $\mathbb{R}^{N_{pesos}}$.

Algumas adaptações foram propostas na tentativa de sanar algumas destas desvantagens.

⇒ <u>Polak-Ribière (PR)</u>:

- α_i é obtido com o auxílio de uma busca unidimensional
- β_i é calculado de forma aproximada:

$$\beta_i = \frac{g_{i+1}^T (g_{i+1} - g_i)}{g_i^T g_i}$$ ← usa apenas informação do gradiente (1ª ordem)

⇒ <u>Fletcher-Reeves (FR)</u>:

- também usa a busca unidimensional
- $\beta_i = \dfrac{\|g_{i+1}\|^2}{\|g_i\|^2}$

⇒ <u>Gradiente Conjugado Escalonado</u> (SCG):

• evita a busca unidimensional usando uma abordagem parecida com aquela introduzida no método Levenberg-Marquardt p/ escalonar o tamanho do passo α.

O passo α_i no método do gradiente conjugado é dado por $\alpha_i = -\dfrac{d_i^T \nabla J(\theta_i)}{d_i^T \nabla^2 J(\theta_i) d_i}$

A idéia de Moller (1993) é estimar o termo $s_i = \nabla^2 J(\theta_i) d_i$ através da aproximação:

$$s_i = \nabla^2 J(\theta_i) d_i \approx \frac{\nabla J(\theta_i + \sigma_i d_i) - \nabla J(\theta_i)}{\sigma_i} \quad ; \quad 0 < \sigma_i << 1$$

Para evitar problemas c/ a positividade (implícita) da hessiana, Moller (1993) introduziu um termo de regularização (no mesmo espírito de Levenberg-Marquardt):

$$s_i = \frac{\nabla J(\theta_i + \sigma_i d_i) - \nabla J(\theta_i)}{\sigma_i} + \lambda_i d_i$$

O denominador de α_i, $d_i^T \nabla^2 J(\theta_i) d_i$ será representado por $\delta_i = d_i^T \cdot s_i$

Se usássemos a matriz hessiana, o denominador sempre seria positivo. Por isso, o parâmetro λ_i será ajustado a cada iteração olhando o sinal de δ_i.

No artigo, Moller (1993) mostra como ajustar λ_i de modo a garantir $\delta_i > 0$.

Além disso, como λ_i escala a matriz hessiana de uma maneira artificial, a aproximação quadrática da função – $J_{quad}(\theta)$ – sobre a qual o algoritmo atua pode não ser uma boa aproximação de $J(\theta)$ em alguns pontos. Por isso, mesmo quando a matriz é positiva definida, pode ser necessário ajustar λ_i visando manter uma boa aproximação de $J(\theta)$.

- Moller, então, define:

$$\Delta_i = \frac{J(\theta_i) - J(\theta_i + \alpha_i d_i)}{J(\theta_i) - J_{quad}(\alpha_i d_i)} = \frac{2\delta_i [J(\theta_i) - J(\theta_i + \alpha_i d_i)]}{\mu_i^2},$$

onde $\mu_i = -d_i^T \nabla J(\theta_i)$.

Δ_i é uma medida de quão boa é a aproximação quadrática $J_{quad}(\alpha_i d_i)$ em relação a $J(\theta + \alpha_i d_i)$. Quanto mais próximo de 1, melhor a aproximação. Por isso, Moller propõe:

$\Delta_i > 0{,}75$, então $\lambda_i = 1/2\, \lambda_i$

$\Delta_i < 0{,}25$, então $\lambda_i = 4\lambda_i$

⇒ <u>Gradiente Conjugado Escalonado Modificado (SCGM)</u>

- em vez de trabalhar com uma aproximação (s_i) do produto da hessiana pela direção d_i, utiliza-se o operador diferencial proposto por Pearlmutter (1994) para obter o vetor produto <u>de maneira exata</u>.

Considere a expansão de Taylor do vetor gradiente $\nabla J(\cdot)$ em torno de θ:

$$\nabla J(\theta + \Delta\theta) = \nabla J(\theta) + \nabla^2 J(\theta)\Delta\theta + O(\|\Delta\theta\|^2)$$

Para $\Delta\theta = av$, c/ a constante e próximo de zero e $v \in \mathbb{R}^{N_{pesos}}$ um vetor arbitrário:

$$\nabla J(\theta + av) = \nabla J(\theta) + \nabla^2 J(\theta) av + O(a^2)$$

$$\therefore \quad \nabla^2 J(\theta) \cdot v = \frac{\nabla J(\theta + av) - \nabla J(\theta)}{a} + O(a)$$

Fazendo o limite $a \to 0$:

$$\nabla^2 J(\theta) \cdot v = \lim_{a \to 0} \frac{\nabla J(\theta + av) - \nabla J(\theta)}{a} = \frac{\partial}{\partial a} \nabla J(\theta + av)\Big|_{a=0}$$

<u>OPERADOR $R\{\cdot\}$</u>: $R_v\{f(\theta)\} = \frac{\partial}{\partial a} f(\theta + av)\Big|_{a=0}$

Logo, $\underline{\underline{R_v\{\nabla J(\theta)\} = \nabla^2 J(\theta) \cdot v}}$

Como $R\{\cdot\}$ é um operador diferencial, ele obedece às regras usuais de diferenciação.