**UNIVERSIDADE FEDERAL DE MINAS GERAIS**

**Programa de Especialização em Estatística**

# MÉTODO DO NÚCLEO PARA ESTIMAR FUNÇÕES DE DENSIDADE E FUNÇÕES DE REGRESSÃO

Milton Pifano Soares Ferreira

**Belo Horizonte**

**2011**

**Milton Pifano Soares Ferreira**

# MÉTODO DO NÚCLEO PARA ESTIMAR FUNÇÕES DE DENSIDADE E FUNÇÕES DE REGRESSÃO

Dissertação apresentada ao Programa de Especialização em Estatística como requisito para obtenção do título de Especialista em Estatística pela Universidade Federal de Minas Gerais.

Orientador: Gregório Sarávia Atuncar

**Belo Horizonte**

**2011**

## AGRADECIMENTOS

A realização deste trabalho, para complementação do curso de Especialização em Estatística da Universidade Federal de Minas Gerais, em muito se deve à colaboração e apoio de diversas pessoas, às quais transmito os meus agradecimentos:

*Aos meus familiares*:

- Minha esposa, Aglaia e meu filho, Rodrigo.

que me apoiaram, e continuam me apoiando, nesta nova etapa de aprendizado.

*À minha amiga :*

- Cláudia Galinkin.

que me disponibilizou um modelo de projeto LaTeX, agilizando meu aprendizado na ferramenta e, com isso, viabilizando a utilização da mesma para a elaboração deste documento.

*A todo o Departamento de Estatística da UFMG, em especial aos professores :*

- Gregório Sarávia Atuncar.
- Marcelo Azevedo Costa.
- Ela Mercedes Medrano de Toscano.
- Sueli Aparecida Mingoti.
- Glaura da Conceição Franco.
- Roberto da Costa Quinino.
- Frederico Cruz.

que, com todo o conhecimento e experiência, tiveram muita paciência e atenção com todos nós, e transmitiram, com simplicidade e maestria, conteúdo estatístico de elevada qualidade. Destaque para o professor Gregório, que concordou em me orientar no estudo de um assunto que tanto me interessa.

*Às secretárias :*

- Rosiane Araújo Gonçalves.
- Maria Cristina Morandi.
- Márcia Fileto.
- Rogéria Figueiredo.

que, com muito carinho e simpatia, suavizavam nossas tardes servindo um cafezinho super saboroso, acompanhado de guloseimas deliciosas.

*A todos aqueles que direta ou indiretamente viabilizaram a realização do curso de Especialização em Estatística da UFMG.*

## RESUMO

Dentre as atividades iniciais para o entendimento do comportamento de um conjunto de dados estão a geração de um gráfico de dispersão e de um histograma. A análise desses gráficos e o entendimento do assunto no qual os dados estão inseridos são itens importantes de apoio na identificação do método a ser utilizado na estimação da função de regressão e/ou da função densidade de probabilidade. Os métodos de estimação utilizados estão divididos em 2 grandes grupos : paramétricos e não paramétricos. Esse trabalho apresenta o estudo desenvolvido utilizando o método não paramétrico denominado *núcleo-estimador*, que compreendeu o estudo teórico dos conceitos e fórmulas matemáticas envolvidas, mas sobretudo a experimentação do método a partir da utilização de rotinas já desenvolvidas no pacote estatístico R por diversos trabalhos anteriores , por exemplo : (BESSEGATO et al, 2006) , (MIRANDA, 2007) e (SILVA, 2008)



Palavras-chave: núcleo-estimador, função de regressão, função densidade.

## LISTA DE FIGURAS

**SUMÁRIO**

# 1 INTRODUÇÃO

## 1.1 Métodos Paramétricos, Não Paramétricos e Núcleo-estimador

O método não paramétrico utilizando a técnica de núcleo-estimador para estimação da função densidade de probabilidade e da função de regressão é mais facilmente entendido partindo-se inicialmente do problema de regressão simples.

Na regressão simples, assume-se que uma amostra de pares $(X_1,Y_1),...,(X_n,Y_n)$ onde Y é a variável resposta e X a variável preditora, satisfaça a condição :

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \ , \ \ onde\ i = 1,2,\ldots,n. \qquad (1)$$

Os erros $\varepsilon_i$ são variáveis aleatórias simétricas com média 0, variância $\sigma^2$ e não correlacionadas. Entretanto, isso implica que as observações estejam aleatoriamente distribuídas em torno de uma reta, o que muitas vezes não ocorre.

*Métodos Paramétricos* : o modelo linear apresentado em (1) é um exemplo de modelo *paramétrico de regressão*. Para melhor entendermos o termo paramétrico, vamos fixar o valor de $X$. Dessa forma, a componente aleatória $\varepsilon_i$ determina as propriedades de $Y$. Supondo, como dito acima, que $\varepsilon_i$ tenha média 0 e variância $\sigma^2$, então a média da variável resposta $Y$ para qualquer valor da variável regressora $X$ é :

$$E(Y|X=x) = \mu_{x|y} = E(\beta_0 + \beta_1 x + \varepsilon) = \beta_0 + \beta_1 x \ \ , \ \ pois \ \ E(\varepsilon) = 0$$

Podemos reescrever a equação (1) como :

$$y_i = m(x_i) + \varepsilon_i \ , \ \ onde\ i = 1,2,\ldots,n. \qquad (2)$$

Na equação (1) estamos, então, assumindo que a forma da função de regressão $m$ é conhecida com exceção dos valores dos 2 parâmetros $\beta_0$ e $\beta_1$. Daí o termo *paramétrico* uma vez que a família de funções do modelo pode ser especificado por um número finito de parâmetros.

A restrição da função $m$ pertencer à família paramétrica impõe uma rigidez que muitas

vezes pode não ser desejável. A figura abaixo apresenta um exemplo onde a hipótese de que as observações estão aleatoriamente dispersas ao longo de uma reta está claramente longe de ser verdadeira.

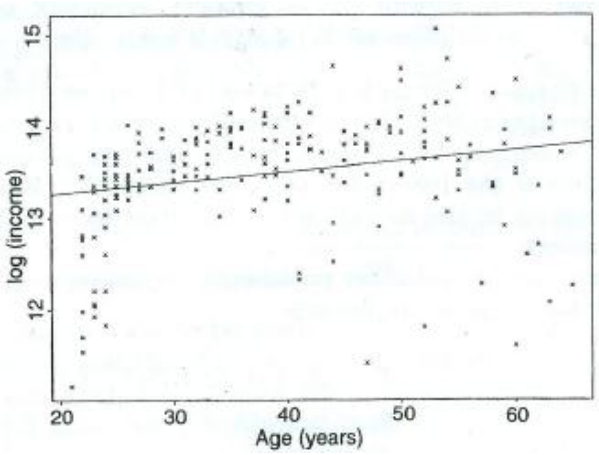


Figura 1: Diagrama de Dispersão Idade/Log(salário) de 205 Trabalhadores Canadenses
(Fonte: Ullah,1985)

Ao se escolher um modelo paramétrico que não é apropriado para um determinado conjunto de dados, há o perigo de se chegar a conclusões incorretas para a análise de regressão.

A rigidez da regressão paramétrica pode ser contornada pela remoção da restrição da função *m* pertencer a uma família paramétrica.

*Métodos Não Paramétricos* : A motivação para utilização de modelos não paramétricos para a regressão é bem objetiva, ou seja, ao se deparar com um gráfico de dispersão que não apresenta de forma clara um modelo funcional simples, tem-se a necessidade de deixar os dados decidirem qual função os descreve melhor sem as restrições impostas por um modelo paramético (algumas vezes isso é referenciado como "deixar os dados falarem por si".)

Vários são os métodos não paramétricos existentes para obter-se uma estimativa de regressão para a função *m*. O presente trabalho descreve o método conhecido como *núcleo-estimador*.

*Núcleo-estimador* : A figura 2 mostra um estimador de *m* para os dados de idade / log(salário) utilizando o que é conhecido como *núcleo-estimador linear local*.

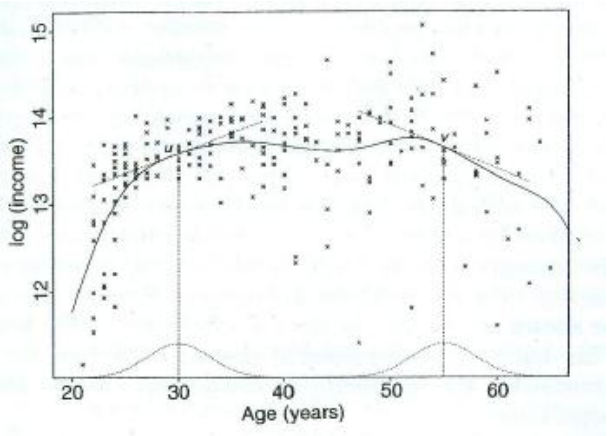


Figura 2:
Núcleo-estimador usando a regressão linear local baseado nos dados de Idade/Log(salário) de 205 Trabalhadores Canadenses.A curva sólida é a estimativa.As curvas pontilhadas são os pesos e a reta ajusta 2 pontos : $u$ e $v$.
(Fonte: Wand and Jones,1995)

A função que aparece na parte de baixo do gráfico é uma função de *núcleo-estimador* que normalmente é definida como uma função densidade de probabilidade simétrica, tal como a distribuição normal. O valor estimado em um determinado ponto é obtido pelo ajuste dos dados a uma reta utilizando mínimos quadrados ponderados, onde os pesos são definidos de acordo com a altura da função do núcleo. Isso significa que os dados mais próximos do ponto considerado tem mais influência no ajuste linear do que aqueles que estão mais distantes. Esses estimadores fazem parte da classe de estimadores de regressão *polinomial local*.

Apesar de termos apresentados, até então, a aplicação do método do núcleo-estimador na regressão não paramétrica, esse método pode também ser aplicado na estimação de funções de densidade de probabilidade.

## 1.2 Histogramas e estimação da densidade

O problema da estimação do função densidade de probabilidade é uma questão fundamental no entendimento do comportamento dos dados. Nesse tópico vamos apresentar um paralelo entre os histogramas e o método não paramétrico do núcleo-estimador.

Suponha que $X_1, ..., X_n$ seja uma amostra aleatória de uma v.a. contínua X com função densidade $f$. O modelo paramétrico de estimação da função $f$ assume que $f$ pertence a uma família paramétrica de distribuições, tal como a família normal ou gama, e, a partir de então, procura-se estimar os parâmetros desconhecidos pelo método da máxima verossimilhança,por exemplo. Por outro lado, um *estimador de densidade não paramétrico* não assume uma forma específica para a função $f$.

O histograma é o estimador de densidade não paramétrico mais antigo e mais utilizado. Ele é normalmente formado pela alocação dos dados reais em intervalos de tamanhos iguais, frequentemente chamados de *bloco* ou *"bins"*. O histograma é, então, uma função degrau (*step function*) sendo a altura a proporção da amostra contida no *bloco* dividido pelo comprimento de todos os *blocos*.

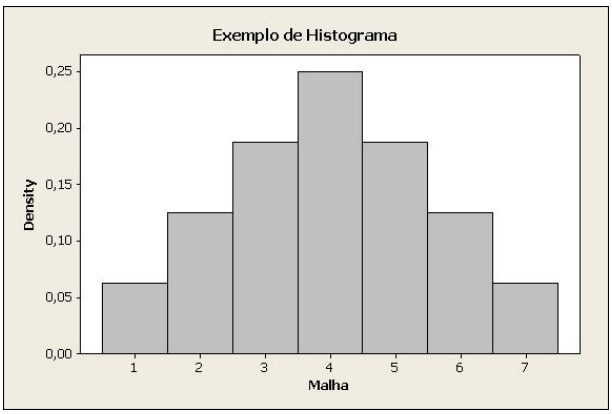


Figura 3: Exemplo de Histograma
(Fonte: Própria, 2011)

Na figura 3, acima, podemos ver um exemplo de histograma totalmente simétrico. A estimativa da função histograma pode assim ser escrita :

$$f_H(x;b) = \frac{número\ de\ observações\ no\ bloco\ que\ contém\ x}{nb}$$

onde :

$n$ = tamanho da amostra

$b$ = largura do bloco, normalmente chamada janela.

Para se construir histogramas, 2 definições devem ser efetuadas a priori : o tamanho do bloco (*binwidth*) e os pontos iniciais e finais da malha.

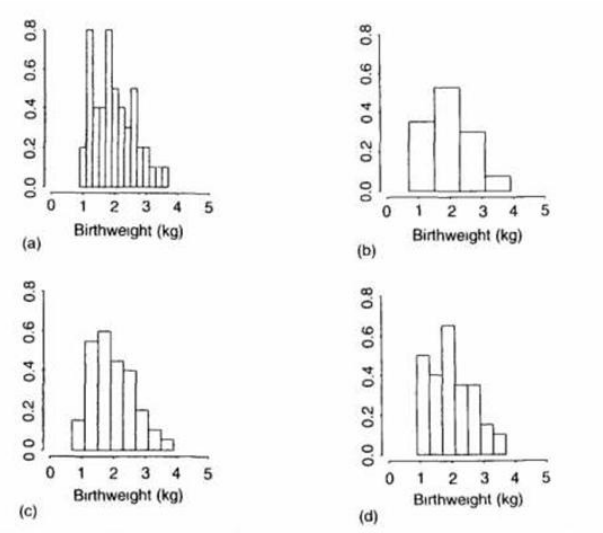


Figura 4: Histogramas com tamanhos de bloco diferentes,(a) e (b); e histogramas com pontos extremos diferentes, mas com blocos de mesmo tamanho, (c) e (d). (Fonte: Wand and Jones,1995)

Os 4 gráficos acima apresentam histogramas baseados no mesmo conjunto de dados. Os gráficos a) e b) são baseados em blocos estreitos e largos, ou seja, b=0.2 e b=0.8 respectivamente. Já os gráficos c) e d) mostra que o posicionamento dos pontos extremos também tem um efeito significativo uma vez que os formatos das densidades sugeridas por esses histogramas são um pouco diferentes.

O tamanho do bloco (*binwidth*) *b* é normalmente chamado de *parâmetro de suavização* uma vez que ele controla o grau de suavidade a ser aplicado aos dados.

A sensibilidade do histograma ao posicionamento dos pontos extremos da malha não ocorre com estimadores de densidade do tipo *núcleo-estimador*. Outro problema que ocorre com os histogramas é o fato dele sempre estimar todas as densidades por uma função degrau (*step function*).

## 1.3 Metodologia

A metodologia utilizada para o desenvolvimento deste trabalho foi baseada na leitura de livros e artigos, discussões com o orientador e experimentação prática a partir do desenvolvimento de rotinas utilizando o pacote R. As rotinas básicas de estimação foram copiadas de trabalhos anteriores, em especial de (BESSEGATO et al, 2006) , (MIRANDA, 2007) e (SILVA, 2008).

# 2 MOTIVAÇÃO E OBJETIVOS

## 2.1 Objetivo Geral

O objetivo geral do desenvolvimento deste trabalho foi o de obter o conhecimento básico de métodos não paramétricos para estimação de funções densidade e de funções de regressão.

## 2.2 Objetivos Específicos

Estudar o método do núcleo-estimador para estimação da função densidade e da função de regressão a partir da utilização prática de rotinas desenvolvidas no pacote estatístico R.

## 2.3 Motivação

A curiosidade por métodos não paramétricos, mesmo sem ainda saber da existência deles, surgiu já no início do curso de Especialização em Estatística, na segunda parte da disciplina "Inferência Estatística e Probabilidades", ministrada pelo professor Gregório S. Atuncar. Durante o aprendizado das diversas distribuições (Normal, Gama, Weibull, Exponencial, Erlang, Poisson, Binomial) muitas vezes surgia o questionamento : "E se nenhuma das premissas ou parte delas fossem satisfeitas, como estimar a função densidade?". Como praticamente todas as disciplinas do curso de Especialização em Estatísticas são baseadas em métodos paramétricos, o trabalho de final de curso foi a oportunidade que vislumbrei para conhecer um pouco de como estimar a função densidade sem estar preso às restrições impostas pelos modelos paramétricos. Ou seja, a minha busca era por algo referenciado por Wand e Jones em (WAND; JONES, 1995) como : *"Letting the data speak for themselves"*, isto é, "Deixar os dados falarem por si!".

# 3 ESTIMAÇÃO DA FUNÇÃO DENSIDADE PELO MÉTODO DO NÚCLEO

## 3.1 Conceituação

A estimação da densidade pelo método não paramétrico é uma importante ferramenta para análise de dados que provê um meio muito efetivo de mostrar a estrutura de um conjunto de dados logo no início da análise. Ele é especialmente efetivo quando os modelos paramétricos padrões não são apropriados. A figura Exp 1-F mostra como uma estrutura bimodal desvendada pelo método do núcleo pode não aparecer quando utilizamos um modelo paramétrico unimodal como a normal. Considerando que um dos principais objetivos da análise de dados é descobrir estruturas importantes nos dados é desejável estar de posse de uma ferramenta que estime a densidade sem assumir que ela tenha uma forma funcional específica.

No decorrer deste capítulo, será assumido que estamos trabalhando com amostras aleatórias $X_1, ..., X_n$ extraídas de uma função densidade $f$ contínua e univariada. Um sinal de integral não qualificado $\int$ deve ser entendido como a integração por todo os números reais, $\mathbb{R}$. A notação $\phi_\sigma = (2\pi\sigma^2)^{-1/2} exp[-x^2/(2\sigma^2)]$ será utilizado para denotar a densidade $N(0, \sigma^2)$.

A fórmula do núcleo-estimador para a função densidade é dada por :

$$\hat{f}(x;h) = (nh)^{-1} \sum_{i=1}^{n} K\{(x - X_i)/h\} \qquad (3)$$

Na fórmula (3), $K$ é uma função que satisfaz a identidade $\int K(x)dx = 1$, que é chamada de *núcleo*, e $h$ é um número positivo, normalmente chamado de *tamanho da janela* ou "*bandwidth*". Uma fórmula mais compacta para o núcleo-estimador pode ser obtida através da introdução da notação $K_h(u) = h^{-1}K(u/h)$. Com isso, pode-se escrever :

$$\hat{f}(x;h) = n^{-1} \sum_{i=1}^{n} K_h(x - X_i).$$

Normalmente escolhe-se $K$ como sendo uma função densidade de probabilidade unimodal que é simétrica com respeito a zero. Isso assegura que $\hat{f}(x;h)$ seja também uma função

densidade. Entretanto, núcleos que não são densidades também podem ser usados. A figura 5 ilustra uma estimativa de densidade pelo método do núcleo construída utilizando 5 observações com a função núcleo definida como uma densidade $N(0,1)$,

$$K(x) = \phi(x) = (2\pi)^{-1/2} e^{-x^2/2}$$

É importante destacar que foram apresentadas apenas 5 observações por motivos didáticos, pois estimativas reais de densidade envolvem normalmente um número bem maior de observações.

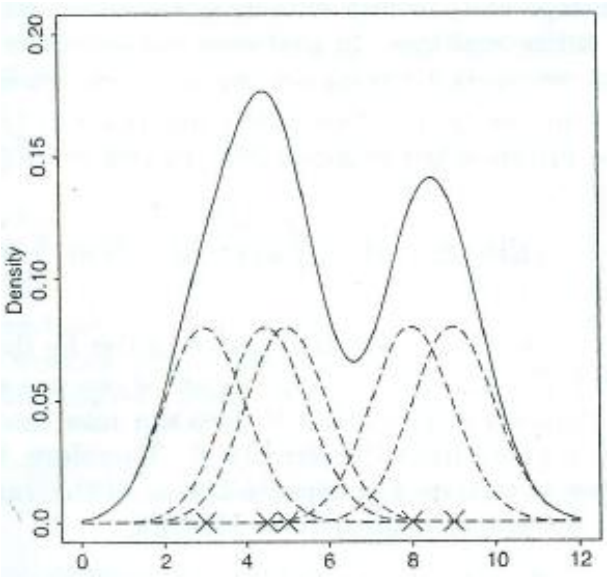


Figura 5: Núcleo-Estimador baseado em 5 observações
(Fonte: Wand and Jones,1995)

Neste caso é possível perceber que $K_h$ é simplesmente a densidade $N(0,h^2)$ onde $h$ assume o papel de fator de escala determinando o nível de espalhamento do núcleo. A estimativa pelo núcleo é construída a partir da centralização do núcleo, baseado em uma escala, para cada observação. O valor de cada estimativa em cada ponto $x$ é simplesmente a média do somatório ponderado (pela função núcleo) das diferenças entre $x$ e todos os demais pontos. A combinação das contribuições de cada ponto significa que nas regiões em que há muitas observações a estimativa do núcleo deve assumir um valor maior assim como é esperado que a densidade real tenha um valor maior. O oposto deve ocorrer em regiões onde há relativamente menos observações.

É importante informar que a escolha da forma da função núcleo não é importante (não é escopo deste trabalho mostrar que essa afirmativa é verdadeira). Entretanto, o valor do tamanho da janela é muito importante. A figura 6 apresenta 3 estimativas de núcleo em uma amostra de tamanho n=1000 para uma determinada função densidade (Fonte: Wand and Jones,1995). A intensidade do alisamento do núcleo efetuado em cada caso é indicado pela escala do núcleo $K_h$

na base de cada gráfico.

Na figura 6(a), com $h = 0,06$, pode ser visto que o estreitamento do núcleo significa que a média do processo gerada em cada ponto é baseada, relativamente, em poucas observações, resultando em uma estimativa rudimentar para $f$. Estimativas com esse formato são chamadas de $sub-alisadas$ ($undersmoothed$).

Na figura 6(b), temos $h = 0,54$, que resultou em uma estimativa mais alisada, mas o alisamento foi super elevado a ponto de fazer desaparecer a bimodalidade da estrutura. Este é um exemplo de superalisamento e estimativas semelhantes são chamadas de *superalisadas* (*supersmoothed*).

Entretanto, na figura 6(c) ocorreu um equilíbrio da estimativa em relação à verdadeira densidade $f$, com $h = 0,18$. Ou seja, os picos da estimativa *subalisada* foram "alisados"e a estrutura bimodal foi reestabelecida, obtendo, com isso, uma estimativa mais próxima da verdadeira densidade.

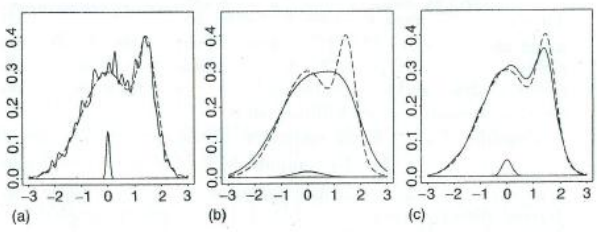


Figura 6: Influência do Tamanho da Janela na Estimativa do Núcleo
(Fonte: Wand and Jones,1995)

## 3.2 Experimento 1

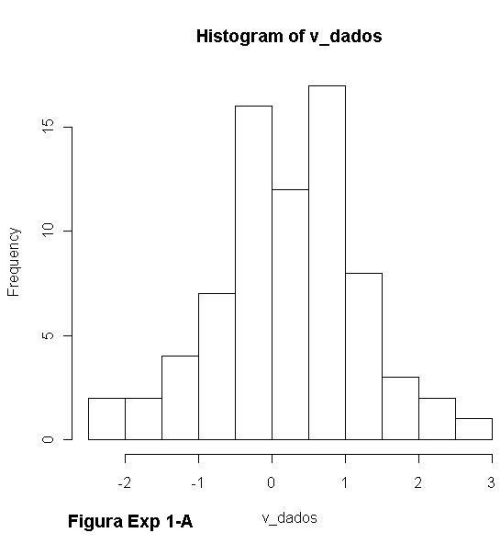


Figura Exp 1-A

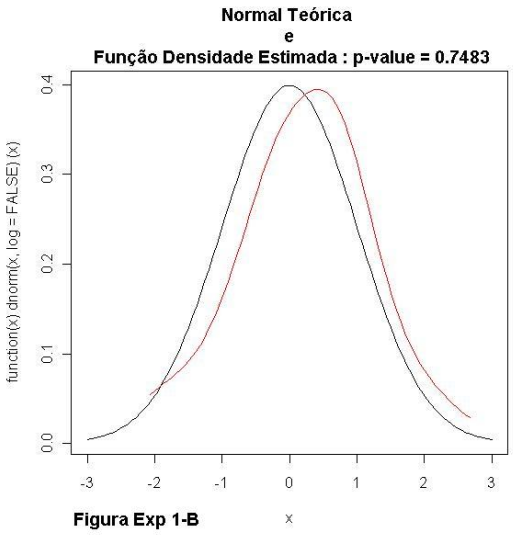


Figura Exp 1-B

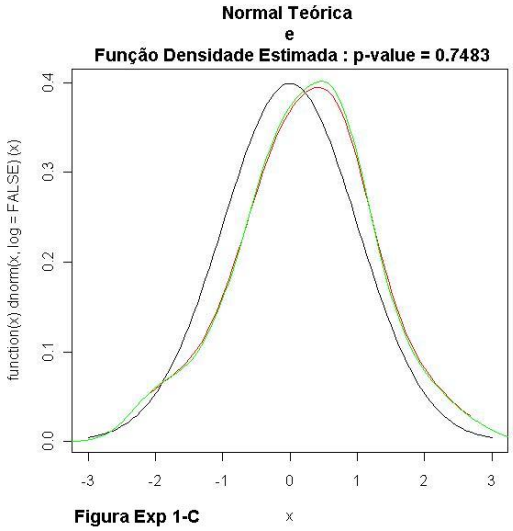


Figura Exp 1-C

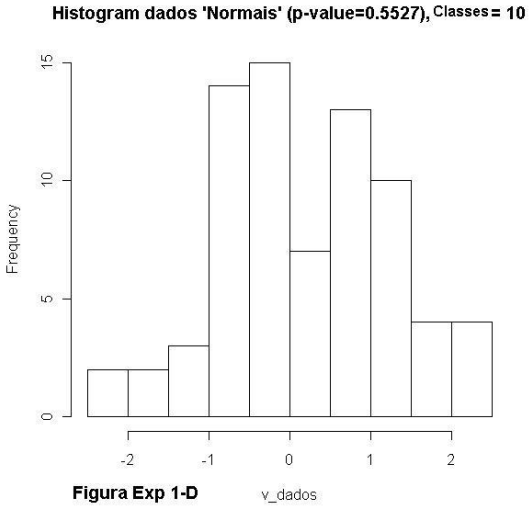


Figura Exp 1-D

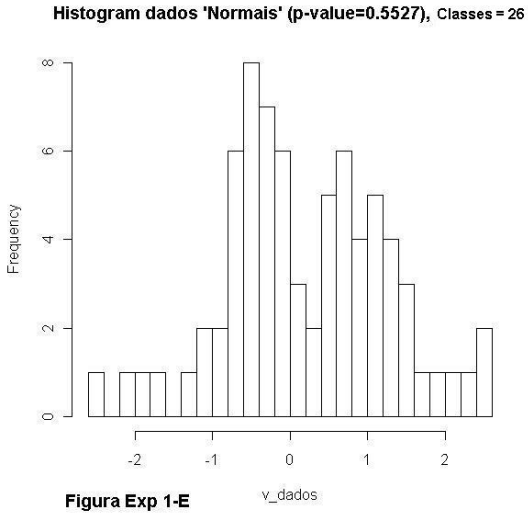


Figura Exp 1-E

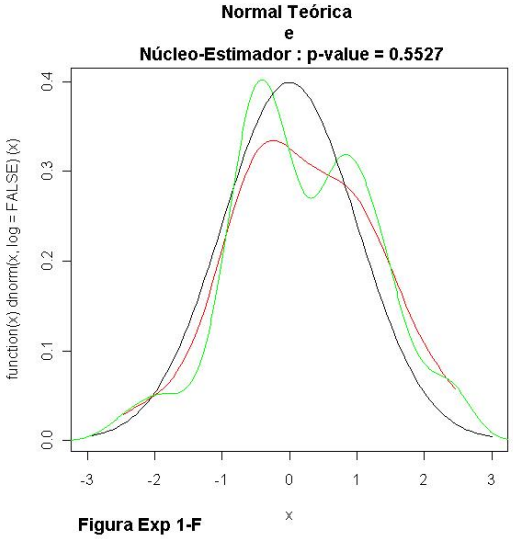


Figura Exp 1-F

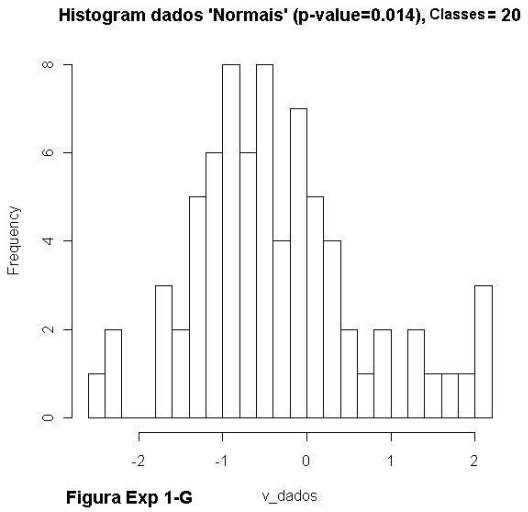


Figura Exp 1-G

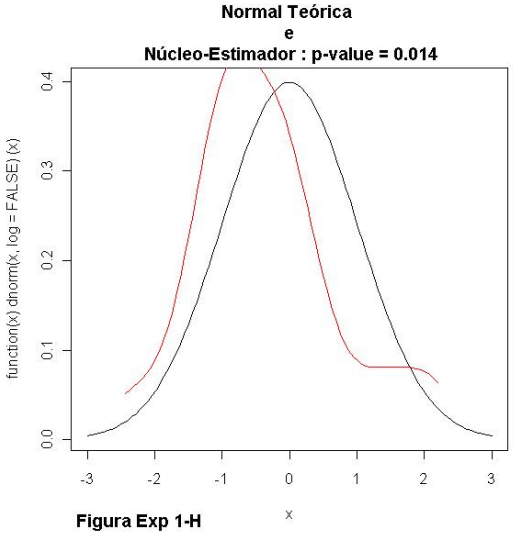


Figura Exp 1-H

O Experimento 1 faz uma análise entre a função densidade estimada pelo método do núcleo a partir de 2 amostras aleatórias normal padrão N(0,1) com p-value igual a 0.7483 e p-value igual a 0.5527, e a normal teórica.

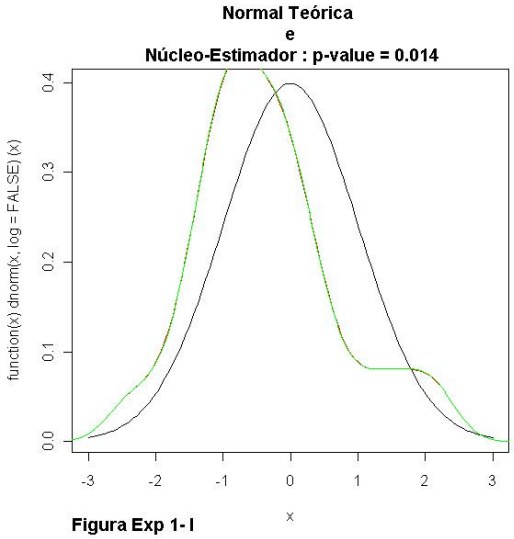


**Figura Exp 1- I**

*Figuras Exp 1-A, Exp 1-D, Exp 1-E e Exp 1-G:* Histogramas

*Figuras Exp 1-B, Exp 1-C, Exp 1-F, Exp 1-H e Exp 1-I :*

Curva Preta => Normal Teórica

Curva Vermelha => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas descritas em (BESSEGATO et al, 2006).

Curva Verde => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas constantes no pacote R (método de Sheather e Jones (SHEATHER,JONES, 1995)).

*Análise figuras Exp 1-A, Exp 1-B e Exp 1-C :*

O confronto do gráfico gerado pelo método do núcleo-estimador e o histograma nos permite verificar que o gráfico do método do núcleo-estimador reflete muito bem o comportamento dos dados. Isso pode ser visto pelo tombamento da curva para o lado direito onde, pelo histograma, vemos que existe uma concentração de dados próximos a 1. Ocorrendo, com isso, um deslocamento da média para a direita de 0. Efetuando o teste de normalidade de Shapiro-Wilk, vemos que o p-value = 0.7483 é bem significativo o que é confirmado pelo alto grau de simetria dos Figuras.

*Análise figuras Exp 1-D, Exp 1-E , Exp 1-F, Exp 1-G, Exp 1-H e Exp 1-I :*

As figuras Exp 1-D, Exp 1-E e Exp 1-F mostram que, mesmo com um p-value elevado (0.5527), os dados podem não ser simétricos tanto quanto esperado e a estimativa pelo núcleo mostra com destaque como que eles não são simétricos. Interessante, também, é compararmos esses gráficos com os das figuras Exp 1-G, Exp 1-H e Exp 1-I, que foram construídos a partir

de uma amostra aleatória N(0,1), mas com p-value igual a 0.014 (rejeita-se a hipótese de normalidade). Ou seja, os gráficos G,H e I apresentam dados com maior grau de simetria que os gráficos D,E e F.

Destaque para o comportamento das rotinas em R onde, para o conjunto de dados utilizado em Exp 1-I, as funções obtiveram praticamente a mesma estimativa, ao contrário do que ocorreu em Exp 1-F, onde o método Sheather e Jones foi mais sensível à variabilidade dos dados e apresentou um resultado mais próximo do histograma.

*Dados da Amostra Exp 1-A/B/C :*

mínimo : -2.067705 ; máximo : 2.676397

Shapiro-Wilk normality test : W = 0.9886, p-value = 0.7483

*Dados da Amostra Exp 1-D/E/F :*

mínimo : -2.467991 ; máximo : 2.461741 ; média : 0.2033009 ; desvio padrão : 1.052206

Shapiro-Wilk normality test : W = 0.9854, p-value = 0.5527

*Dados da Amostra Exp 1-G/H/I :*

mínimo : -2.435902 ; máximo : 2.199226 ; média : 0.2033009 ; desvio padrão : 1.052206

Shapiro-Wilk normality test : W = 0.9575, p-value = 0.014

## 3.3 Experimento 2

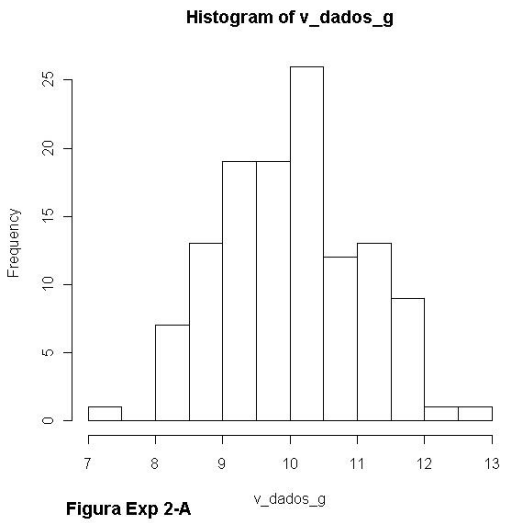


Figura Exp 2-A

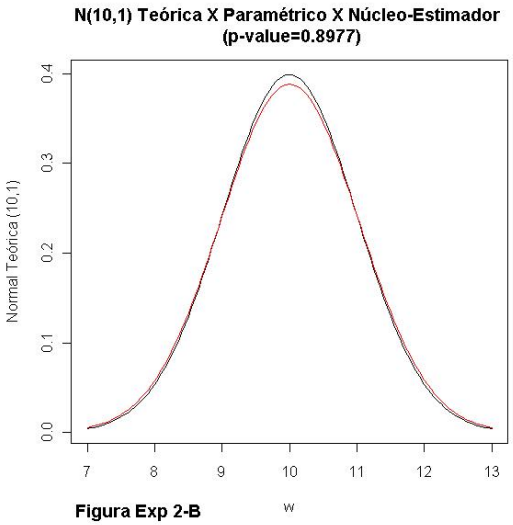


Figura Exp 2-B

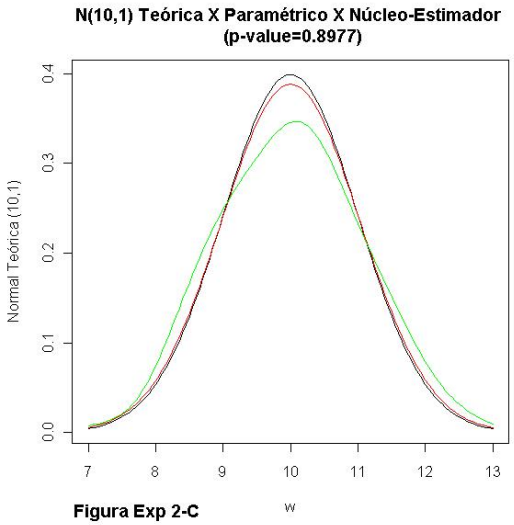


**Figura Exp 2-C**

O Experimento 2 faz uma análise entre a função densidade estimada pelo método do núcleo a partir de uma amostra aleatória N(10,1) com p-value igual a 0.8978, a normal teórica e a normal paramétrica estimada.

*Figura Exp 2-A :* Histograma

*Figuras Exp 2-B e Exp 2-C :*

Curva Preta => Normal Teórica (10,1)

Curva Vermelha => Normal Paramétrica Estimada

Curva Verde => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas descritas em (BESSEGATO et al, 2006).

Analisando os gráficos podemos ver claramente como a função estimada pelo método do núcleo é mais realista na descrição dos dados. O método paramétrico não detectou a maior distribuição dos dados à esquerda do 10. Fato, esse, apresentado pelo núcleo-estimador a partir de uma suspensão da curva à esquerda do 10.

Interessante destacar que o p-value dos dados do experimento 2 é maior do que o do experimento 1, no entanto, os dados apresentam menos simetria !

*Dados da Amostra Exp 2-A/B/C :*

mínimo : 7.18514 ; máximo : 12.6136 ; média : 10.00237 ; desvio padrão : 1.028019

Shapiro-Wilk normality test : W = 0.9941, p-value = 0.8978

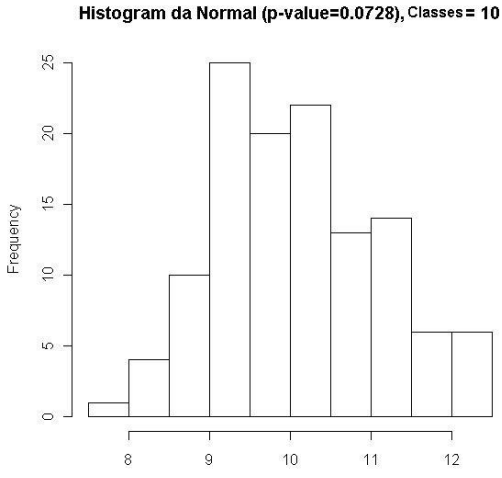


Figura Exp 3-A

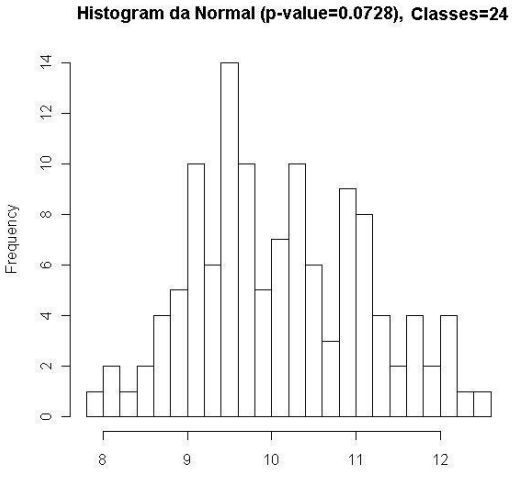


Figura Exp 3-B

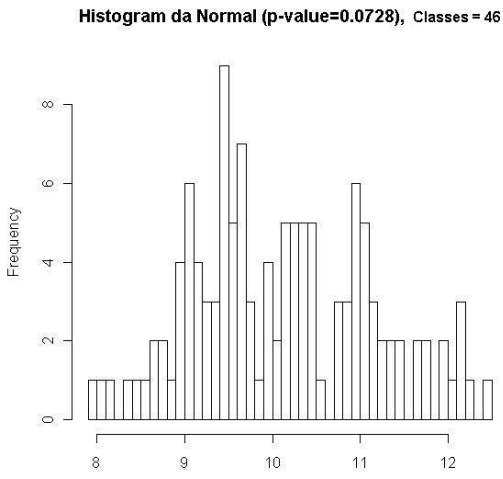


Figura Exp 3-C

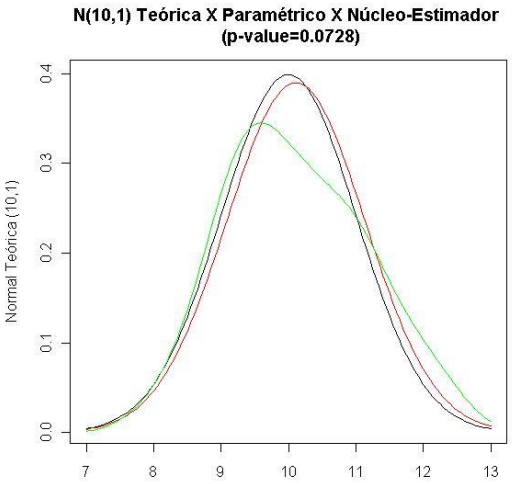


Figura Exp 3-D

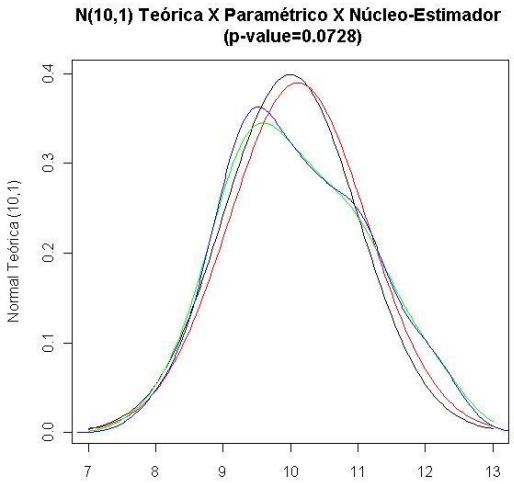


Figura Exp 3-E

## 3.4 Experimento 3

O Experimento 3 faz uma análise entre a função densidade estimada pelo método do núcleo a partir de uma amostra aleatória normal N(10,1) com p-value igual a 0.0728, a normal

teórica e a normal paramétrica estimada :

*Figura Exp 3-A :* Histograma com 10 classes.

*Figura Exp 3-B :* Histograma com 24 classes.

*Figura Exp 3-C :* Histograma com 46 classes.

É interessante notar como o histograma é sensível ao tamanho da janela e a medida que aumentamos o número de classes (diminuímos o tamanho das janelas) mais visível é a assimetria dos dados.

*Figuras Exp 3-D e Exp 3-E :*

Curva Preta => Normal Teórica (10,1)

Curva Vermelha => Normal Paramétrica Estimada

Curva Verde => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas descritas em (BESSEGATO et al, 2006).

Curva Azul => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas constantes no pacote R (método de Sheather e Jones (SHEATHER,JONES, 1995)).

A função estimada pelo método do núcleo mostra como que os dados não são simétricos e a concentração de dados em torno do número 11. Já a estimação pela normal paramétrica estimada não mostra nenhum indício de não simetria. Ocorre, apenas, um ligeiro deslocamento da média.

*Dados da Amostra Exp 3-A/B/C/D/E :*

mínimo : 7.975464 ; máximo : 12.43843 ; média : 10.10871 ; desvio padrão : 1.023901

Shapiro-Wilk normality test : W = 0.9802, p-value = 0.0728

## 3.5 Experimento 4

O Experimento 4 faz uma análise entre a função densidade estimada pelo método do núcleo a partir de uma amostra aleatória weibull W(2,5) e a Weibull teórica.

*Figuras Exp 4-A e Exp 4-B :* Histogramas

*Figuras Exp 4-C e Exp 4-D :*

Curva Preta => Weibull Teórica (2,5)

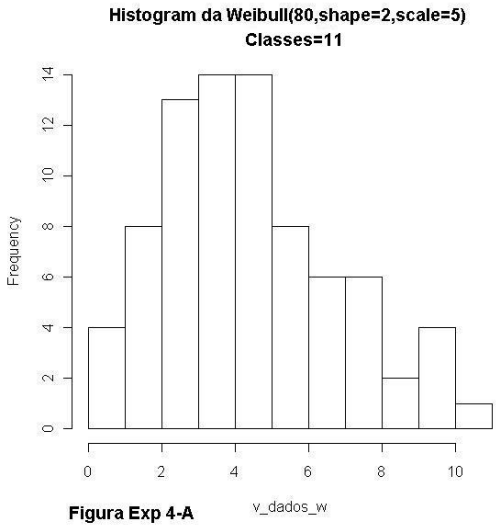


Figura Exp 4-A

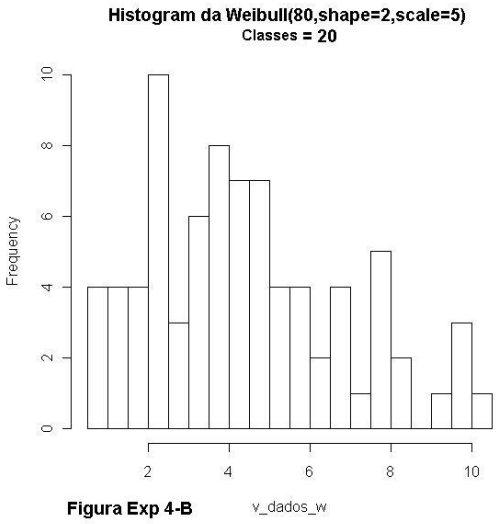


Figura Exp 4-B

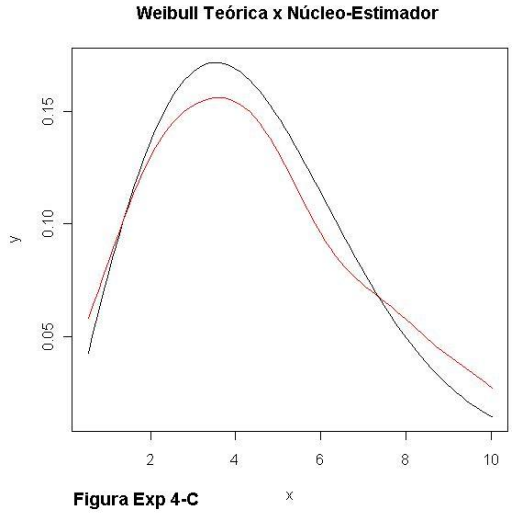


Figura Exp 4-C

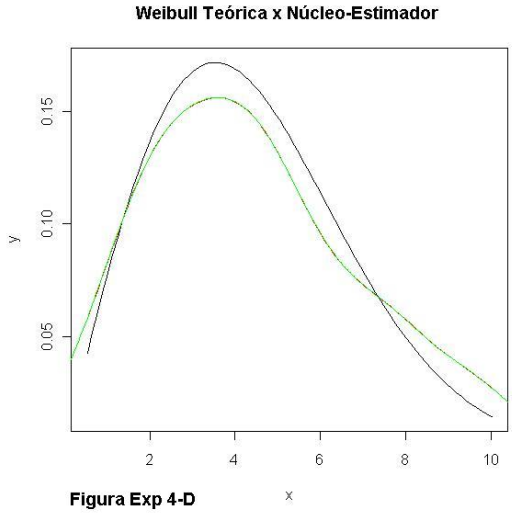


Figura Exp 4-D

Curva Vermelha => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas descritas em (BESSEGATO et al, 2006).

Curva Verde => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas constantes no pacote R (Sheather e Jones (SHEATHER,JONES, 1995)).

Analisando os gráficos podemos ver como a função estimada pelo método do núcleo é sensível ao conjunto de dados em torno do número 8, fazendo com que o decaimento da curva seja mais suave do que o decaimento da Weibull teórica.

O teste de hipótese para verificar se a distribuição é Weibull obteve o seguinte resultado:

One-sample Kolmogorov-Smirnov test : D = 0.0592, p-value = 0.9259 ; alternative hypothesis: two-sided

## 3.6 Aplicação 5

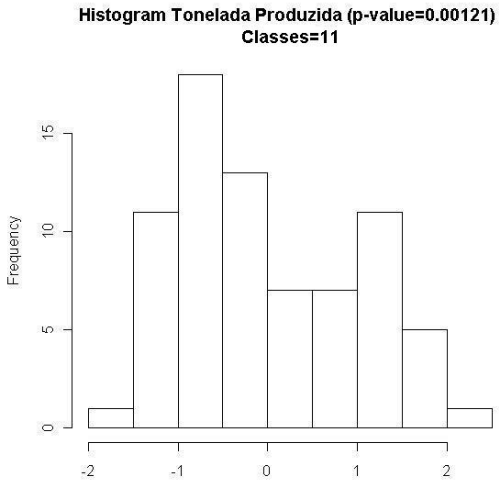


Figura Apl 5-A

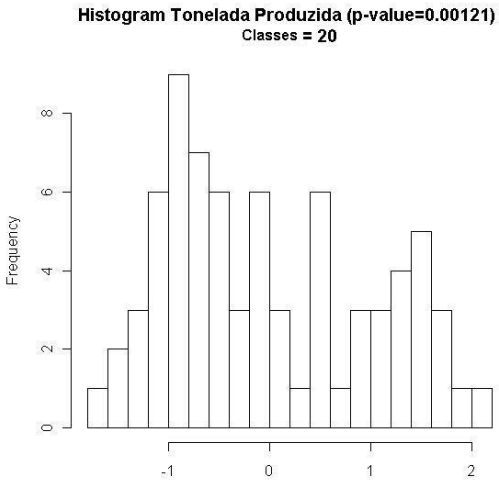


Figura Apl 5-B

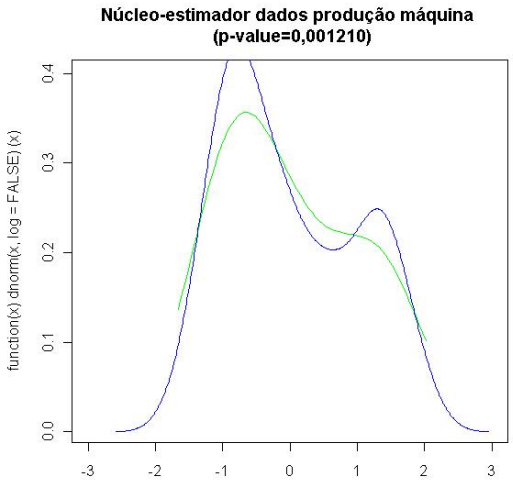


Figura Apl 5-C

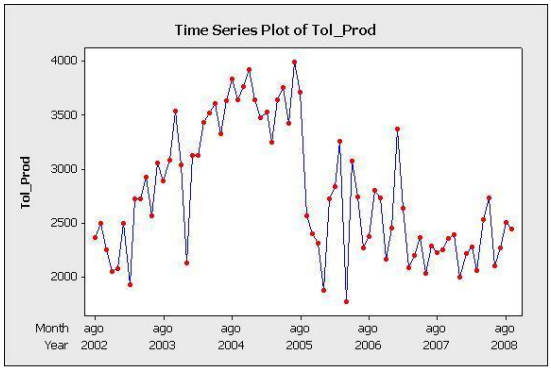


Figura Apl 5-D

A Aplicação 5 faz uma análise da função densidade estimada pelo método do núcleo a partir de dados reais de produção de um determinado produto siderúrgico por uma determinada máquina durante o período de agosto/2002 a setembro/2008 que se mostrou não ser normal com p-value igual a 0.00121 [1] :

*Figuras Apl 5-A e Apl 5-B :* Histogramas com classes igual a 10 e 20, respectivamente.

Para efetuarmos as análises, padronizamos os dados subtraindo a média e dividindo pelo desvio padrão.

*Figura Apl 5-C :*

[1] Por questões sigilosas a fonte dos dados não pode ser revelada.

Curva Verde => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas descritas em (BESSEGATO et al, 2006).

Curva Azul => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas constantes no pacote R (método de Sheather e Jones (SHEATHER,JONES, 1995)).

*Figura Apl 5-D :*

Diagrama da Série Temporal de produção em toneladas de produto siderúrgico no período de agosto/2002 a setembro/2008.

Comparando os histogramas e as estimativas pelo método do núcleo vemos que o método do núcleo mais uma vez apresenta uma estimativa bem ajustada.

*Dados da Amostra Apl 5-A/B/C/D :*

mínimo : -1.665510 ; máximo : 2.018143 ; média : -1.481225e-16 ; desvio padrão : 1

Shapiro-Wilk normality test : W = 0.9376, p-value = 0.001210

## 3.7 Aplicação 6

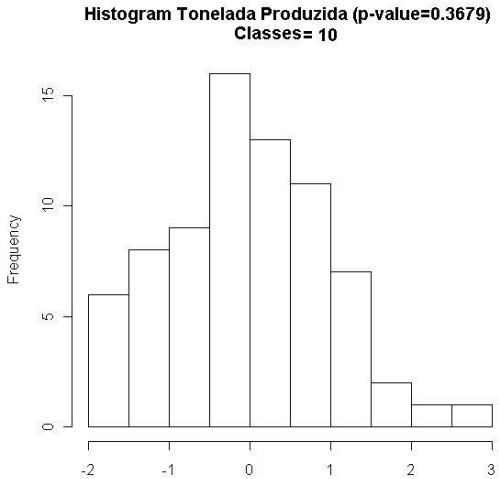


**Figura Apl 6-A**

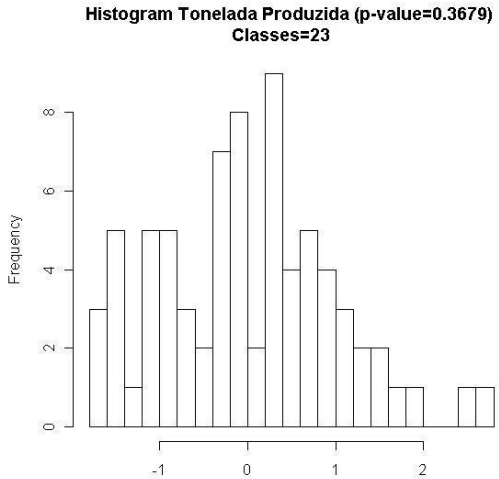


**Figura Apl 6-B**

A Aplicação 6 é equivalente à aplicação 5. A diferença está no conjunto de dados que, neste caso, são normais (p-value = 0,3679 ). Os dados são referentes à produção de um determinado produto siderúrgico em uma determinada operação do processo produtivo no período de agosto/2002 a setembro/2008 :

*Figuras Apl 6-A e Apl 6- B :* Histogramas com classes iguais a 10 e 20, respectivamente.

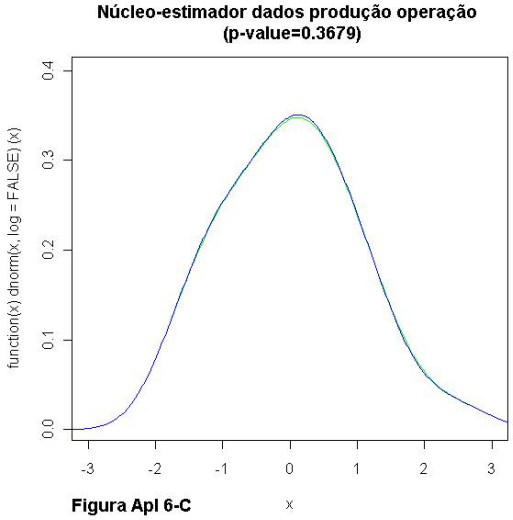


**Figura Apl 6-C**

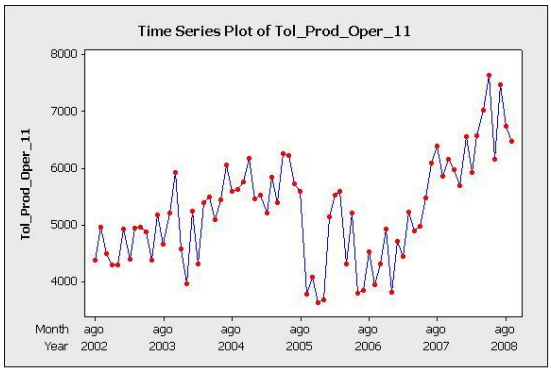


**Figura Apl 6-D**

Para efetuarmos as análises padronizamos os dados subtraindo a média e dividindo pelo desvio padrão.

*Figura Apl 6-C :*

Curva Verde => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas descritas em (BESSEGATO et al, 2006).

Curva Azul => Função densidade de probabilidade estimada pelo método do núcleo utilizando as rotinas constantes no pacote R (método de Sheather e Jones (SHEATHER,JONES, 1995)).

*Figura Apl 6-D :* Diagrama da Série Temporal de produção em toneladas de produto siderúrgico, em uma determinada operação do processo produtivo, no período de agosto/2002 a setembro/2008.

Comparando os histogramas e as estimativas pelo método do núcleo vemos que o método do núcleo mais uma vez apresenta uma estimativa bem ajustada.

*Dados da Amostra Apl 6-A/B/C/D :*

mínimo : -1.789730 ; máximo : 2.633549 ; média : 1.864828e-16 ; desvio padrão : 1

Shapiro-Wilk normality test : W = 0.9819, p-value = 0.3679

# 4 ESTIMAÇÃO DA FUNÇÃO DE REGRESSÃO PELO MÉTODO DO NÚCLEO

## 4.1 Conceituação

Atualmente existem várias técnicas de regressão não paramétrica. As mais populares são aquelas baseadas em funções de núcleo, funções *splines* e *wavelets*. Para cada uma delas, existem outras técnicas específicas. No contexto da regressão pelo núcleo, técnicas tradicionais incluem o *estimador Nadaraya-Watson*(Nadaraya, 1964, (WATSON, 1964)) e outras alternativas (Priestley e Chao,1972, Gasser e Müller,1979).

Neste item apresentaremos uma classe de estimadores de núcleo para regressão chamados de *estimadores de núcleo por polinômios locais* (Stone,1977,Cleveland, 1979,Müller, 1987, Fan, 1992a). Eles estimam a função de regressão em um ponto específico através do ajuste "local"dos dados a um polinômio de grau $p$ pelo método dos mínimos quadrados ponderados. Essa classe inclui, como um caso especial, o estimador de Nadaraya-Watson uma vez que pode-se mostrar que ele corresponde ao ajuste de polinômios de grau zero, ou seja, constantes locais. De particular importância e simplicidade é o *núcleo-estimador linear local*, correspondente ao polinômio de grau 1 (p=1). O *núcleo-estimador linear local* também possui similaridades com os estimadores de núcleo tradicionais citados anteriormente, embora ele tenha propriedades assintóticas e comportamento nas extremidades mais favoráveis se comparado com os demais. Adicionalmente será visto que as propriedades do erro quadrático médio do *núcleo-estimador linear local* é análogo àqueles do núcleo-estimador de densidade. Isso significa que a maioria das idéias desenvolvidas no contexto da estimação da densidade são aplicáveis no contexto da regressão.

A regressão não paramétrica é analisada em dois contextos : *fixo e aleatório*. No contexto univariado fixo o modelo consiste de $x_1, ..., x_n$ que são números não aleatórios ordenados. Um modelo fixo *espaçado igualitariamente* é aquele no qual $x_{i+1} - x_i$ é constante para todo $i$. Para modelos fixos assume-se que as variáveis resposta satisfazem :

$$Y_i = m(x_i) + v^{1/2}(x_i)\varepsilon_i, \qquad i = 1, ..., n$$

onde $\varepsilon_1, ..., \varepsilon_n$ são v.a. independentes para as quais

$$E(\varepsilon_i) = 0 \quad e \quad Var(\varepsilon_i) = 1$$

Chama-se $m$ de *função de regressão média* ou simplesmente *função de regressão*, uma vez que $E(Y_i) = m(x_i)$, equanto $v$ é chamada de *função variância*, uma vez que $Var(Y_i) = v(x_i)$. Com frequência assume-se que $v(x_i) = \sigma^2$ para todo $i$, e nesse caso o modelo é considerado como sendo *homocedástico*. Caso contrário, o modelo é *heterocedástico*.

O modelo de regressão aleatório surge quando observa-se uma amostra bivariada $(X_1, Y_1), ..., (X_n, Y_n)$ de pares aleatórios onde o modelo pode ser escrito como :

$$Y_i = m(X_i) + v^{1/2}(X_i)\varepsilon_i, \qquad i = 1, ..., n$$

sendo condicional em $X_1, ..., X_n$, e $\varepsilon_i$ são v.a. independentes com média zero e variância unitária. Entretanto, no contexto do modelo aleatório

$$m(x) = E(Y|X = x) \quad e \quad v(x) = Var(Y|X = x)$$

são, respectivamente, a média condicional e a variância condicional de $Y$ dado $X = x$.

Na figura 7, abaixo, é apresentado um exemplo que mostra como o núcleo-estimador linear local é construído. A função de regressão verdadeira é

$$m(x) = 2^{\left[\frac{x^2}{(0.3)^2}\right]} + 3^{\left[\frac{(x-1)^2}{(0.7)^2}\right]} \qquad (3)$$

compreendida no intervalo [0,1] e é representada pela curva tracejada. As ordenadas $Y_1, ..., Y_n$ foram geradas utilizando

$$Y_i = m(x_i) + 0.075\varepsilon_i, \qquad i = 1, ..., 100 \qquad (4)$$

onde $x_i = i/100$, e $\varepsilon_i$ são v.a. independentes N(0,1). Os pares $(x_i, Y_i)$ são representados por *.

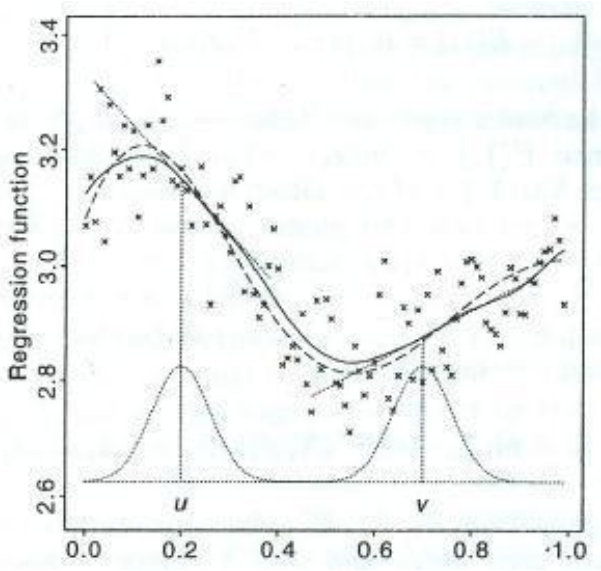


Figura 7: Núcleo-estimador linear local - curva sólida - da função de regressão dada em (3) baseada em 100 simulações (4) - representada pelos x's. A curva tracejada é a verdadeira função m. As curvas pontilhadas são os núcleos ponderados e os ajustes lineares nos pontos u e v.
(Fonte: Wand and Jones,1995)

Em cada um dos pontos (u e v), a reta pontilhada foi obtida através do ajuste de uma reta a $Y_i$ utilizando mínimos quadrados ponderados onde os pesos são escolhidos de acordo com a altura da função de núcleo centralizado no ponto, o que é mostrado pela curva pontilhada na base da figura 7. Quando o processo de ajuste local é efetuado em cada ponto $x \ \varepsilon \ [0,1]$ o resultado é a curva sólida. Se $K_h$ for uma função de núcleo com um tamanho de janela $h$ então, para a estimativa de um determinado $x$, o peso atribuído a um ponto particular $Y_i$ é $K_h(x_i - x)$. Dado a forma usual do núcleo isso significa que aquelas observações próximas de $x$ tem maior influência na estimativa da regressão em $x$ do que aquelas que estão distantes. O montante dessa influência relativa é controlado pelo tamanho da janela $h$ que exerce papel análogo àquele do núcleo-estimador da densidade. Se $h$ é pequeno o processo de ajuste linear depente muito das observações que estão mais próximas de $x$ e tendem a gerar uma estimativa equivalente à uma curva ligando os pontos. Isso é mostrado na figura 8(a) onde um tamanho de janela muito pequeno foi utilizado. Por outro lado, tamanhos de janela $h$ grandes tendem a ponderar as observações de forma igualitária e a medida que $h$ cresce a estimativa tende a uma reta. A estimativa na figura 8(b) foi obtida a partir de um tamanho de janela muito grande.

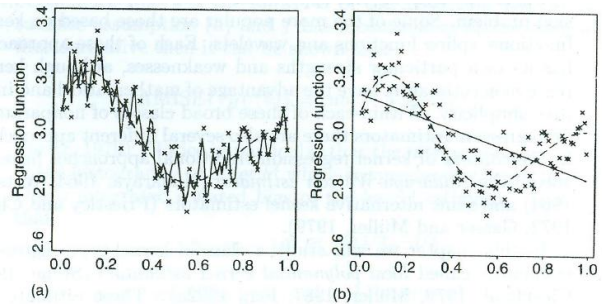


Figura 8: Núcleo-estimador linear local baseado no mesmo conjunto de dados da figura 7, mas com um tamanho de janela muito pequeno (a) e um tamanho de janela muito grande (b).
(Fonte: Wand and Jones,1995)

Uma extensão natural do núcleo-estimador linear local é aquele que ajusta, localmente, polinômios de maior grau. A figura 9 mostra um ajuste local cúbico. Note que os picos e vales de $m$ são melhor estimados pelo ajuste local cúbico uma vez que têm mais graus de liberdade nas regiões de curvatura mais elevada do que as retas, embora essa estimativa seja computacionalmente mais complexa e sofre mais influência de um grau maior da variabilidade da amostra.

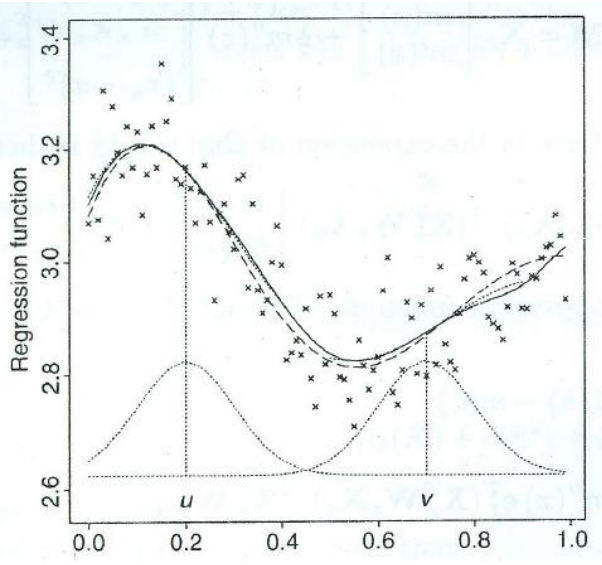


Figura 9: Núcleo-estimador cúbico local - curva sólida - da função de regressão dada em (3) baseada em 100 simulações (4) - representada pelos x's. A curva tracejada é a verdadeira função m. As curvas pontilhadas são os núcleos ponderados e os ajustes lineares nos pontos u e v.
(Fonte: Wand and Jones,1995)

*Expressão para o núcleo-estimador polinomial local* : Apresentamos, a seguir, a derivação da expressão para o núcleo-estimador polinomial local para o contexto *fixo*. Para o contexto *aleatório* a expressão é a mesma, sendo necessário apenas substituir $x_i$ por $X_i$.

Seja $p$ o grau de um polinômio a ser ajustado. No ponto $x$ o estimador $\hat{m}(x;p,h)$ é obtido

a partir do ajuste do polinômio

$$\beta_0 + \beta_1(.-x) + ... + \beta_p(.-x)^p$$

ao par $(x_i, Yi)$ usando mínimos quadrados ponderados com os pesos sendo a função núcleo $K_h(x_i - x)$. O valor de $\hat{m}(x; p, h)$ é o ponto do ajuste $\hat{\beta}_0$ onde $\hat{\beta} = (\hat{\beta}_0, ..., \hat{\beta}_p)^T$ minimiza

$$\sum_{i=1}^{n} \{Y_i - \beta_0 - ... - \beta_p(x_i - x)^p\}^2 K_h(x_i - x).$$

Assumindo a invertibilidade de $\mathbf{X}_x^T \mathbf{W}_x \mathbf{X}_x$, a teoria padrão dos mínimos quadrados ponderados conduz à solução

$$\hat{\beta} = (\mathbf{X}_x^T \mathbf{W}_x \mathbf{X}_x)^{-1} \mathbf{X}_x^T \mathbf{W}_x \mathbf{Y}$$

onde $\mathbf{Y} = (Y_1, ..., Y_n)^T$ é o vetor das variáveis resposta,

$$\mathbf{X}_x = \begin{bmatrix} 1 & x_1 - x & \cdots & (x_1 - x)^p \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n - x & \cdots & (x_n - x)^p \end{bmatrix}$$

é uma matrix $n \times (p+1)$ e

$$\mathbf{W}_x = \text{diag}\{K_h(x_1 - x), ..., K_h(x_n - x)\}$$

é uma matrix diagonal $n \times n$ de pesos. Uma vez que o estimador $m(x)$ é o coeficiente do intercepto obtemos

$$\hat{m}(x; p, h) = \mathbf{e}_1^T (\mathbf{X}_x^T \mathbf{W}_x \mathbf{X}_x)^{-1} \mathbf{X}_x^T \mathbf{W}_x \mathbf{Y}$$

onde $\mathbf{e}_1$ é o vetor $(p+1) \times 1$ com 1 na primeira entrada e 0 nas demais. Para $p = 0$ existe uma fórmula simples para o estimador *Nadaraya-Watson*

$$\hat{m}(x; 0, h) = \frac{\sum_{i=1}^{n} K_h(x_i - x) Y_i}{\sum_{i=1}^{n} K_h(x_i - x)} \qquad (5)$$

e o estimador linear local $(p = 1)$ :

$$\hat{m}(x; 1, h) = n^{-1} \sum_{i=1}^{n} \frac{\{\hat{s}_2(x; h) - \hat{s}_1(x; h)(x_i - x)\} K_h(x_i - x) Y_i}{\hat{s}_2(x; h)\hat{s}_0(x; h) - \hat{s}_1(x; h)^2} \qquad 6)$$

onde

$$\hat{s}_r(x; h) = n^{-1} \sum_{i=1}^{n} (x_i - x)^r K_h(x_i - x)$$

Pelo fato da função núcleo $K$ ser simétrica, pode-se escrever $K_h(x - x_i)$ ao invés de $K_h(x_i - x)$, como é para o núcleo-estimador da densidade. A notação $K_h(x_i - x)$ enfatiza o fato

de que o núcleo-estimador linear local é uma regressão ponderada nos dados, centralizada em $x$.

## 4.2 Experimento 7

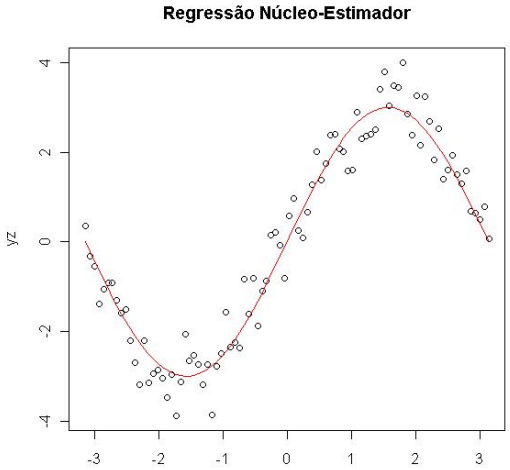


Figura Exp 7-A

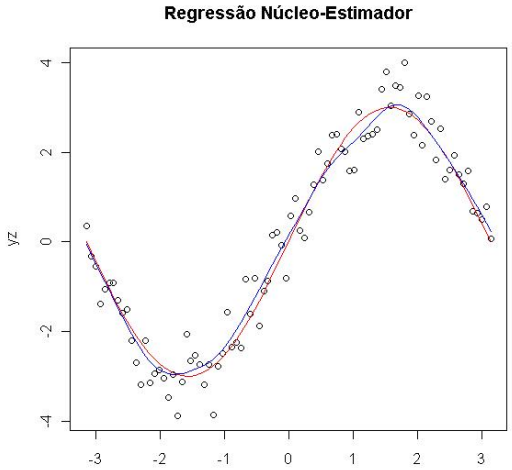


Figura Exp 7-B

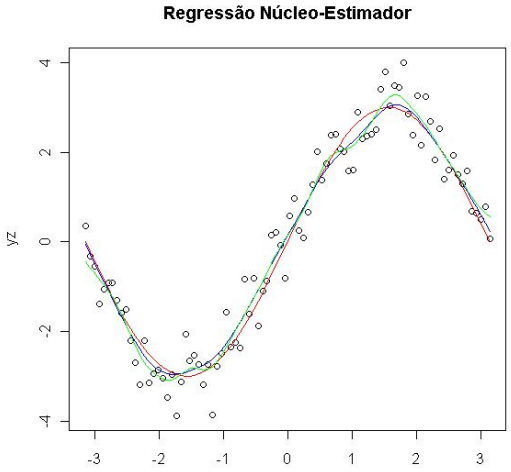


Figura 7-C

O Experimento 7 faz uma análise entre a função de regressão estimada pelo método do núcleo e a função $3 * sen(x)$ acrescida de um erro, onde o erro tem uma distribuição normal padrão N(0,1), conforme a seguir :

$$f(x) = 3 * sen(x) + \varepsilon, \quad onde \quad \varepsilon \quad N(0,1)$$

*Figura Exp 7-A :* Gráfico de dispersão de $f(x)$ e a função $3 * sen(x)$.

*Figuras Exp 7-B e Exp 7-C :*

Curva Vermelha => função $3 * sen(x)$

Curva Azul => Função de Regressão estimada pelo método do núcleo utilizando as rotinas descritas em (BESSEGATO et al, 2006) (estimação gerada no estágio 2).

Curva Verde => Função de Regressão estimada pelo método do núcleo utilizando a rotina *ksmooth* constante no pacote R.

Podemos ver que a estimação pelo método do núcleo é bem próxima da função $3 * sen(x)$ , sendo sensível aos erros acrescidos.

## 4.3 Aplicação 8

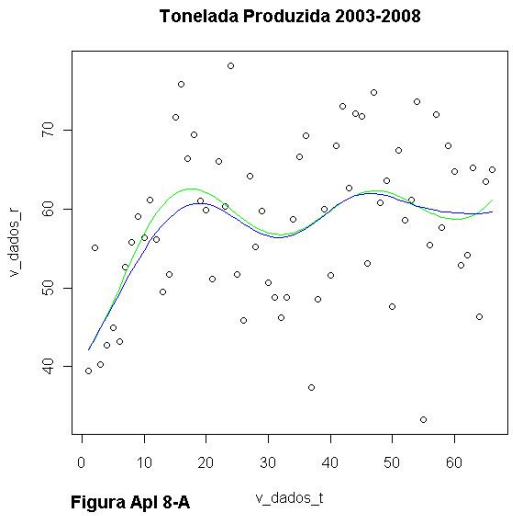


Figura Apl 8-A

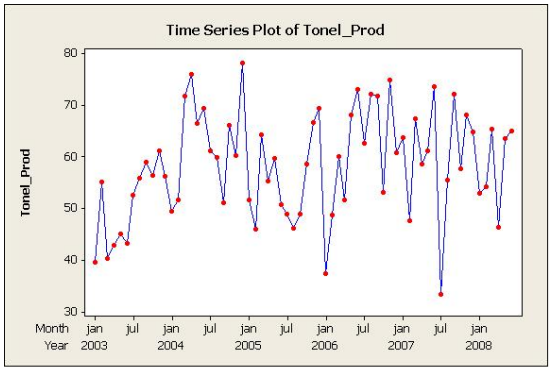


Figura Apl 8-B

A Aplicação 8 mostra a função de regressão estimada pelo método do núcleo a partir de dados reais de produção em toneladas de um produto siderúrgico durante o período de janeiro/2003 a junho/2008 [1]. Os dados reais foram divididos por 1.000 para que as rotinas suportassem as transformações matriciais efetuadas sobre os dados.

*Figura Apl 8-A :*

Curva Verde => Estimação da função estágio 1.

Curva Azul => Estimação da função estágio 2.

Podemos ver que a estimação pelo método do núcleo consegue captar a evolução dos dados no tempo.

[1] Por questões sigilosas a fonte dos dados não pode ser revelada.

## 4.4 Aplicação 9

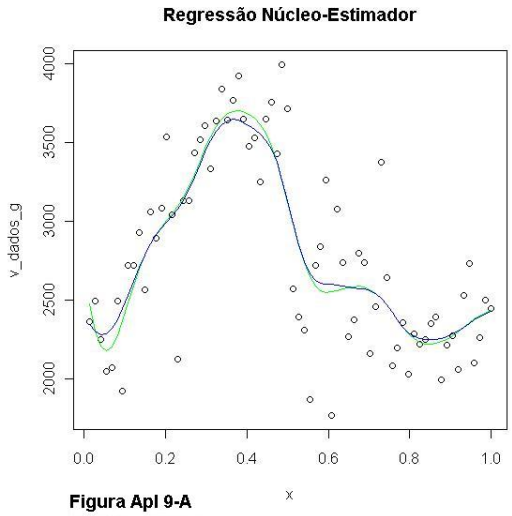


Figura Apl 9-A

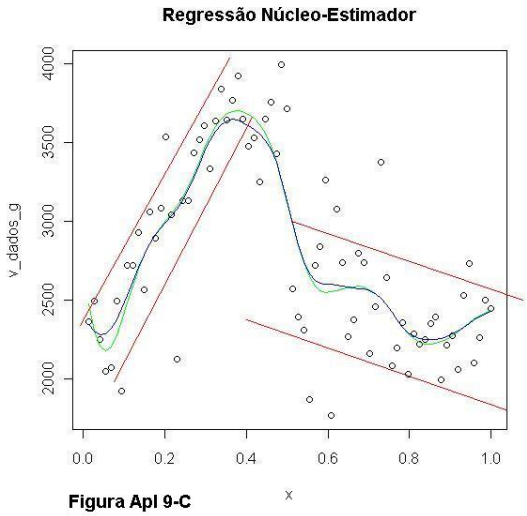


Figura Apl 9-C

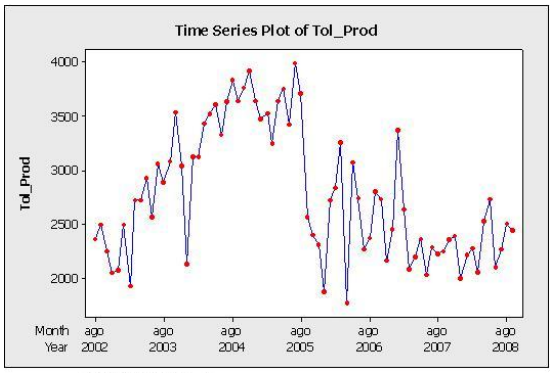


Figura Apl 9-B

A Aplicação 9 mostra a função de regressão estimada pelo método do núcleo a partir de dados da aplicação 5, onde é apresentado a produção, em toneladas, de um determinado produto siderúrgico por uma determinada máquina durante o período de agosto/2002 a setembro/2008 que se mostrou não ser normal com p-value igual a 0.00121.[2] As ordenadas $x$ foram compostas por $i/n$, onde $i$ é o mês representado por número inteiro de 1 a 74, e $n$ = 74 o total de meses.

Vemos, mais uma vez, que a função de regressão estimada ajusta bem ao dados.

Direcionando nosso foco de análise para o comportamento dos dados, podemos ver o seguinte (Figura Exp 9-C):

- Existem 2 períodos distintos : o primeiro período possui tendência não linear ascendente e o segundo descendente .
- As variâncias dos 2 grupos parecem ser diferentes.

[2] Por questões sigilosas a fonte dos dados não pode ser revelada.

- Há uma queda acentuada entre o final do primeiro período e início do segundo.
- As funções de regressão de cada grupo serão iguais?

Considerando que os dados podem ser descritos como uma série temporal não estacionária, pois ocorrem dados discrepantes, mudanças na média durante o período e a variabilidade se mostra diferente ao longo do tempo, o método do núcleo pode ser uma alternativa para análise de séries temporais dessa natureza.

Finalmente, a análise inicial nos indica que algum evento ou eventos fizeram com que houvesse uma alteração no comportamento dos dados, ou seja, da produção do produto siderúrgico. Essa análise mostra a necessidade de um estudo mais aprofundado para identificar as causas dessa mudança de forma a poder preparar um plano de ação a ser aplicado no processo produtivo.

# 5 CONCLUSÕES E TRABALHOS FUTUROS

A partir dos experimentos desenvolvidos tanto com dados simulados como com dados reais, e pela literatura existente sobre o assunto foi possível perceber que o método de estimação pelo núcleo é um método não paramétrico robusto e maduro, tanto para a estimação da função densidade quanto para estimação da função de regressão.

Essa constatação desperta, ainda mais, nossa curiosidade no conhecimento das bases matemáticas que tornam possível tal estimação, bem como a expansão dessa análise para o contexto multivariado.

O aprofundamento teórico é fundamental para consolidar o conhecimento sobre o assunto e um facilitador para expandir sua aplicabilidade. Dentre as possíveis aplicações, visualizamos sua utilização na análise de 2 amostras aleatórias de um processo produtivo com o objetivo de verificar a existência ou não de desvios / anomalias (exemplo : Aplicação 9). Com isso, pretendemos propor para a empresa um plano de ação para acompanhamento da produção da máquina cujos dados de produção foram utilizados como base para análise da aplicação 9.

Diferentemente dos métodos utilizados na análise de séries temporais que consideram apenas a variável resposta como parâmetro, o método do núcleo estimador também considera o tempo como parâmetro (veja (5) e (6)). Ou seja, outra linha de estudo que vislumbramos é a comparação dos métodos de previsão baseado nas técnicas de séries temporais com o método do núcleo.

Finalmente, podemos resumir os trabalhos futuros em :

- Conhecimento das bases matemáticas do método do núcleo-estimador;
- Aprofundar o estudo da aplicação 9 a ponto de propor um plano de ação para a empresa, se for o caso;
- Expansão do estudo para o contexto multivariado;
- Comparação das estimativas geradas pelo método no núcleo para 2 amostras aleatórias de

um processo produtivo;

- Comparação dos métodos de previsão baseado nas técnicas de séries temporais com o método do núcleo.

Sendo assim, esperamos aprofundar, em futuro próximo, o conhecimento sobre o assunto a ponto de apresentar resultados para alguns dos itens acima.

# REFERÊNCIAS

ATUNCAR, G.S., DAMASCENO, E.C., AND MENDONÇA, P.P. (1998) *Choosing the Bandwidth in Nonparametric Functional Estimation.* Relatório Técnico. Departamento de Estatística da UFMG.

BESSEGATO, L.F., ATUNCAR, G.S., AND DUCZMAN, L.H. (2006) *Rotinas em R para Técnicas de Suavização por Núcleos Estimadores.* RTP-01/2006, Departamento de Estatística da UFMG, (http://www.est.ufmg.br/portal/arquivos/rts/rtp0601.pdf)

BOWMAN, A.W.; AZZALINI, A.(1997) *Applied Smoothing Techniques for Data Analysis : The Kernel Approach with S-Plus Illustrations.* Oxford Science Publications.

FAN, J., GIJBELS, I. (1995) *Data-driven Bandwidth Selection in Local Polynomial Fitting: Variable Bandwidth and Spatial Adaptation.* Journal of the Royal Statistical Society. Series B (Methodological).

MIRANDA, M.F. (2007) *Estimação dos coeficientes de um processo de difusão.* Dissertação de Metrado. Departamento de Estatística da UFMG.

SHEATHER, S.J., JONES, M.C. (1995) *A Reliable Data-based Bandwidth Selection Method for Kernel Density Estimation.* Journal of the Royal Statistical Society.

SILVA, F. R. da (2008). Comparação de Funções de Regressão com Abordagem Não Paramétrica : Uma Aplicação a Problemas Reais. *Laboratório II*, Departamento de Estatística da UFMG.

WAND, M.P.; JONES, M.C.(1995) *Kernel Smoothing.* CHAPMAN HALL/CRC, 1995. ISBN 13: 9780412552700

WATSON, G.S. (1964) *Smooth regression analysis.* Sankhya Ser. A 26, 101-16.

# APÊNDICE A – ROTINAS R

## A.1 Código Experimento 1

```
#####################################################################
# Experimento 1
#####################################################################
### Gera Base de Dados Aleatória
#####################################################################
### Normal
#####################################################################
rm(list=ls(all=TRUE))
n <- 74
v_dados <-rnorm(n) # Gera amostra aleatória normal
hist(v_dados,breaks=10) # Desenha o histograma
shapiro.test(v_dados) # Efetua teste de normalidade

#####################################################################
### Comando  Argumentos  Descrição
#####################################################################
### func.phi2  (vetor.dados,x) : Calcula Módulo de phi ao quadrado
### alg.lambda  (vetor.dados, cota.sup=10,cota.inf=0) : Calcula Lambda
### alg.G  (vetor.dados, lambda)  : Calcula G
### alg.H  (vetor.dados, lambda)  : Calcula H
### alg.hop.dens  (vetor.dados,G ,flag)  :  Calcula hop densidade
### alg.hop.F  (vetor.dados, H ,flag)  : Calcula hop distribuição
### func.F.norm  (x,hop,vetor.dados)  : Calcula Fn(x) (núcleo gaussiano)
#####################################################################
### Carrega as rotinas relacionadas ao método do núcleo-estimador
#####################################################################
```

```
source("C:\\Nucleo Estimadores - RotinasR\\func.phi2.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.lambda.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.G.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.H.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.hop.dens.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.hop.F.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.f.dens.norm.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.F.norm.R")


#####################################################################
# Pesquisa de Lambda (Limite superior de integracao)
#####################################################################
lambda <- alg.lambda(v_dados)

#####################################################################
# Estimativa de G
#####################################################################
G_hat <- alg.G(v_dados,lambda)

#####################################################################
# Estimativa da Janela - Núcleo Gaussiano
#####################################################################
flag=0
janela=alg.hop.dens(v_dados,G_hat,flag)
janela

#####################################################################
# Estimativas Efetuadas
#####################################################################
linhas<-c("Lambda","G","Janela")
valores<-c(lambda, G_hat, janela)
tabela<-matrix(valores,ncol=1,byrow=T)
dimnames(tabela)<-list(linhas,"")
cat("\n ESTIMATIVAS \n")
```

```
print(tabela,width=10)

pontos<-74
maximo<-max(v_dados)
minimo<-min(v_dados)
passo<-(maximo-minimo)/pontos
x.malha<-seq(minimo,maximo,passo) # Gera a malha (eixo do x )

rotulo=''
eixo.x<-paste("",rotulo)
windows()
######################
# Desenha a normal teórica
######################
plot(function(x) dnorm(x, log=FALSE), -3, 3,
     main = "Normal Teórica \n e \n Função Densidade Estimada : p-value = 0.7483")
################################################
# Desenha a função estimada pelo método núcleo-estimador
################################################
lines(x.malha,func.f.dens.norm(x.malha,janela,v_dados),
      type="l",ylab="densidade",xlab=eixo.x,col='red')

#####################################################################
# Desenha a função estimada pelo método núcleo-estimador :
# Rotinas do pacote R
#####################################################################
lines(density(v_dados,bw="SJ"),col='green')
```

## A.2 Código Experimento 2

```
#####################################################################
# Experimento 2
#####################################################################
rm(list=ls(all=TRUE))
```

```
w<-seq(7,13,0.05)
length(w)
f<-(1/sqrt(2*pi))*exp(-1/2*(w-10)^2) # Calcula os dados pela normal teórica

######################################################################
# Desenha normal teórica com média 10 e desvio padrão 1
######################################################################
plot(w,f,type="l",ylab="Normal Teórica (10,1)",xlab="w",
     main="N(10,1) Teórica X Paramétrico X Núcleo-Estimador \n (p-value=0.8977)")

######################################################################
## Normal Gerada p-value : 0.8977
######################################################################
# Foi gerada uma amostra aleatória normal com p-value = 0.8977.
# O resultado foi armazenado em arquivo para reprodução posterior.
# Segue abaixo o código utilizado para geração da amostra e
# gravação em arquivo :
#    n <- length(w)   # veja a criação de "w" no código acima
## gera uma amostra aleatória normal
#    v_dados_g <-rnorm(n,10,1)
## abre uma conexão de arquivo para gravação
#    arq_v_dados_g <- file("dados_normal_8977.data", "w")
## grava os dados no arquivo
#    cat(v_dados_g,file = arq_v_dados_g, sep = "\n")
## fecha arquivo
#    close(arq_v_dados_g)
######################################################################
# Lê arquivo com os dados normais gerados
dt <- read.table("C:\\Nucleo Estimadores - RotinasR\\dados_normal_8977.data")

v_dados_g <-dt.V1
# Efetua teste de normalidade
  shapiro.test(v_dados_g)

# Calcula a média
```

```
  media<-mean(v_dados_g)
# Calculo o desvio padrão
  s <-sd(v_dados_g)
# Calcula a função pelo modelo paramétrico
  fx<-(1/(sqrt(2*pi)*s))*exp((-1/(2*s^2))*(w-media)^2)
# Desenha a curva da normal paramétrica estimada
  lines(w,fx,type="l",ylab="Densidade",xlab="w",col='red')

#####################################################################
### Carrega as rotinas relacionadas ao método do
### núcleo-estimador da densidade
#####################################################################
source("C:\\Nucleo Estimadores - RotinasR\\func.phi2.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.lambda.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.G.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.H.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.hop.dens.R")
source("C:\\Nucleo Estimadores - RotinasR\\alg.hop.F.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.f.dens.norm.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.F.norm.R")

#####################################################################
## Normal Núcleo-Estimador - p-value : 0.8978
#####################################################################

lambda <- alg.lambda(v_dados_g)
G_hat <- alg.G(v_dados_g,lambda)

flag=0
janela=alg.hop.dens(v_dados_g,G_hat,flag)

linhas<-c("Lambda","G","Janela")
valores<-c(lambda, G_hat, janela)
tabela<-matrix(valores,ncol=1,byrow=T)
dimnames(tabela)<-list(linhas,"")
```

```
cat("\n ESTIMATIVAS \n")
print(tabela,width=10)

#  Desenha a curva da função estimada pelo núcleo-estimador
   lines(w,func.f.dens.norm(w,janela,v_dados_g),type="l",col='green')
#  Desenha o histograma em outra janela R
   windows()
   hist(v_dados_g,breaks=10)
```

## A.3 Código Experimento 3

```
#####################################################################
# Experimento 3
#####################################################################
# Normal Teórica(10,1) X Normal Paramétrica Estimada X núcleos estimadores
#####################################################################
rm(list=ls(all=TRUE))
w<-seq(7,13,0.05)
length(w)
# Calcula os dados pela normal teórica
  f<-(1/sqrt(2*pi))*exp(-1/2*(w-10)^2)
#####################################################################
# Desenha normal teórica com média 10 e desvio padrão 1
#####################################################################
plot(w,f,type="l",ylab="Normal Teórica (10,1)",xlab="w",
     main="N(10,1) Teórica X Paramétrico X Núcleo-Estimador \n (p-value=0.0728)")

#####################################################################
## Normal Gerada p-value : 0.0728
#####################################################################
# Foi gerada uma amostra aleatória normal com p-value = 0.0728.
# O resultado foi armazenado em arquivo para reprodução posterior.
# Segue abaixo o código utilizado para geração da amostra e gravação em arquivo :
```

```
#   n <- length(w)  # veja a criação de "w" no código acima
# gera uma amostra aleatória normal
#   v_dados_bp <-rnorm(n,10,1)
# abre uma conexão de arquivo para gravação
#   arq_v_dados_bp <- file("dados_normal_0728.data", "w")
# grava os dados no arquivo
#   cat(v_dados_bp,file = arq_v_dados_bp, sep = "\n")
#   close(arq_v_dados_bp)
#####################################################################

dt <- read.table("C:\\Nucleo Estimadores - RotinasR\\dados_normal_0728.data")
v_dados_bp <-dt.V1       #observação : substitir o ponto de dt.V1 por cifrão
shapiro.test(v_dados_bp)
tj<-10
hist(v_dados_bp,tj,
     main = paste("Histogram da Normal (p-value=0.0728), Classes=",tj))
tj<-20
hist(v_dados_bp,tj,
     main = paste("Histogram da Normal (p-value=0.0728), Classes=",tj))
tj<-35
hist(v_dados_bp,tj,
     main = paste("Histogram da Normal (p-value=0.0728), Classes=",tj))

# Calcula a média
  media<-mean(v_dados_bp)
# Calculo o desvio padrão
  s <-sd(v_dados_bp)
# Calcula a função pelo modelo paramétrico
  fx<-(1/(sqrt(2*pi)*s))*exp((-1/(2*s^2))*(w-media)^2)
# Desenha a curva da normal paramétrica estimada
  lines(w,fx,type="l",ylab="Densidade",xlab="w",col='red')

#################################################
## Normal Núcleo Estimadores p-value : 0.0728
#################################################
```

```
lambda <- alg.lambda(v_dados_bp)
G_hat <- alg.G(v_dados_bp,lambda)

flag=0
janela=alg.hop.dens(v_dados_bp,G_hat,flag)

linhas<-c("Lambda","G","Janela")
valores<-c(lambda, G_hat, janela)
tabela<-matrix(valores,ncol=1,byrow=T)
dimnames(tabela)<-list(linhas,"")
cat("\n ESTIMATIVAS \n")
print(tabela,width=10)
lines(w,func.f.dens.norm(w,janela,v_dados_bp),type="l",col='green')
lines(density(v_dados_bp,bw="SJ"),col='blue')
```

## A.4 Código Experimento 4

```
#####################################################################
# Experimento 4
#####################################################################
### Análise Núcleo-estimador para Distribuição Weibull
#####################################################################

n <-80
v_dados_w <-rweibull(n, 2, 5)
tj<-10
hist(v_dados_w,tj,
     main = paste("Histogram da Weibull(80,shape=2,scale=5) \n Classes=",tj))
tj<-20
hist(v_dados_w,tj,
     main = paste("Histogram da Weibull(80,shape=2,scale=5) \n Classes=",tj))

maximo <- max(v_dados_w)
```

```
minimo <- min(v_dados_w)
passo <- (maximo-minimo)/n
x <-seq(minimo,maximo,by=passo)
y <-dweibull(x,2,5)
ks.test(v_dados_w,"pweibull", shape=2,scale=5)

#########################################################################
# Desenha Weibull Teórica
#########################################################################
plot(x,y,type="l",main="Weibull Teórica x Núcleo-Estimador ")

# Verificar se tem como testar weibull sem shape e scale
#########################################################################
# Pesquisa de Lambda (Limite superior de integracao)
#########################################################################
lambda <- alg.lambda(v_dados_w)
lambda

#########################################################################
# Estimativa de G
#########################################################################
G_hat <- alg.G(v_dados_w,lambda)
G_hat
#########################################################################
# Estimativa da Janela - Nucleo Gaussiano
#########################################################################
flag=0
janela=alg.hop.dens(v_dados_w,G_hat,flag)
janela

#########################################################################
# Estimativas Efetuadas
#########################################################################
linhas<-c("Lambda","G","Janela")
valores<-c(lambda, G_hat, janela)
```

```
tabela<-matrix(valores,ncol=1,byrow=T)
dimnames(tabela)<-list(linhas,"")
cat("\n ESTIMATIVAS \n")
print(tabela,width=10)

x.malha<-seq(minimo,maximo,passo)

rotulo=''
eixo.x<-paste("",rotulo)
#####################################################################
# Desenha Estimativa pelo Núcleo : Rotinas Desenvolvidas
#####################################################################
lines(x.malha,func.f.dens.norm(x.malha,janela,v_dados_w),type="l",col='red')

#####################################################################
# Desenha Estimativa pelo Núcleo : Sheather e Jones
#####################################################################
lines(density(v_dados_w,bw="SJ"),col='green')
```

## A.5 Código Aplicação 5

```
#####################################################################
# Aplicação 5
#####################################################################
# Dados Reais NÃO NORMAIS
#####################################################################
# Normal Teórica x Dados Reais Paramétrica Estimada x Núcleos Estimadores
#####################################################################
rm(list=ls(all=TRUE))

dt<-read.table("C:\\Dados Reais Nao Normais-Toneladas Produzidas Maquina 5000.csv")
v_dados_g <-dt.V1          # substituir . por cifrão
shapiro.test(v_dados_g)
```

```
media<-mean(v_dados_g)
s <-sd(v_dados_g)
v_dados_z<-(v_dados_g-media)/s
#####################################################################
## Desenha Histograma dos dados Padronizados
#####################################################################
tj<-10
hist(v_dados_z,tj,
     main = paste("Histogram Tonelada Produzida (p-value=0.00121) \n Classes=",tj))
tj<-20
hist(v_dados_z,tj,
     main = paste("Histogram Tonelada Produzida (p-value=0.00121) \n Classes=",tj))

mediaz<-mean(v_dados_z)
sz <-sd(v_dados_z)
# Calcula a função pelo modelo paramétrico
  fx<-(1/(sqrt(2*pi)*sz))*exp((-1/(2*sz^2))*(v_dados_z-mediaz)^2)
#####################################################################
## Desenha a Normal Teórica N(0,1)
#####################################################################
plot(function(x) dnorm(x, log=FALSE), -3, 3,
main = "N(0,1) Teórica x Dados Reais Paramétrico (p-value=0,001210) \n x  Núcleo")

#####################################################################
## Desenha a Normal Paramétrica Estimada
#####################################################################
lines(v_dados_z,fx,type="l",ylab="Densidade",xlab="w",col='red')

#####################################################################
## Normal Núcleo Estimadores p-value : 0,00121
#####################################################################

lambda <- alg.lambda(v_dados_z)
G_hat <- alg.G(v_dados_z,lambda)
```

```
flag=0
janela=alg.hop.dens(v_dados_z,G_hat,flag)

linhas<-c("Lambda","G","Janela")
valores<-c(lambda, G_hat, janela)
tabela<-matrix(valores,ncol=1,byrow=T)
dimnames(tabela)<-list(linhas,"")
cat("\n ESTIMATIVAS \n")
print(tabela,width=10)

pontos<-length(v_dados_z)
maximo<-max(v_dados_z)
minimo<-min(v_dados_z)
passo<-(maximo-minimo)/pontos
x.malha<-seq(minimo,maximo,passo)

#####################################################################
## Desenha Função Densidade Estimada pelo Núcleo - Rotinas desenvolvidas
#####################################################################
lines(x.malha,func.f.dens.norm(x.malha,janela,v_dados_z),type="l",col='green')

#####################################################################
## Desenha Função Densidade Estimada pelo Núcleo - Rotinas R (Sheader e Jones)
#####################################################################
lines(density(v_dados_z,bw="SJ"),col='blue')
```

## A.6 Código Aplicação 6

```
#####################################################################
# Aplicação 6
#####################################################################
# Dados Reais NORMAIS
#####################################################################
rm(list=ls(all=TRUE))
```

```
#####################################################################
## Normal Dados Reais p-value : 0.3679
#####################################################################
dt <- read.table("C:\\Dados Reais Normais - Toneladas Produzidas Operacao 11.csv")
v_dados_r <-dt.V1
shapiro.test(v_dados_r)
media<-mean(v_dados_r)
s <-sd(v_dados_r)
v_dados_z<-(v_dados_r-media)/s
#####################################################################
## Desenha Histograma dos dados Padronizados
#####################################################################
tj<-10
hist(v_dados_z,tj,
     main = paste("Histogram Tonelada Produzida (p-value=0.3679) \n Classes=",tj))
tj<-20
hist(v_dados_z,tj,
     main = paste("Histogram Tonelada Produzida (p-value=0.3679) \n Classes=",tj))

mediaz<-mean(v_dados_z)
sz <-sd(v_dados_z)
# Calcula a função pelo modelo paramétrico
  fx<-(1/(sqrt(2*pi)*sz))*exp((-1/(2*sz^2))*(v_dados_z-mediaz)^2)

#####################################################################
## Desenha a Normal Teórica N(0,1)
#####################################################################
plot(function(x) dnorm(x, log=FALSE), -3, 3,
main = "N(0,1) Teórica x Dados Reais Paramétrico (p-value=0.3679) \n x  Núcleo")

#####################################################################
## Desenha a Normal Paramétrica Estimada
#####################################################################
lines(v_dados_z,fx,type="l",ylab="Densidade",xlab="w",col='red')
```

```
#####################################################################
## Normal Núcleo Estimadores p-value : 0.3679
#####################################################################

lambda <- alg.lambda(v_dados_z)
G_hat <- alg.G(v_dados_z,lambda)

flag=0
janela=alg.hop.dens(v_dados_z,G_hat,flag)

linhas<-c("Lambda","G","Janela")
valores<-c(lambda, G_hat, janela)
tabela<-matrix(valores,ncol=1,byrow=T)
dimnames(tabela)<-list(linhas,"")
cat("\n ESTIMATIVAS \n")
print(tabela,width=10)

pontos<-length(v_dados_z)
maximo<-max(v_dados_z)
minimo<-min(v_dados_z)
passo<-(maximo-minimo)/pontos
x.malha<-seq(minimo,maximo,passo)

#####################################################################
## Desenha Função Densidade Estimada pelo Núcleo - Rotinas desenvolvidas
#####################################################################
lines(x.malha,func.f.dens.norm(x.malha,janela,v_dados_z),type="l",col='green')

#####################################################################
## Desenha Função Densidade Estimada pelo Núcleo - Rotinas R (Sheader e Jones)
#####################################################################
lines(density(v_dados_z,bw="SJ"),col='blue')
```

## A.7 Código Experimento 7

```
#####################################################################
## KERNEL REGRESSION
#####################################################################
# Experimento 7
#####################################################################
rm(list=ls(all=TRUE))
# Gera malha -pi a pi
  pp<-seq(-pi,pi,len=90)
# Gera dados da função 3*sen(x)
  y<-3*sin(pp)

dt <- read.table("C:\\Nucleo Estimadores - RotinasR\\dados_normal_8494.data")
v_dados_r <-dt.V1
shapiro.test(v_dados_r)
# Soma aos dados da função 3*sen(x) erros aleatórios normais
  yz<- y + v_dados_r

#####################################################################
# Desenha gráfico de dispersão entre x e y
#####################################################################
plot(pp,yz,type='p',col='black',main="Regressão Núcleo-Estimador")

#####################################################################
# Desenha Função 3*sen(x)
#####################################################################
lines(pp,y,type='l',col='red')

#####################################################################
# Carrega Rotinas para Estimação Não Paramétrica da Regressão
#####################################################################

source("C:\\Nucleo Estimadores - RotinasR\\func.RSC.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.IRSC.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.MSE.R")
```

```
source("C:\\Nucleo Estimadores - RotinasR\\func.IMSE.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.MH.R")

p<-1
a<-2

#OBS: xestrela é o vetor de dados

xestrela <-pp
yestrela <-yz

amplitude<-max(xestrela)-min(xestrela)
malhax0<-seq(min(xestrela),max(xestrela),length=101)
tmalha<-length(malhax0)
txestrela<-length(xestrela)

H1<-seq(amplitude/txestrela+0.05,amplitude,0.2)
H2<-seq(amplitude/txestrela+0.03,amplitude,0.2)

#OBS: O acréscimo de 0.05 e 0.03 deve-se ao fato de quando o valor de H
#    é muito pequeno as matrizes SN e SN* podem não ser invertíveis.

system.time(
estagio1<-sapply(H1,function(h) func.IRSC(malhax0,tmalha,h,xestrela,yestrela,p+a))
)
auxiliarh<-(estagio1<=min(estagio1))
for (i in 1:length(H1)){
    if (auxiliarh[i]==TRUE){
    h.hat<-H1[i]}
}

system.time(
estagio2<-sapply(H2,function(h) func.IMSE(malhax0,h,xestrela,h.hat,yestrela,p+a,p))
)
auxiliarh2<-(estagio2<=min(estagio2))
```

```
for (i in 1:length(H2)){
    if (auxiliarh2[i]==TRUE){
    h.hat2<-H2[i]}
}

#####################################################################
# Estimação da Função
#####################################################################

Mp1<-sapply(xestrela, function(x0) func.MH(h.hat,xestrela,yestrela,x0,p+a))
Mp2<-sapply(xestrela, function(x0) func.MH(h.hat2,xestrela,yestrela,x0,p))
#####################################################################
# Desenha estimação : Passo 1
#####################################################################
lines(pp,Mp1,type='l',col='yellow')
#####################################################################
# Desenha estimação : Passo 2
#####################################################################
lines(pp,Mp2,type='l',col='blue')

#####################################################################
# Desenha estimação : ksmooth
#####################################################################
lines(ksmooth(pp,yz,kernel="normal"),col="green")
```

## A.8 Código Aplicação 8

```
#####################################################################
# Aplicação 8
#####################################################################
rm(list=ls(all=TRUE))

dt <- read.table("C:\\Regressao_Dados_Reais_Toneladas_Produzidas_Div_1000.csv")
v_dados_r <-dt.V1
```

```
v_dados_t <-seq(1,66,1)
shapiro.test(v_dados_r)

plot(v_dados_t,v_dados_r,col='black',main="Tonelada Produzida 2002-2008")

#####################################################################
# FUNÇÕES PARA ESTIMAÇÃO NÃO PARAMÉTRICA
#####################################################################

source("C:\\Nucleo Estimadores - RotinasR\\func.RSC.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.IRSC.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.MSE.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.IMSE.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.MH.R")

p<-1
a<-2

#OBS: xestrela é o vetor de dados

xestrela <-v_dados_t
yestrela <-v_dados_r

amplitude<-max(xestrela)-min(xestrela)
malhax0<-seq(min(xestrela),max(xestrela),length=length(v_dados_r))
tmalha<-length(malhax0)
txestrela<-length(xestrela)

H1<-seq(amplitude/txestrela+0.05,amplitude,0.2)
H2<-seq(amplitude/txestrela+0.03,amplitude,0.2)

#OBS: O acréscimo de 0.05 e 0.03 deve-se ao fato de quando o valor de H
#    é muito pequeno as matrizes SN e SN* podem não ser invertíveis.

system.time(
```

```
estagio1<-sapply(H1,function(h) func.IRSC(malhax0,tmalha,h,xestrela,yestrela,p+a))
)
auxiliarh<-(estagio1<=min(estagio1))
for (i in 1:length(H1)){
    if (auxiliarh[i]==TRUE){
    h.hat<-H1[i]}
}

system.time(
estagio2<-sapply(H2,function(h) func.IMSE(malhax0,h,xestrela,h.hat,yestrela,p+a,p))
)
auxiliarh2<-(estagio2<=min(estagio2))
for (i in 1:length(H2)){
    if (auxiliarh2[i]==TRUE){
    h.hat2<-H2[i]}
}

#####################################################################
# ESTIMAÇÃO DA FUNÇÃO
#####################################################################

Mp1<-sapply(xestrela, function(x0) func.MH(h.hat,xestrela,yestrela,x0,p+a))
Mp2<-sapply(xestrela, function(x0) func.MH(h.hat2,xestrela,yestrela,x0,p))
#####################################################################
# Desenha estimação : Passo 1
#####################################################################
lines(v_dados_t,Mp1,type='l',col='green')
#####################################################################
# Desenha estimação : Passo 2
#####################################################################
lines(v_dados_t,Mp2,type='l',col='blue')
```

## A.9 Código Aplicação 9

```
#####################################################################
# Aplicação 9 - Ordenada x = i/n , onde 1...74, n=74 => número de meses
#####################################################################
rm(list=ls(all=TRUE))

n=74
x<-c(1:n)
for (i in 1:n) {
  x[i]<-i/n
}

dt<-read.table("C:\\Dados Reais Nao Normais-Toneladas Produzidas Maquina 5000.csv")
v_dados_g <-dt.V1
shapiro.test(v_dados_g)

#####################################################################
## Desenha Histograma dos dados Padronizados
#####################################################################
tj<-10
hist(v_dados_g,tj,
     main = paste("Histogram Tonelada Produzida (p-value=0.00121) \n Classes=",tj))
tj<-20
hist(v_dados_g,tj,
     main = paste("Histogram Tonelada Produzida (p-value=0.00121) \n Classes=",tj))

#####################################################################
# Desenha gráfico de dispersão entre x e y
#####################################################################
plot(x,v_dados_g,type='p',col='black',main="Regressão Núcleo-Estimador")

#####################################################################
# Carrega Rotinas para Estimação Não Paramétrica da Regressão
#####################################################################
```

```
source("C:\\Nucleo Estimadores - RotinasR\\func.RSC.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.IRSC.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.MSE.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.IMSE.R")
source("C:\\Nucleo Estimadores - RotinasR\\func.MH.R")

p<-1
a<-2

#OBS: xestrela é o vetor de dados

xestrela <-x
yestrela <-v_dados_g

amplitude<-max(xestrela)-min(xestrela)
malhax0<-seq(min(xestrela),max(xestrela),length=101)
tmalha<-length(malhax0)
txestrela<-length(xestrela)

H1<-seq(amplitude/txestrela+0.05,amplitude,0.2)
H2<-seq(amplitude/txestrela+0.03,amplitude,0.2)

#OBS: O acréscimo de 0.05 e 0.03 deve-se ao fato de quando o valor de H
#    é muito pequeno as matrizes SN e SN* podem não ser invertíveis.

system.time(
estagio1<-sapply(H1,function(h) func.IRSC(malhax0,tmalha,h,xestrela,yestrela,p+a))
)
auxiliarh<-(estagio1<=min(estagio1))
for (i in 1:length(H1)){
    if (auxiliarh[i]==TRUE){
    h.hat<-H1[i]}
}

system.time(
```

```
estagio2<-sapply(H2,function(h) func.IMSE(malhax0,h,xestrela,h.hat,yestrela,p+a,p))
)
auxiliarh2<-(estagio2<=min(estagio2))
for (i in 1:length(H2)){
    if (auxiliarh2[i]==TRUE){
    h.hat2<-H2[i]}
}

#####################################################################
# Estimação da Função
#####################################################################

Mp1<-sapply(xestrela, function(x0) func.MH(h.hat,xestrela,yestrela,x0,p+a))
Mp2<-sapply(xestrela, function(x0) func.MH(h.hat2,xestrela,yestrela,x0,p))
#####################################################################
# Desenha estimação : Passo 1
#####################################################################
lines(x,Mp1,type='l',col='green')
#####################################################################
# Desenha estimação : Passo 2
#####################################################################
lines(x,Mp2,type='l',col='blue')
```