

Núcleos e sinais polinomais

Praciano-Pereira, T

Sobral Matemática

17 de setembro de 2011

tarcisio@member.ams.org

préprints da Sobral Matemática

no. 2011.06

Editor Tarcisio Praciano-Pereira,

tarcisio@member.ams.org

Resumo

Neste artigo vou mostrar como podemos construir kernels (núcleos) e sinais polinomiais de classe C^n e a suporte compacto, num espaço de dimensão finita. A metodologia usada é essencialmente a descrita em [1], splines por convolução, em que o caso de dimensão um foi tratado, e aqui estou expondo a construção para um espaço qualquer de dimensão n . O objetivo é produzir a ferramenta com a qual eu vou construir uma partição da unidade subordinada a cobertura de um retângulo conduzindo a um projetor de interpolação. Em trabalho posterior vou mostrar como a convolução pode ser facilmente substituída, na prática, por translações. Aqui vou me restringir à construção teórica em que a convolução tem um papel central.

palavras chave: *splines multivariados, n-splines multivariados a suporte compacto, projetor de interpolação.*

In this paper I will show how we can build *compact support polynomial kernels and signals* of class C^n in a finite dimensional space. The methodology used is essentially the one described in [1], convolution splines, where the one dimensional case was treated and here I am explaining the construction to any space of dimension n . This is the tool I need to build a partition of unity subordinate to a cover of a rectangle, leading to an interpolation projector. In a later work I will show how the convolution can be easily replaced, in practical works, by a translation. Here I will restrict myself to the theoretical construct in which the convolution has a central role.

key words: *multivariate splines, compact support multivariate n-splines, interpolation projector.*

1 Definições

Neste artigo vou chamar de **sinal** a uma função quase-contínua, quer dizer, contínua exceto, possivelmente, num conjunto de medida zero, e a suporte compacto. Um exemplo típico é uma função característica de um intervalo compacto da reta, ou uma combinação linear finita delas. Em dimensão maior do que um o exemplo seria de uma combinação linear finita de funções características de domínios compactos, conjuntos compactos com interior não vazio, por exemplo, funções características de retângulos, em que “retângulo” é um intervalo compacto ou um produto cartesiano de $n \geq 1$ intervalos compactos em que a quantidade de fatores é a dimensão do espaço.

Podemos atingir uma grande generalidade nos restringindo a retângulos sendo esta a minha atitude neste artigo.

Kernel é um sinal positivo cuja integral é 1. Na literatura, em português, também se usa a palavra **núcleo**, mas vou dar preferência ao nome de origem alemã, por se aproximar melhor da nomenclatura internacional e porque ambos os nomes são usados na literatura portuguesa. Um exemplo simples de kernel é $\frac{1}{2\epsilon}\chi_{[-\epsilon, \epsilon]}$ a função característica do intervalo $[-\epsilon, \epsilon]$ corrigida com o fator $\frac{1}{2\epsilon}$ para garantir que a integral seja 1, ou, num espaço de dimensão n

$$\left(\frac{1}{2\epsilon}\right)^n \chi_{[-\epsilon, \epsilon]^n}$$

em que a potência do intervalo significa produto cartesiano.

Usando a notação η_ϵ para qualquer dos exemplos acima de kernel, podemos facilmente construir uma “*unidade aproximada*”, seguindo a denominação usada em [4]:

$$\eta_{n,\epsilon}(t) = n\eta_\epsilon(nt) \tag{1}$$

uma sucessão que converge fracamente para a medida de Dirac concentrada na origem. Esta é uma grande importância dos núcleos, podemos produzir aproximações com eles.

Teorema 1 (novo kernel) *Convolução de kernels*

A convolução de dois kernels, é um kernel, se a medida for invariante por translação.

A medida será invariante por translação uma vez que estarei sempre usando a medida de Lebesgue em \mathbf{R}^n . A convolução de dois kernels é ainda um kernel, porém o suporte fica dilatado porque será a união dos suportes dos dois kernels o que reduz a precisão na aproximação por convolução. Isto pode ser facilmente contornado, para obter suporte, arbitrariamente limitado, aplicando uma dilação como na equação (1).

Como consequência destes resultados tenho um estoque abundante de kernels e sinais de classe de diferenciabilidade arbitrária, polinomiais, construídos com funções características de retângulos.

Para os objetivos que tenho em mente, superfícies diferenciáveis obtidas como gráficos de funções diferenciáveis definidas num domínio convexo, trabalhar com malhas uniformes cujos elementos sejam retângulos é suficiente para

fazer aproximações com grande precisão, apenas não vou tratar desta aproximações neste trabalho e sim em outro vindouro. O uso da uniformidade da malha permite construir programas mais simples, mas não é uma questão essencial, como ser visto em [2]. Parece-me que a construção do algoritmo para o caso multidimensional será uma simples generalização do caso unidimensional e isto será assunto para um outro artigo também.

Para terminar esta seção introdutória mostrarei alguns cálculos que certamente irão satisfazer certas curiosidades, como a convolução de dois kernels definidos com funções características de retângulos, no \mathbf{R}^2 . Para uma dimensão maior o trabalho é apenas mais intenso. Há dois scripts de `gnuplot`, parcialmente apresentados aqui, que permitem uma visualização muito melhor do que os gráficos incluídos no artigo, ver [3], onde há um link para a página de onde eles podem ser baixados e executados localmente no computador.

1.1 Convolução de dois kernels: 1-splines bivariado

Usando a notação definida acima,

$$\eta_\epsilon = \frac{1}{2\epsilon} \chi_{[-\epsilon, \epsilon]} \quad (2)$$

vou agora definir o *kernel* Q_ϵ usando a *função característica do retângulo*

$$[-\epsilon, \epsilon] \times [-\epsilon, \epsilon]$$

multiplicada por fator de correção (para garantir integral seja 1)

$$Q_\epsilon(x, y) = \eta_\epsilon(x)\eta_\epsilon(y) \quad (3)$$

é fácil de ver-se que esta expressão define uma função característica com a amplitude alterada, do retângulo mencionado acima, se x ou y tiver módulo maior do que ϵ a expressão é nula, e se ambos forem menores ou igual a ϵ o produto vale $\frac{1}{4\epsilon^2}$. Esta expressão vai simplificar imensamente o cálculo da integral, e vou me resumir a calcular o quadrado por convolução de Q_ϵ .

$$Q_\epsilon * Q_\epsilon(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} Q_\epsilon(s, t) Q_\epsilon(x - s, y - t) ds dt = \quad (4)$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \eta(s)\eta(t)\eta(x - s)\eta(y - t) ds dt = \quad (5)$$

$$= \left(\int_{-\infty}^{\infty} \eta(s)\eta(x - s) ds \right) \left(\int_{-\infty}^{\infty} \eta(t)\eta(y - t) dt \right) = \quad (6)$$

$$= \left(\int_{-\epsilon}^{\epsilon} \eta(s)\eta(x - s) ds \right) \left(\int_{-\epsilon}^{\epsilon} \eta(t)\eta(y - t) dt \right) = \quad (7)$$

$$= \left(\int_{-\epsilon}^{\epsilon} \eta(x - s) ds \right) \left(\int_{-\epsilon}^{\epsilon} \eta(y - t) dt \right) = \quad (8)$$

$$= \left(- \int_{x+\epsilon}^{x-\epsilon} \eta(s) ds \right) \left(- \int_{x+\epsilon}^{x-\epsilon} \eta(t) dt \right) = \quad (9)$$

$$= \left(\int_{x-\epsilon}^{x+\epsilon} \eta(s) ds \right) \left(\int_{x-\epsilon}^{x+\epsilon} \eta(t) dt \right) \quad (10)$$

As integrais na equação (10) são idênticas definindo expressões em x ou em y que podem ser facilmente calculadas analisando a possibilidade de que o limite superior de integração esteja antes do suporte ou dentro do suporte, e depois a mesma análise para o limite inferior de integração estando dentro do suporte ou posterior ao suporte, exatamente nesta ordem, o que produz as possibilidades:

$$\begin{cases} x + \epsilon < -\epsilon & x < -2\epsilon \\ x + \epsilon \in [-\epsilon, \epsilon] & -2\epsilon \leq x \leq 0 \\ x - \epsilon \in [-\epsilon, \epsilon] & 0 \leq x \leq 2\epsilon \\ x - \epsilon > \epsilon & x > 2\epsilon \end{cases} \quad (11)$$

dando as seguintes possibilidades de valores para a integral, em x :

$$\begin{cases} x < -2\epsilon & 0 \\ -2\epsilon \leq x \leq 0 & x + 2\epsilon \\ 0 \leq x \leq 2\epsilon & 2\epsilon - x \\ x > 2\epsilon & 0 \end{cases} \quad (12)$$

e trocando x por y se tem o valor da outra integral. Para cada uma destas possibilidades temos todas as quatro outras, em relação a outra variável, mas algumas ficam naturalmente excluídas, por exemplo, quando $x < -2\epsilon$ para qualquer das hipóteses sobre y o resultado será zero. O mesmo vai acontecer quando $|x| > 2\epsilon$, $|y| > 2\epsilon$, porque o suporte da potência por convolução, neste caso de suportes centrados na origem, é a soma dos diâmetros dos suportes. As possibilidades finalmente serão:

$$\begin{cases} x < -2\epsilon & 0 \\ x \leq 0; y < -2\epsilon & 0 \\ x \leq 0; -2\epsilon \leq y < 0 & (x + 2\epsilon)(y + 2\epsilon) \\ x \leq 0; 0 \leq y < 2\epsilon & (x + 2\epsilon)(2\epsilon - y) \\ 0 < x \leq 2\epsilon; y < -2\epsilon & 0 \\ 0 < x \leq 2\epsilon; -2\epsilon \leq y < 0 & (2\epsilon - x)(y + 2\epsilon) \\ 0 < x \leq 2\epsilon; y \leq 2\epsilon & (2\epsilon - x)(2\epsilon - y) \\ x > 2\epsilon & 0 \end{cases} \quad (13)$$

Este sistema de equações pode ser traduzido assim na linguagem de `gnuplot`, ou em qualquer linguagem com a sintaxe de `C++`.

```
ep = 1; ## epsilon na variável ep
set xrange [-5:5];
set yrange [-5:5];
Q(x,y) = (x <= -2*ep)?0:\
```

```

(x<=0)*(y<-2*ep)?0:\
(x<=0)*(y<0)?(y+2*ep)*(x+2*ep):\
(x<=0)*(y<2*ep)?(2*ep-y)*(x+2*ep):\
(x<=2*ep)*(y<-2*ep)?0:\
(x<=2*ep)*(y<0)?(y+2*ep)*(2*ep-x):\
(x<=2*ep)*(y<2*ep)?(2*ep-y)*(2*ep-x):0
splot Q(x,y)

```

Os programas relacionados com este artigo são descritos em [3], onde há um link para a página de onde eles podem ser baixados.

Rodando o programa, ver [3], é possível ter uma visão melhor do gráfico que aparece na figura (1) página 4,

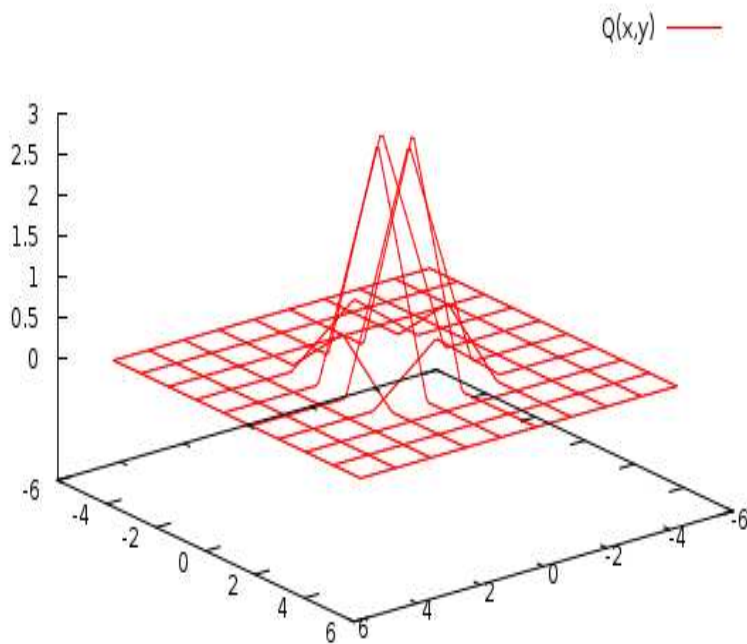


Figura 1: Kernel bivariado 1-spline

1.2 O plano do trabalho

Na segunda seção vou construir uma partição da unidade de classe \mathcal{C}^n associada a uma partição particular de um retângulo. Na terceira seção vou mostrar como facilmente posso obter o mesmo resultado com uma partição arbitrária de um domínio Ω de \mathbf{R}^2 . Na quarta seção vou construir o projetor de interpolação associado à partição da unidade relativamente a uma partição de um conjunto Ω de \mathbf{R}^2 . Este trabalho é uma generalização de [2]. A generalização para trabalhar com domínios do \mathbf{R}^n poderá ser feita posteriormente.

2 Partição da unidade sobre um retângulo do plano

O título não corresponde à definição de uma partição da unidade, irei fazer a correção ao fazer a construção da *partição da unidade*. O objetivo do título é indicar que a região Ω de referência é um retângulo.

Seja Ω um retângulo do plano com lados paralelos aos eixos coordenados e uma dupla sequência finita,

$$(x_i, y_j)_{i \in I = \{0, \dots, n-1\}; j \in J = \{0, \dots, m-1\}} \quad (14)$$

em que $I, J \subset \mathbf{N}$. Ou seja tenho uma rede definida em Ω , como produto de redes uniformes definidas nos lados. Assim Ω está particionado em $n \times m$ retângulos sendo (x_i, y_j) o ponto central do subretângulo que ele indexa, ou o baricentro do subretângulo que ele indexa.

Considere um kernel η_ϵ como definido na primeira seção, como produto de funções características,

$$\eta_\epsilon(s, t) = \frac{1}{4\epsilon^2} \chi_\epsilon(s) \chi_\epsilon(t) \quad (15)$$

multiplicado pelo peso $\frac{1}{4\epsilon^2}$.

Observe que chamando $\chi_{i,j}$ a função característica do interior do subretângulo indexado por (x_i, y_j) é verdade que

$$\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \chi_{i,j} = 1 \quad (q.s.) \quad (16)$$

falhando de valer 1 sobre as fronteiras dos sub-retângulos onde terá o valor 0 porque os subretângulos são abertos. No próximo teorema vai-se ver que esta falha de continuidade é irrelevante.

Teorema 2 (Partição da Unidade) *Partição da unidade de classe \mathcal{C}^0*

A convolução do kernel η_ϵ com a família $\chi_{i,j}$ produz uma partição da unidade de classe \mathcal{C}^0 subordinada a um aberto que contém Ω .

Dem:

Chame

$$f(x, y) = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \chi_{i,j} \quad (17)$$

então

$$f_\epsilon = \eta_\epsilon * f = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \eta_\epsilon * \chi_{i,j} \quad (18)$$

é uma soma de funções contínuas. O suporte de $f_\epsilon = \eta_\epsilon * f$ é o fecho da união dos suportes de $\eta_\epsilon * \chi_{i,j}$ que é uma ϵ vizinhança fechada de Ω portanto o seu interior contém Ω e cada um dos suportes das funções $\eta_\epsilon * \chi_{i,j}$ sendo o aberto ao qual está subordinada a esta partição da unidade.

q.e.d .

Antes de enunciar o próximo teorema, vou definir

Definição 1 (Átomos) da partição da unidade

$$\phi_{i,j,\epsilon} = \eta_\epsilon * \chi_{i,j} \quad (19)$$

O que tem de especial com esta *partição da unidade* é que seus *átomos* são 1-splines a suporte compacto. Como na primeira parte mostrei que tenho kernels com suportes arbitrariamente pequenos

- de medida pequena;
- concentrados em torno da origem - com suporte contido em uma bola de centro de zero com raio arbitrariamente pequeno;
- com classe de diferenciabilidade arbitrária,

então, substituindo η_ϵ por um tal kernel produz o teorema

Teorema 3 (Spline Partição da Unidade) *Partição da unidade de classe \mathcal{C}^{n-1}*

A família de funções $\phi_{i,j,\epsilon} = \eta_\epsilon * \chi_{i,j}$ em que η_ϵ é um kernel- n -spline com suporte centrado na origem é uma partição da unidade de classe \mathcal{C}^n subordinada a um aberto que contém Ω , cujos elementos são n -splines.

Observe que há uma *simplificação de notação* em que estou usando o símbolo η_ϵ com significados distintos. No teorema 3, η_ϵ é um kernel de classe \mathcal{C}^n .

3 Partição da unidade e malha não uniforme

Nesta seção vou mostrar como o trabalho extenso feito em [2], embora uma construção meticulosa, foi inútil porque pode ser obtido o mesmo resultado sem o trabalho artesanal ali desenvolvido, entretanto ficam obfuscas as equações envolvidas, produzindo-se um resultado quase do tipo “teorema de existência” (não o é, porque é computacional). Como na prática este resultado vai ser usado dentro de um programa de computador, o defeito de produzir um resultado

implícito será naturalmente absorvido, uma vez que não precisamos de equações explícitas. Isto ficará inteiramente justificado na sequência do trabalho.

Considere Ω um domínio compacto (com interior não vazio) do \mathbf{R}^2 e uma quasi-partição, por abertos, de Ω

$$(W_i)_{i \in I = \{0, 1, \dots, n-1\}} \quad (20)$$

$$m(\Omega - \bigcup_{i \in I} \partial(W_i)) = 0 \quad (21)$$

$$\chi_i \text{ a função característica de } W_i \quad (22)$$

$$\eta_\epsilon = \frac{1}{m(W_i)} \chi_{W_i} \quad (23)$$

em que m representa a medida de Lebesgue no plano. O conjunto dos pontos da união dos W_i difere de Ω por um conjunto de medida zero, por exemplo a reunião das fronteiras.

Observação 1 (Um exemplo) Família particular $(W_i)_{i \in I}$

Considere um domínio compacto Ω do \mathbf{R}^2 e um retângulo R que o contenha. Como anteriormente considere a ϵ, δ -rede que produz a família $Q_{i,j}; i \in I, j \in J$ de retângulos particionando R e conseqüentemente também Ω . Reenumere a família alfabeticamente para ter

$$Q_{i \in \{0, \dots, n-1\}}$$

e defina

$$W_i = Q_i \cap \Omega$$

Nestas condições

Teorema 4 (Quase) Partição da unidade

$$\chi_i = \chi_{W_i} \quad (24)$$

$$f(x, y) = \sum_{i=0}^{n-1} \chi_i(x, y) = 1 \text{ (q.s.)} \quad (25)$$

sendo uma função quase-contínua, positiva, e nula fora de Ω .

Não é uma partição da unidade porque a soma deixa de ser 1 sobre um conjunto de medida zero, também a continuidade falha sobre o mesmo conjunto de medida zero, a união das fronteiras dos subdomínios W_i .

Teorema 5 (partição C^0) Partição contínua da unidade

Considere um núcleo-1-spline η_ϵ como definido na equação (23), e a família de funções características definidas na (22). Então

$$(\eta_\epsilon * \chi_i)_{i=0}^{n-1} \quad (26)$$

é uma partição contínua da unidade subordinada a um aberto que contém Ω com

$$f_\epsilon(x, y) = \sum_{i=0}^{n-1} \eta_\epsilon \chi_i(x, y) \quad (27)$$

Este é essencialmente o resultado obtido em [2] por meio de uma construção realmente artesanal mas apresentando minuciosamente as equações dos elementos da partição da unidade no caso da *partição contínua da unidade*. Para uma partição da unidade de classe C^n com $n > 0$ o trabalho é muito grande, mas se as fronteiras dos sub-conjuntos W_i tiverem uma definição formal, um programa de computador substitui estas equações e nos interessam algoritmos que possam traduzidos por programas de computador, afinal.

Teorema 6 (partição C^n) *Partição da unidade*

Considere um núcleo- n -spline η_ϵ substituindo a definição da equação (23), por um núcleo potência n -ésima por convolução como foi feito no primeiro parágrafo, e a família de funções características exatamente definidas como na (22). Então

$$(\eta_\epsilon * \chi_i)_{i=0}^{n-1} \quad (28)$$

é uma partição de classe C^n da unidade subordinada a um aberto que contém Ω com

$$f_\epsilon(x, y) = \sum_{i=0}^{n-1} \eta_\epsilon \chi_i(x, y) \quad (29)$$

4 Projotor de interpolação

Vou seguir usando a notação da parte final da seção anterior, em particular as famílias de subdomínios e funções características estão enumeradas com uma sucessão simples de índices obtidos por uma reordenação alfabética dos subretângulos de um retângulo R que contenha o domínio Ω .

A função f_ϵ definida na equação (29) é identicamente 1 sobre um conjunto compacto contido no aberto Ω , sendo zero fora de Ω .

Seja agora uma função $F(x, y)$ de classe C^n definida num aberto que contenha o domínio Ω . Considere os coeficientes

$$\omega_i = F(x_i, y_i) \quad (30)$$

$$(x_i, y_i) \text{ são os baricentros dos subdomínios } W_i \quad (31)$$

e deixe-me modificar a equação (29) usando os coeficientes ω_i para obter

$$F_\epsilon(x, y) = \sum_{i=0}^{n-1} \omega_i \eta_\epsilon \chi_i(x, y) \quad (32)$$

Como esta soma na equação (29) vale 1 sobre o ponto (x_i, y_i) então na equação (32) temos

$$F_\epsilon(x_i, y_i) = F(x_i, y_i) \quad (33)$$

e portanto a função F_ϵ coincide com F sobre os pontos (x_i, y_i) , chamados de *pontos de precisão*.

Demonstrei assim:

Teorema 7 (Projektor de interpolação) *Projetando uma classe de funções*
A equação (32) define um projetor de interpolação de uma certa classe de funções no espaço de n -splines de classe C^n gerados pelos átomos da partição da unidade, $\eta_\epsilon * \chi_i$ construída no teorema 6.

Referências

- [1] Praciano-Pereira, T.
Splines por convolução
Préprints da Sobral Matemática - 2007.09
<http://www.sobralmatematica.org/preprints>
- [2] Praciano-Pereira, T.
An interpolation projector associated to a non uniform partition
Préprints da Sobral Matemática - 2008.08
<http://www.sobralmatematica.org/preprints>
- [3] Praciano-Pereira, T.
Convolução de funções características em \mathbf{R}^2
Préprints da Sobral Matemática - 2011.05
<http://www.sobralmatematica.org/preprints>
- [4] Rudin, W.
Functional Analysis
McGraw-Hill Series in Higher Mathematics - 1975