Notação
Negrito = vetor.
Exemplos:
$\mathbf{x}$ é um vetor $[x_0, ..., x_n]^T$
$\boldsymbol{\phi}(x)$ é um vetor de funções $[\phi_0(x), ..., \phi_n(x)]^T$

1 Introdução

Este texto é um resumo de alguns conceitos sobre regressão apresentados no livro *Pattern Recognition and Machine Learning*, e a documentação de algumas ideias que eu tive sobre como gerar curvas de regressão suaves minimizando, além do erro entre as predições e os dados, a derivada segunda da curva de regressão. O texto serve a quem desejar compreender os algorítimos implementados aqui, e a mim como referência futura.

2 Regressão

O objetivo da regressão é fazer a previsões dos valores de uma *target variable* t dado o valor de uma *input variable* $\mathbf{x}$. No geral $\mathbf{x}$ é um vetor de dimensão qualquer, mas tratarei aqui apenas do caso em que $\mathbf{x}$ tem dimensão 1.

Dado um conjunto de dados composto de observações $\{\mathbf{x}_n\}$ e suas respectivas *target variables* $\{t_n\}$, o que se pretende fazer é a previsão do valor de t dado um valor de $\mathbf{x}$ que não necessariamente está presente nas observações. Em outras palavras, quer-se encontrar uma função $y(\mathbf{x})$ que associa cada valor possível de $\mathbf{x}$ a uma previsão de t.

A função $y(\mathbf{x})$ pode ser encontrada através do ajuste de parâmetros de uma função $y(\mathbf{x}, \mathbf{w})$, onde $\mathbf{w}$ é um vetor de parâmetros ajustáveis. Os valores finais dos parâmetros ajustáveis $\mathbf{w}_f$ devem ser escolhidos de maneira a minimizar (ou quase isso) o valor de $\mathcal{L}(\mathbf{w})$, onde $\mathcal{L}$ é uma função de perda adequada. A função de perda $\mathcal{L}$ é o criterio pelo qual se julga o quão adequadas são as previsões de $y(\mathbf{x}) = y(\mathbf{x}, \mathbf{w}_f)$ dado o conjunto de dados.

Um método pelo qual os parâmetros ajustáveis podem ser atualizados é o da *descida de gradiente*, onde os parâmetros são modificados de maneira iterativa na direção do negativo do gradiente da função de perda,

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \lambda \nabla \mathcal{L}(\mathbf{w})$$

ou seja, a cada passo os parâmetros ajustáveis $\mathbf{w}$ são modificados de maneira a diminuir o valor da função de perda $\mathcal{L}(\mathbf{w})$. Em alguns casos, porém, uma fórmula fechada para os valores ótimos de $\mathbf{w}$ pode ser encontrada, como será visto mais adiante.

3 Modelos Lineares de Regressão

Os modelos lineares de regressão são aqueles cuja função $y(\mathbf{x}, \mathbf{w})$ é linear em relação aos parâmetros ajustáveis $\mathbf{w}$, o que não quer dizer que $y(\mathbf{x}, \mathbf{w})$ seja necessariamente linear com relação a $\mathbf{x}$. No geral, $y(\mathbf{x}, \mathbf{w})$ é uma combinação linear de *basis functions* $\phi(\mathbf{x})$, funções de $\mathbf{x}$ que podem ou não ser lineares,

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{M-1} w_i \phi_i(\mathbf{x}_n)$$

onde M é a quantidade de parâmetros ajustáveis. A equação anterior pode ser simplificada para $\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$ usando notação vetorial, assumindo que $\phi_0 = 1$.

A função de perda utilizada aqui é a soma dos quadrados dos erros, isto é, as diferenças entre as previsões do modelo e o valor real das observações $t_n - y(\mathbf{x}_n, \mathbf{w})$ são elevadas ao quadrado e somadas, por todas as instâncias de observação, para gerar um valor que representa o quão adequada é a escolha de valores para $\mathbf{w}$.

$$\mathcal{L}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{t_n - y(\mathbf{x}_n, \mathbf{w})\}^2$$

onde N é a quantidade de observações. Tal escolha de função de perda é justificada se assumirmos que os dados observados são gerados a partir de uma função determinística somada a um ruído gaussiano, o que é o caso, os dados artificiais gerados aqui são da forma $t = \mathcal{N}(t|sin(x), \sigma)$.

Essa escolha de função de perda, combinada ao fato de $y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$ ser linear em relação a $\mathbf{w}$, nos leva a seguinte equação para o gradiente da função de perda

$$\nabla \mathcal{L}(\mathbf{w}) = \sum_{n=1}^{N} \{t_n - \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n)\} \boldsymbol{\phi}(\mathbf{x}_n)^T$$

Seja

$$\boldsymbol{\Phi} = \begin{bmatrix} \phi_0(x_0) & \phi_1(x_0) & \cdots & \phi_{M-1}(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_{M-1}(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \cdots & \phi_{M-1}(x_n) \end{bmatrix},$$

a equação anterior para o gradiente da função de perda pode ser reescrita da seguinte maneira:

$$\nabla \mathcal{L}(\mathbf{w}) = (\mathbf{t}^T - \mathbf{w}^T \boldsymbol{\Phi}^T) \boldsymbol{\Phi}$$

expandindo as matrizes,

$$\nabla \mathcal{L}(\mathbf{w}) = \begin{bmatrix} t_0 - \mathbf{w}^T \boldsymbol{\phi}(x_0), \cdots, t_n - \mathbf{w}^T \boldsymbol{\phi}(x_n) \end{bmatrix} \begin{bmatrix} \phi_0(x_0) & \cdots & \phi_{M-1}(x_0) \\ \vdots & \ddots & \vdots \\ \phi_0(x_n) & \cdots & \phi_{M-1}(x_n) \end{bmatrix}$$

fazendo o gradiente da função de perda igual a zero temos,

$$(\mathbf{t}^T - \mathbf{w}^T\mathbf{\Phi}^T)\mathbf{\Phi} = 0$$
$$\mathbf{t}^T\mathbf{\Phi} - \mathbf{w}^T\mathbf{\Phi}^T\mathbf{\Phi} = 0$$
$$\mathbf{w}^T\mathbf{\Phi}^T\mathbf{\Phi} = \mathbf{t}^T\mathbf{\Phi}$$
$$\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{w} = \mathbf{\Phi}^T\mathbf{t}$$
$$\mathbf{w} = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{t}$$

uma fórmula fechada para os valores ótimos de $\mathbf{w}$.

4 Regularização

Se o modelo sendo ajustado é flexível demais para a quantidade de dados disponíveis, o resultado é uma curva cheia de mudanças bruscas que falha em capturar o padrão principal presente nos dados, pois, sendo flexível demais, se adapta ao ruído nos dados, o que faz ofuscar o padrão geral. Esse problema é denominado *overfitting*.

Uma maneira de lidar com o problema de *overfitting* é a penalização de valores muito grandes para os parâmetros $\mathbf{w}$. Uma maneira simples de fazer isso é adicionar a soma dos quadrados dos parametros $\mathbf{w}^T\mathbf{w}$ a função de perda

$$\mathcal{L}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - y(\mathbf{x}_n, \mathbf{w})\}^2 + \frac{1}{2}\lambda\mathbf{w}^T\mathbf{w}$$

onde λ é o coeficiente de regularização, que controla a força da penalização aplicada.

Da adição do fator do termo de regularização na função de perda, resulta o seguinte gradiente

$$\nabla\mathcal{L}(\mathbf{w}) = (\mathbf{t}^T - \mathbf{w}^T\mathbf{\Phi}^T)\mathbf{\Phi} + \lambda\mathbf{w}^T$$

A vantagem de se utilizar a soma dos quadrados como termo de regularização é que a função de perda resultante ainda é uma função quadrática do vetor de parâmetros $\mathbf{w}$, portanto, possui formula fechada para os valores ótimos de $\mathbf{w}$

$$(\mathbf{t}^T - \mathbf{w}^T\mathbf{\Phi}^T)\mathbf{\Phi} + \lambda\mathbf{w}^T = 0$$
$$\mathbf{w}^T\mathbf{\Phi}^T\mathbf{\Phi} - \lambda\mathbf{w}^T = \mathbf{t}^T\mathbf{\Phi}$$
$$\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{w} - \lambda\mathbf{w} = \mathbf{\Phi}^T\mathbf{t} \tag{1}$$
$$\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{w} - \lambda I_M\mathbf{w} = \mathbf{\Phi}^T\mathbf{t}$$
$$(\mathbf{\Phi}^T\mathbf{\Phi} - \lambda I_M)\mathbf{w} = \mathbf{\Phi}^T\mathbf{t}$$
$$\mathbf{w} = (\mathbf{\Phi}^T\mathbf{\Phi} - \lambda I_M)^{-1}\mathbf{\Phi}^T\mathbf{t} \tag{2}$$

A equação (2) contém um erro de sinal, por algum motivo o resultado que segue do desenvolvimento em (1) tem o termo $-\lambda I_M$ enquanto no livro esse termo é positivo. Não consegui entender de onde surgiu esse erro, mas não tem muita importância, visto que o termo já esta sendo multiplicado por um fator λ.

5 Penalização de curvatura

Me veio a ideia de que as curvas de regressão poderiam ser feitas mais suaves se minimizarmos, além do erro, a derivada de segunda ordem da curva $y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$. Para tal, adiciona-se o seguinte termo a função de perda

$$\int_a^b \left(\frac{d^2}{dx^2} [\mathbf{w}^T \boldsymbol{\phi}(x)] \right)^2 dx \tag{3}$$

o intervalo $[a, b]$ é o intervalo relevante para a regressão.

A rigor, o termo em (3) não representa corretamente a curvatura de $y(\mathbf{x}, \mathbf{w})$, a representação correta utilizaria o módulo da derivada segunda de y a fim de penalizar tanto as curvas convexas quanto as concavas, ao invés do quadrado; o quadrado é mais coveniente, no entanto.

Da adição de termo (3) na função de perda resulta

$$\nabla \mathcal{L}(\mathbf{w}) = (\mathbf{t}^T - \mathbf{w}^T \boldsymbol{\Phi}^T)\boldsymbol{\Phi} + \lambda \mathbf{w}^T \boldsymbol{\Phi}'' \tag{4}$$

onde

$$\boldsymbol{\Phi}'' = \begin{bmatrix} \int \phi_0''(x)\phi_0''(x)dx & \int \phi_0''(x)\phi_1''(x)dx & \cdots & \int \phi_0''(x)\phi_n''(x)dx \\ \int \phi_1''(x)\phi_0''(x)dx & \int \phi_1''(x)\phi_1''(x)dx & \cdots & \int \phi_1''(x)\phi_n''(x)dx \\ \vdots & \vdots & \ddots & \vdots \\ \int \phi_n''(x)\phi_0''(x)dx & \int \phi_n''(x)\phi_1''(x)dx & \cdots & \int \phi_n''(x)\phi_n''(x)dx \end{bmatrix}$$

De maneira similar aos casos anteriores, uma fórmula fechada para $\mathbf{w}$ pode ser encontrada

$$\mathbf{w} = (\boldsymbol{\Phi}^T \boldsymbol{\Phi} - \lambda \boldsymbol{\Phi}'')^{-1} \boldsymbol{\Phi}^T \mathbf{t}$$

como em (2) essa equação também provavelmente tem um erro de sinal.

A título de exemplo, quando as bases são polinômios $\phi_i(x) = x^i$ os elementos ϕ_{ij}'' de $\boldsymbol{\Phi}''$ são

$$\phi_{ij}'' = \left(\frac{(j^2 - j)(i^2 - i)x^{i+j-3}}{i + j - 3} \right) \Bigg|_a^b$$

6 Referências

Pattern Recognition and Machine Learning - Christopher Bishop, Capítulo 3.