6
O método de Regressão Construtiva em Variedades Implícitas

O objetivo do método de Regressão Construtiva em Variedades Implícitas (RCVI) é obter uma aproximação implícita construtiva da variedade e, com isso, realizar uma regressão, tendo como dados de entrada apenas pontos esparsos. Neste capítulo combinamos as informações obtidas na votação com tensores (capítulo 4) com a aproximação implícita da variedade (capítulo 5) e explicaremos como realizar a regressão na estrutura de árvore obtida.

O conjunto de dados de entrada é formado apenas por pontos esparsos $(\mathbf{x}, y)$, onde $\mathbf{x} \in \mathbb{R}^{\mathrm{n}}$ e $y \in \mathbb{R}$:

- $\mathbb{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathbb{R}^n$;
- $\mathbb{Y} = \{y_1, \ldots, y_N\} \subset \mathbb{R}$;

Para o método TV, apenas os dados de entrada $\mathbb{X}$ são requeridos. Ao término da execução, obtemos uma estimativa da dimensionalidade intrínseca bem como a informação das tangentes e normais em cada ponto da variedade. No entanto, apenas utilizamos os vetores tangentes para construção implícita da variedade. As estimativas dos vetores normais e tangentes num ponto da variedade, fornecida pelo método TV, são obtidas pelo cálculo dos autovetores de uma matriz R, resultante do acúmulo de votos recebidos por aquele ponto, dos dados dentro de uma vizinhança. Desse modo, não temos informação consistente da orientação das normais obtidas, logo, a informação dos vetores normais não será utilizada na construção do algoritmo de aproximação da variedade.

Admitimos que nosso conjunto de dados representa uma variedade M^m. Dessa maneira, a dimensionalidade intrínseca local deve apontar para a dimensionalidade m e, assim, obtemos o valor da dimensionalidade da variedade baseado na dimensionalidade mais frequente. Ressaltamos que, ao término do processo de votação, decompomos o tensor R, de cada dado, o qual é resultante do acúmulo de votos dos dados vizinhos e, com isso, obtemos uma estimativa da dimensionalidade intrínseca local dos dados, a qual vamos supor que seja

m. Assim, temos que os primeiros $n-m$ autovetores correspondendo aos maiores autovalores são as normais a variedade naquele ponto e os m restantes representam as tangentes. Conforme experimentos realizados no capítulo 4 e em (47), mesmo que a dimensionalidade local estimada não esteja correta, o método apresenta uma estimativa confiável dos vetores normais e tangentes obtidos.

Essas estimativas obtidas do TV, quanto a dimensionalidade e vetores tangentes, serão utilizadas na etapa seguinte de aproximação implícita da variedade. Nesta etapa, utiliza-se a partição da unidade de maneira a trabalhar com formas mais simples em cada região particionada do domínio. Para isso, podem ser utilizadas malhas 2^n-ádicas ou árvores de partição binária do espaço. As informações de cada região do domínio podem ser unidas de forma a se obter uma aproximação global. No caso da malha 2^n-ádica as soluções locais são unidas por funções peso suaves e, no caso da árvore de partição binária do espaço, utilizam-se funções de transição suaves. Uma vez construída a árvore, essa estrutura será aproveitada para realizar uma regressão conforme estudaremos neste capítulo. Vale a pena ressaltar ainda que, trabalhamos com os dados no espaço de entrada original, isto é, não empregamos nenhum método de redução da dimensionalidade, o que fazemos é usar o TV para extrair informações que são necessárias para construção implícita da variedade. Na fase seguinte, ao particionar os dados de entrada em conjuntos de dados menores, ainda que a dimensionalidade do espaço de entrada seja grande, os conjuntos de dados em cada sub-região do domínio podem ser ajustados simultaneamente utilizando sistema em paralelo, o que permite reduzir o tempo computacional.

Portanto, o método RCVI divide-se em duas etapas distintas: a partir de dados esparsos, obtêm-se informações sobre a dimensionalidade dos dados de entrada e dos vetores tangentes a variedade em cada ponto, empregando o método TV. Em seguida, utiliza-se uma partição da unidade a fim de implementar uma aproximação construtiva da variedade implícita.

O método RCVI foi implementado utilizando uma linguagem de programação dinâmica, orientada a objetos, chamada Python[1]. Para execução do algoritmo TV é necessária uma busca pelos vizinhos mais próximos. Para realizar essa tarefa, utilizamos o módulo python chamado scikit-learn [2] (51) e realizamos a busca pelos vizinhos mais próximos usando uma estrutura de árvore chamada *Kd-Tree* (5). Nesse módulo, estão também disponíveis alguns conjuntos de dados padrões, além de métodos de aprendizagem supervisionada, não supervisionada e de aprendizagem de variedades. Utilizamos alguns

[1] http://www.python.org/
[2] http://scikit-learn.org

dos métodos de aprendizagem supervisionada desta classe para comparar o desempenho deles com o do método proposto, cujos resultados serão exibidos no próximo capítulo. Vamos brevemente revisar alguns métodos de regressão baseados em árvore e exibir como será aplicado ao método RCVI.

Seja $\mathbf{x}_j \in \mathbb{X} \subset \mathbb{R}^n$ um vetor contendo n variáveis explicativas para uma resposta univariada contínua $y_j \in \mathbb{R}$. Queremos obter uma função f que descreva a relação entre $\mathbf{x_j}$ e y_j. A relação entre $\mathbf{x}_j$ e y_j segue o modelo de regressão:

$$y_j = f(\mathbf{x}_j) + \epsilon_j,$$

onde f é desconhecida e não existem hipóteses sobre a distribuição do termo aleatório ϵ_j.

Dentre os métodos baseados em árvore, o modelo CART (10) é um dos mais populares. No método CART, a função de predição f é dada por:

$$f(\mathbf{x}) = \sum_{k=1}^{L} c_k(\mathbf{x})\chi_k(\mathbf{x}),$$

onde L é o número de folhas, $\{\mathcal{R}_k\}_{k=1}^{L}$ são retângulos que formam uma partição do domínio de f, obtidos pela estrutura de árvore binária, com hiperplanos ortogonais aos eixos das variáveis preditoras; c_k são constantes em $\mathcal{R}_k$ e χ é a função característica da região $\mathcal{R}_k$ conforme descrita em 5.2.2. O modelo CART associa um valor constante a cada região retangular, logo, o valor resposta da função estimada quando o ponto $\mathbf{x}$ está na região $\mathcal{R}_k$ é c_k. Em geral, essas constantes c_k são obtidas como a média de todos os valores de y presentes na região $\mathcal{R}_k$.

No modelo STR-Tree (15), cada nó interno guarda uma função logística definida como:

$$g(\mathbf{x}) = \frac{1}{1 + e^{-\lambda(x_j - a_j)}},$$

onde j é a coordenada de divisão desse nó; a é o valor onde essa divisão é feita e λ é um parâmetro que controla a suavidade da função logística. A função de predição fica definida como:

$$f(\mathbf{x}) = \sum_{k=1}^{L} c_k(\mathbf{x})B_k(\mathbf{x}),$$

onde B_k é o produto de funções logísticas.

No modelo RCRI (30), um novo modelo de regressão baseado em árvores é proposto. Nele, a função logística é substituída por uma função degrau polinomial e a partição do domínio é feita por regiões definidas implicitamente.

No modelo RCVI deste trabalho, consideramos que em cada região $\mathcal{R}_k$ teremos uma função $f_k(\mathbf{x}) = c_k$ constante. Temos duas propostas de partição do domínio: malhas 2^n-ádicas e árvores de partição binária do espaço. A função de predição, dependendo da subdivisão do domínio é, respectivamente, dada por:

$$\widehat{f}(\mathbf{x}) = \frac{\sum_{k=1}^{L} \omega_k(\mathbf{x}) c_k}{\sum_{k=1}^{L} \omega_k(\mathbf{x})} \quad \text{e} \quad \widehat{f}(\mathbf{x}) = \sum_{k=1}^{L} B_k(\mathbf{x}) c_k$$

Com relação as funções de transição utilizadas na construção da árvore de partição binária do espaço, $B_k(\mathbf{x})$ indica um peso associado a regressão na folha k quando o dado de entrada é $\mathbf{x}$. Por este motivo, chamaremos as funções de transição B_k de funções peso. A proposição 6.1 nos mostra que o erro global na estimativa da saída $y = f(\mathbf{x})$ pode ser controlado através do erro local, isto é, pela estimativa de $f_k(\mathbf{x})$ obtida em cada folha k e sua vizinhança.

Proposição 6.1. *Seja f uma função real definida em $\mathbb{R}^n, \{\mathcal{R}_k\}_{k=1}^{L}$ uma partição de $\mathbb{R}^n$ e $\mathcal{N}_k$ a vizinhança de $\mathcal{R}_k$. Considere $B_1, B_2, \ldots, B_k$ funções de transição correspondentes aos subconjuntos $\{\mathcal{R}_k\}_{k=1}^{L}$, tais que $B_k = 0$, se $\mathbf{x} \notin \mathcal{R}_k \cup \mathcal{N}_k$ e $\sum_{k=1}^{L} B_k(\mathbf{x}) = 1$. Se para cada $k = 1, \ldots, L$, f_k é uma aproximação local de f tal que $|f(\mathbf{x}) - f_k(\mathbf{x})| < \epsilon, \forall\, \mathbf{x} \in \mathcal{R}_k \cup \mathcal{N}_k$ então a aproximação global definida por $\widehat{f}(\mathbf{x}) = \sum_{k=1}^{l} B_k f_k(\mathbf{x})$ satisfaz a seguinte propriedade:*

$$|f(\mathbf{x}) - \widehat{f}(\mathbf{x})| < \epsilon, \forall\, \mathbf{x} \in \mathbb{R}^n.$$

A demonstração pode ser obtida em (30).

Vamos obter duas maneiras diferentes de estimar c_k seguindo a estimativa local e global apresentadas em (30).

6.1 Estimativa Local

Nesta formulação, para estimar c_k só serão considerados os pontos pertencentes a região $\mathcal{R}_k \cup \mathcal{N}_k$, isto é, apenas os pontos na região definida pela folha k e sua vizinhança.

O estimador c_k será obtido pela média dos valores de y associados aos valores de $\mathbf{x}$ que estão na região $\mathcal{R}_k \cup \mathcal{N}_k$, ponderada pela função peso em k. A estimativa fornecida por cada folha k será:

- Para a subdivisão do domínio dada pela árvore binária:

$$\hat{c_k} = \frac{\sum_{j=1}^{N} B_k(\mathbf{x}_j) y_j}{\sum_{j=1}^{N} B_k(\mathbf{x_j})} \tag{6-1}$$

- Para a subdivisão do domínio dada pela malha 2^n-ádica:

$$\hat{c_k} = \frac{\sum_{j=1}^{N} \omega_k(\mathbf{x}_j) y_j}{\sum_{j=1}^{N} \omega_k(\mathbf{x}_j)} \tag{6-2}$$

Se para toda folha k tivermos um erro local menor do que ϵ, isto é, se $\forall\ 1 \leq k \leq L$ e $\forall\ 1 \leq j \leq N$ tal que $\mathbf{x}_j \in \mathcal{R}_k \cup \mathcal{N}_{R_k}$, tem-se $erro_j = |y_j - \hat{c_k}| < \epsilon$, então, pela aplicação da proposição 6.1, teremos também erro global menor do que ϵ. Assim, pode-se definir um limite superior para o erro global adicionando-se um critério de parada que controle o erro local.

6.2 Estimativa Global

Para obter a estimativa global para c_k, todos os valores $c_k, 1 \leq k \leq L$ serão estimados ao mesmo tempo, logo, todos os pontos da amostra serão levados em consideração. Nessa formulação, o vetor $\mathbf{c} = (c_1, \ldots, c_L)$ é estimado de modo a minimizar o erro total da amostra. A primeira hipótese a ser feita é a de que $\hat{f}$ representa a relação entre $\mathbf{x_j}$ e y_j, isto é:

$$y_j = \widehat{f}(\mathbf{x}_j) + \varepsilon_j = \sum_{k=1}^{L} B_k(\mathbf{x}_j) c_k + \varepsilon_j \qquad \forall\ 1 \leq j \leq N,$$

onde ε_j é uma variável aleatória normal, de média zero e variância σ^2, que representa o erro na estimativa.

O vetor de coeficientes $\mathbf{c} = (c_1, \ldots, c_L)^t$ é estimado por mínimos quadrados. Portanto, $\mathbf{c}$ é obtido resolvendo-se o seguinte sistema:

$$B^t B \mathbf{c} = B^t \mathbf{y},$$

onde $B, \mathbf{c}$ e $\mathbf{y}$ são definidos como:

$$B = \begin{pmatrix} B_1(\mathbf{x}_1) & B_2(\mathbf{x}_1) & \ldots B_L(\mathbf{x}_1) \\ B_1(\mathbf{x}_2) & B_2(\mathbf{x}_2) & \ldots B_L(\mathbf{x}_2) \\ \vdots & \vdots & \vdots \\ B_1(\mathbf{x}_N) & B_2(\mathbf{x}_N) & \ldots B_L(\mathbf{x}_N) \end{pmatrix} \quad \mathbf{c} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_L \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}$$

Logo,

$$\hat{\mathbf{c}} = (B^t B)^{-1} B^t \mathbf{y}$$

é o estimador dos parâmetros em $\mathbf{c}$.