

THIAGO DA COSTA GONÇALVES

**METODOLOGIAS ALTERNATIVAS AOS GRÁFICOS DE CONTROLE NA
CARACTERIZAÇÃO DE PROCESSOS UNIVARIADOS**

Dissertação apresentada à
Universidade Federal de Viçosa,
como parte das exigências do
Programa de Pós-Graduação em
Estatística Aplicada e Biometria,
para obtenção do título de *Magister
Scientiae*.

**VIÇOSA
MINAS GERAIS – BRASIL
2008**

THIAGO DA COSTA GONÇALVES

**METODOLOGIAS ALTERNATIVAS AOS GRÁFICOS DE CONTROLE NA
CARACTERIZAÇÃO DE PROCESSOS UNIVARIADOS**

Dissertação apresentada à
Universidade Federal de Viçosa,
como parte das exigências do
Programa de Pós-Graduação em
Estatística Aplicada e Biometria,
para obtenção do título de *Magister
Scientiae*.

APROVADA: 07 de Novembro de 2008.

Carlos Henrique Osório Silva
(Co-Orientador)

Nerilson Terra Santos

Antônio Policarpo Souza Carneiro

Paulo Roberto Cecon

José Ivo Ribeiro Júnior
(Orientador)

Dedico a minha mãe Denize, meu pai Vicente,
meus irmãos Júnior, Eder e Letícia e a
minha namorada Caroline.

AGRADECIMENTOS

A Coordenação de Aperfeiçoamento de Pessoas de Nível Superior – CAPES – pela bolsa de estudo concedida que foi de fundamental importância na realização deste trabalho.

A Universidade Federal de Viçosa pela estrutura e oportunidade.

Aos professores do PPESTBIO, Sebastião Martins Filho, José Ivo Ribeiro Júnior, Nerilson Terra Santos, Paulo Roberto Cecon, Carlos Henrique Osório Silva, Cosme Damião Cruz, Antonio Policarpo Souza Carneiro, Luiz Alexandre Peternelli, Fabyano Fonseca e Silva e Adair José Regazzi pelos ensinamentos e ajuda.

Ao Gustavo, Ângelo e Rafael pela ajuda na realização deste trabalho.

Ao Altino pela boa vontade e ajuda.

A minha família e a da minha namorada pelo apoio incondicional, em especial a meu avô Vicentinho.

Aos meus amigos, Joilson, Walber, Vinicius, Livia, Lucas, Hudson, João Paulo, Dininha, Edmilson, Tadeu, Diógenes, Ramon e André pelo incentivo.

A Viçosa que me deu grandes alegrias, e verdadeiros amigos como Dudu, Daniel, Diego, Danilo e Dimião.

Aos companheiros de república, Zé, Rafael, Dudu, Eder, Cristiano, Bernardo, Diego, Marcio, Vander, Diogo, Dimião, Fabiano, Flavio e Marcio, que foram minha família em Viçosa.

Aos companheiros de mestrado, Danilo, Alex, Nilsa, Telma, Fernando, Antonio, Oscar, Willerson, Flavia, Ana Carolina, Bráulia e Moyses pelo companheirismo.

E principalmente ao meu orientador José Ivo Ribeiro Júnior pela atenção, incentivo, orientação e conselhos. Sua ajuda foi fundamental durante todo o trabalho.

SUMÁRIO

| | |
|---|-----|
| LISTA DE FIGURAS | v |
| LISTA DE TABELA | vi |
| LISTA DE ABREVIATURAS E SIGLAS | vii |
| RESUMO | ix |
| ABSTRACT | x |
| 1 – INTRODUÇÃO..... | 1 |
| 2 – REVISÃO DE LITERATURA..... | 2 |
| 2.1 – GRÁFICO DE CONTROLE..... | 2 |
| 2.1.1 – Gráfico de Controle Xbarra de Shewhart..... | 6 |
| 2.1.2 – Gráfico de Controle EWMA | 8 |
| 2.1.3 – Gráfico de Controle de CUSUM | 10 |
| 2.2 – REDES NEURAIAS ARTIFICIAIS | 13 |
| 2.3 – REGRESSÃO LOGÍSTICA | 22 |
| 2.4 – ANÁLISE DISCRIMINANTE..... | 25 |
| 3 – MATERIAIS E MÉTODOS | 29 |
| 3.1 – SIMULAÇÃO DOS DADOS | 29 |
| 3.2 – GRÁFICOS DE CONTROLE | 32 |
| 3.3 – REDES NEURAIAS ARTIFICIAIS | 33 |
| 3.4 – REGRESSÃO LOGÍSTICA | 34 |
| 3.5 – ANÁLISE DISCRIMINANTE | 37 |
| 3.6 – MEDIDAS DE COMPARAÇÃO | 37 |
| 4 – RESULTADOS E DISCUSSÃO | 39 |
| 4.1 – GRÁFICOS DE CONTROLE | 39 |
| 4.1.1 – Xbarra de Shewhart | 39 |
| 4.1.2 – EWMA | 40 |
| 4.1.3 – CUSUM Tabular | 40 |
| 4.2 – REDE NEURAL ARTIFICIAL | 41 |
| 4.3 – ALARMES FALSO E VERDADEIRO | 42 |
| 4.3.1 – Alarme Falso | 42 |

| | |
|--|----|
| 4.3.2 – Alarme Verdadeiro | 45 |
| 4.3.3 – Alarme Falso e Alarme Verdadeiro | 51 |
| 5 – CONCLUSÃO | 53 |
| 6 – REFERÊNCIAS BIBLIOGRÁFICAS | 54 |
| ANEXO I | 59 |
| ANEXO II | 63 |
| ANEXO III | 67 |
| ANEXO IV | 68 |

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1. Gráfico de Controle do comportamento de uma característica de qualidade variando em torno da linha média (LM) e dentro dos limites inferior (LIC) e superior (LSC) de controle num processo sob controle estatístico..... | 4 |
| Figura 2. Estrutura básica de um neurônio artificial | 15 |
| Figura 3. Primeiro treinamento onde as primeiras observações foram simuladas com média μ_0 seguidas de observações com médias μ_1 e μ_0 | 30 |
| Figura 4. Segundo treinamento onde as primeiras observações foram simuladas com média μ_0 seguidas de observações com média μ_1 | 31 |
| Figura 5. Estimativa do alarme verdadeiro em função de delta sem a presença da primeira autocorrelação..... | 49 |
| Figura 6. Estimativa do alarme verdadeiro em função de delta com a presença da primeira autocorrelação ($\rho_1 = 0,3$)..... | 49 |
| Figura 7. Estimativa do alarme verdadeiro em função de delta com a presença da primeira autocorrelação ($\rho_1 = 0,6$)..... | 50 |
| Figura 8. Estimativa do alarme verdadeiro em função de delta com a presença da primeira autocorrelação ($\rho_1 = 0,9$)..... | 50 |

LISTA DE TABELA

| | |
|---|----|
| Tabela 1. Tipos de funções de ativação utilizados em RNA..... | 17 |
| Tabela 2. Funções de ligações mais comumente utilizadas..... | 24 |
| Tabela 3. Descrição e características..... | 33 |
| Tabela 4. Equações de regressão ajustadas em função de k e δ para os alarmes falso e verdadeiro e os respectivos coeficientes de determinação..... | 39 |
| Tabela 5. Equações de regressão ajustadas em função de p , δ e λ para os alarmes falso e verdadeiro e os respectivos coeficientes de determinação..... | 40 |
| Tabela 6. Equações de regressão ajustadas em função de p e δ para os alarmes falso e verdadeiro e os respectivos coeficientes de determinação..... | 41 |
| Tabela 7. Análise de Variância para alarme falso | 42 |
| Tabela 8. Médias dos alarmes falsos dos métodos para pontos fora de controle estatístico posicionados ao meio (0) e ao final (1) do conjunto de dados..... | 43 |
| Tabela 9. Médias dos alarmes falsos dos métodos para deltas iguais a 1, 2 e 3.. | 44 |
| Tabela 10. Médias dos alarmes falsos dos métodos para autocorrelações iguais a 0, 0,3, 0,6 e 0,9..... | 45 |
| Tabela 11. Análise de Variância para alarme verdadeiro..... | 46 |
| Tabela 12. Médias dos alarmes falsos dos métodos para deltas iguais a 1, 2 e 3. | 47 |
| Tabela 13. Médias dos alarmes verdadeiros dos métodos para autocorrelações iguais a 0, 0,3, 0,6 e 0,9..... | 48 |
| Tabela 14. Estimativas em percentuais dos alarmes verdadeiros..... | 48 |

LISTA DE ABREVIATURAS E SIGLAS

- AD – Análise Discriminante
- AR – Auto-Regressivo
- ARIMA – Média Móvel Auto-Regressiva Integrada
- CEQ – Controle Estatístico da Qualidade
- FV – Fonte de Variação
- GC – Gráfico de Controle
- GCs – Gráficos de Controle
- GC CUSUM – Gráfico de Controle da Soma Cumulativa
- GC CUSUM Tabular – Gráfico de Controle da Soma Cumulativa Tabular
- GC de Shewhart – Gráfico de Controle de Shewhart
- GC EWMA – Gráfico de Controle da Média Móvel Exponencialmente Ponderada
- GL – Grau de Liberdade
- H – Intervalo de decisão
- H_0 – Hipótese de nulidade
- H_1 – Hipótese alternativa
- h^* – Valor padronizado de H
- K – Valor de referência
- k^* – Valor padronizado de K
- LIC – Limite Inferior de Controle
- LM – Linha Média
- LSC – Limite Superior de Controle
- m – método
- MSE – Mean Square Error (Erro Médio Quadrático)
- NMA – Número Médio de Amostras
- p – posição
- Pd – poder
- $P\hat{d}$ – poder estimado
- QM – Quadrado Médio

R^2 – Coeficiente de determinação
RL – Regressão Logística
RNA – Rede Neural Artificial
RNAs – Redes Neurais Artificiais
 S_H – CUSUM Unilateral Superior
 S_L – CUSUM Unilateral Inferior
SQ – Soma de Quadrados
trainbr – Bayesian regularization *backpropagation*
traingda – Gradient descent with adaptive learning rate *backpropagation*
traingdx – Gradient descent with momentum and adaptive learning rate
backpropagation
trainlm – Levenberg-Marquardt *backpropagation*
UFV – Universidade Federal de Viçosa
 α – Alarme falso
 $\hat{\alpha}$ – Alarme falso estimado
 β – Falta de detecção
 δ – Delta
 λ – Peso da amostra
 μ – Média populacional
 ρ_1 – Coeficiente de primeira correlação
 σ – Desvio-padrão populacional

RESUMO

GONÇALVES, Thiago da Costa, M.Sc., Universidade Federal de Viçosa, novembro de 2008. **Metodologias alternativas aos gráficos de controle na caracterização de processos univariados.** Orientador: José Ivo Ribeiro Júnior. Co-Orientadores: Carlos Henrique Osório Silva e Fabyano Fonseca e Silva

As empresas têm investido cada vez mais na qualidade, no sentido de buscar maior sobrevivência no mercado, ganhar mais clientes satisfeitos e aumentar a produtividade e a lucratividade. Para buscá-la com mais êxito com auxílio do controle estatístico, faz-se uso dos gráficos de controle que monitoram o processo e sinalizam se há necessidade de corrigi-lo, de maneira que o produto final possa estar dentro dos padrões exigidos pelos consumidores. Na entanto, metodologias alternativas aos gráficos de controle, também podem ser utilizadas para classificar ou discriminar se o processo está ou não sob controle. Neste trabalho foram aplicadas as seguinte: análise discriminante, regressão logística e redes neurais artificiais. Para aplicar cada um dos métodos propostos utilizaram-se dados simulados, onde as medidas de comparação entre eles foram baseadas nas incidências dos alarmes falsos e verdadeiros sobre a classificação desses valores em dentro ou fora de controle estatístico. Foram simulados valores normais e independentemente distribuído sob controle estatístico. Posteriormente, foram impostas variações para que ao final do conjunto de dados saíssem de controle ou que apresentassem autocorrelações. As redes neurais artificiais e regressão logística se mostraram capazes de substituírem os melhores tipos de gráficos de controle, em sinalizarem pontos fora de controle situados ao meio ou ao final do conjunto de dados, sob diferentes distâncias da média de controle e distribuído de forma independente ou não.

ABSTRACT

GONÇALVES, Thiago da Costa, M.Sc., Universidade Federal de Viçosa, November, 2008. **Alternative approaches to the characterization of control charts for cases univariate.** Advisor: José Ivo Ribeiro Júnior. Co-Advisors: Carlos Henrique Osório Silva and Fabyano Fonseca e Silva

The companies have invested more in quality, to find more survivor in the market, to win more satisfied customers and to increase productivity and profitability. To get it with more success with the aid of statistical control, can be used control charts that monitor the process and indicate whether is need to fix it, so that the final product can be within the standards required by consumers. However, alternative methods to the control charts, also can be used to classify or discriminate if the process is on control or not. In this research, the following were applied: analysis discriminant, logistic regression and artificial neural networks. To apply each one of the proposed methods were used simulated data, where the measures of comparison between them were based on incidences of false and true alarms on the classification these values into or outside the control statistic. Were simulated normal values and independently distributed under control statistics. Subsequently, were imposed some changes to the values set in the middle or the end of data, left the control or to submit autocorrelations. The artificial neural networks and logistic regression were able to replace the best types of control charts in signaled out of control points located in the middle or the end of data set under different distances from the average of control and distributed independently or not.

1 – INTRODUÇÃO

A qualidade tornou-se um dos mais importantes fatores de decisão dos consumidores na seleção de produtos e serviços que competem entre si. O fenômeno é geral, independente do fato do consumidor ser um indivíduo, uma organização industrial, uma loja de varejo, ou um programa militar de defesa (MONTGOMERY, 2004). Com isso, aqueles que apresentarem melhor qualidade, ou seja, menor variabilidade e maior proximidade da média ao valor-alvo nos processos e produtos, provavelmente serão preferidos.

O controle estatístico da qualidade (CEQ) propicia o monitoramento do processo e a sinalização da necessidade ou não de agir sobre ele, de maneira que os resultados possam atingir os padrões pré-definidos. Os gráficos de controle (GCs) são os principais métodos do CEQ, por serem utilizados com maior frequência principalmente na indústria, dado à sua facilidade de utilização e interpretação.

Quando fontes não-usuais de variabilidade estão presentes, os dados amostrais serão plotados fora dos limites de controle. Isto é um sinal da necessidade de alguma investigação do processo e de alguma ação corretiva para a remoção dessas fontes não-usuais de variabilidade (MONTGOMERY, 2004). Portanto, os GCs possibilitam classificar o processo em duas categorias: sob ou fora de controle estatístico de qualidade.

Como alternativa a eles existem as seguintes metodologias, tais como: análise discriminante (AD), regressão logística (RL) e redes neurais artificiais (RNAs). Estes métodos podem substituir os GCs com vantagens de possibilitar a classificação do processo em fora de controle de qualidade no exato momento em que o processo sair de controle. Diferentemente dos GCs, eles permitem monitorar o processo sem que haja a necessidade de serem estabelecidos os limites de controle. No entanto, para usar tais métodos é necessário estabelecer padrões de qualidade, estipulados a partir de dados históricos que forneceram resultados satisfatórios ou a partir de dados simulados obtidos em função de uma especificação pré-definida do processo.

Com objetivo de substituir os GCs pelos métodos AD, RL e RNAs, foram testados cada um dos métodos como alternativas aos melhores GCs na capacidade de sinalizar pontos fora de controle situados ao meio ou ao final do conjunto de dados, em

diferentes distâncias da média de controle e distribuídos de formas independentes ou não. Para tais comparações foram medidas as incidências dos alarmes falsos e verdadeiros.

2 – REVISÃO DE LITERATURA

2.1 – Gráficos de Controle

Sobre qualidade, Montgomery (2004) prefere defini-la como sendo inversamente proporcional à variabilidade. Note que essa definição implica que se a variabilidade de uma característica importante de um produto ou serviço decresce, a qualidade aumenta. No entanto, outros autores já apresentaram definições diferentes.

Costa et al. (2004) fizeram uma boa revisão sobre os autores que apresentaram diferentes conceitos de qualidade. Para Juran, significa adequação ao uso. Para Deming, qualidade significa atender e, se possível, exceder as expectativas do consumidor. Para Crosby, qualidade significa atender às especificações. Para Taguchi, a produção, o uso e o descarte de um produto sempre acarretam prejuízos (“perdas”) para a sociedade; quanto menor for a perda melhor será a qualidade do produto.

Em função da presença da variabilidade, os métodos estatísticos desempenham papel importante nos esforços para a melhoria da qualidade. Em 1924, Walter A. Shewhart, do Bell Telephone Laboratories, desenvolveu o conceito estatístico do gráfico de controle, que é considerado como o começo formal do CEQ. No começo da década de 20, Harold F. Dodge e Harry G. Romig, ambos do Bell Telephone Laboratories, desenvolveram a amostragem de aceitação como uma alternativa a 100% de inspeção. No meio da década de 30, os métodos estatísticos de controle da qualidade eram largamente usados na Wester Electric, a divisão de manufatura da Bell. As décadas de 50 e 60 testemunharam a emergência da engenharia da confiabilidade, a introdução de

vários livros textos importantes em controle da qualidade e o ponto de vista que a qualidade é o caminho para gerenciar a organização (MONTGOMERY, 2004).

Os gráficos de controle são comparações do desempenho do processo com determinados limites de controle e permitem verificar se os pontos distribuem-se segundo padrões aleatórios. Caso isso aconteça, o processo sob análise estará sob controle estatístico, ou seja, somente causas aleatórias (comuns) da variação estão atuando no processo (MONTGOMERY, 2004). Quando apenas essas causas estão presentes, a média e o desvio padrão se mantêm constantes e o processo estará sob controle estatístico (estável) e com comportamento previsível por meio de probabilidades.

Se houver pontos além dos limites ou se padrões não aleatórios forem identificados, a variabilidade do processo poderá não estar se comportando mais de forma aleatória e sim devida a causas especiais, que fazem com que o processo se comporte de modo não previsível ou fora de controle de qualidade. Isso pode levar alterações significativas na média (μ) e/ou no desvio-padrão (σ). Uma investigação imediata deve ser realizada, para verificar se tal alarme deve-se realmente a causas especiais de variação. Em caso positivo, algo precisa ser feito para retornar o processo à condição desejada (JURAN, 1999).

A plotagem dos dados das características de qualidade deve ser feita de forma cronológica ou em função do número da amostra, para que seja possível identificar quais são as causas especiais de variação. O conceito de processo sob ou fora de controle estatístico é tão importante quanto o de variabilidade, e torna-se extremamente importante ressaltar que os critérios para a identificação dos problemas compreendem a existência de padrões não aleatórios, os quais incluem, entre outros, a existência de pontos fora dos limites de controle.

O gráfico de controle (Figura 1) demonstra o comportamento de uma característica de qualidade e suas flutuações em torno de um valor médio e permite distinguir entre os dois tipos de causas de variação. Basicamente, ele é uma representação de um determinado estimador, que consiste de uma linha média (LM) e dos limites inferior (LIC) e superior de controle (LSC), em função

das diversas estratificações temporais ou espaciais possíveis de serem estabelecidas (MONTGOMERY e RUNGER, 2003).

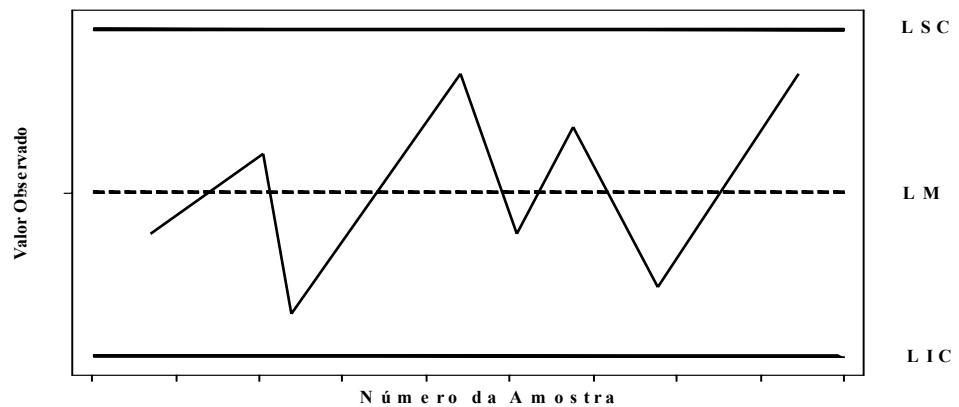


Figura 1. Gráfico de controle do comportamento de uma característica de qualidade variando em torno da linha média (LM) e dentro dos limites inferior (LIC) e superior (LSC) de controle num processo sob controle estatístico.

A presença de causas aleatórias no processo é sugerida pela ocorrência de diferenças não significativas entre a estimativa e a média de controle (LM). Supondo que os dados possuem distribuição normal de probabilidades, e se os limites de controle forem iguais a $\mu \pm 3\sigma$, a probabilidade de um ponto cair fora dos limites é de 0,0027 (0,00135 em cada lado), ou seja, espera-se que aproximadamente 27 de cada 10.000 observações (MARQUARDT, 1984), sendo μ e σ a média e o desvio-padrão populacionais.

Existem vários tipos de gráficos de controle, conforme os valores das características ou a finalidade, podendo-se dividi-los em gráficos de controle por atributos ou por variáveis.

Nos gráficos de controle por atributos existem duas grandes categorias: aqueles que classificam os itens em conformes ou não-conformes, como é o caso dos gráficos da fração defeituosa ou do número de defeituosos e aqueles que consideram o número (quantidade) de não conformidades existentes, tais como os gráficos do número de defeitos na amostra ou por unidade.

Os gráficos de controle por variáveis podem ser utilizados para características de qualidade que podem ser medidas ou que sofrem uma variação contínua, pois o controle do processo exige o monitoramento da média e da variabilidade.

Algumas das metodologias indicadas para a construção dos diferentes gráficos de controle são os gráficos de Shewhart, EWMA (médias móveis exponencialmente ponderadas) e CUSUM (somas cumulativas).

Independente do tipo de gráfico de controle, algumas pressuposições precisam ser satisfeitas pelos dados para validar a sua utilização e os seus resultados. De acordo com Alwan e Roberts (1995), as observações devem ser independentes e identicamente distribuídas e que sigam alguma distribuição de probabilidades específica, tais como a normal, binomial ou Poisson. Desse modo, as regras para identificação de padrões não aleatórios são validas quando pressuposições são satisfeitas. Caso isso não se verifique, a utilização dos gráficos de controle será no mínimo questionável.

Atienza (1997), Faltin (1997) e Montgomery (2004) discutem várias formas de lidar com a ausência de independência, ou seja, com a presença de autocorrelação nos dados, sugerindo ajustar ao processo um modelo de séries temporais ARIMA (média móvel auto-regressiva integrada) e aplicar os gráficos de controle aos resíduos obtidos.

Quando desenvolveu os gráficos de controle, W. A. Shewhart definiu que os limites de controle deveriam ficar a três desvios-padrão da LM do gráfico, o que incluiria praticamente todos os dados (WHEELER, 1995). Os que estivessem além dos limites indicariam a presença de problemas, merecendo investigação. Shewhart fixou o valor em três com base em sua experiência prática. Muitos praticantes de CEQ afirmam que somente quando for possível aproximar a distribuição dos pontos no gráfico por uma distribuição normal tal regra dos três desvios será válida. Mas, Wheeler (1995) declara que independentemente da distribuição da estatística amostral, virtualmente todos os valores irão estar dentro dos limites de três desvios-padrão, sempre que o processo estiver sob controle estatístico. Por outro lado, os dados sob distribuição normal propiciam estimativas dos limites de controle, dada às melhores propriedades do estimador do desvio-padrão.

2.1.1 – Gráfico de Controle Xbarra de Shewhart

O gráfico de controle de Xbarra serve para monitorar processos cuja característica de qualidade de interesse Y é uma grandeza mensurável: o diâmetro de um eixo, o volume de leite no saquinho, o teor de carbono em uma liga metálica, etc. O primeiro modelo formal desse gráfico de controle foi proposto por Walter A. Shewhart (1924), que atualmente leva o seu nome. Suponha que a média (μ_Y) e o desvio-padrão (σ_Y) populacional de Y sejam conhecidos, as seguintes equações são usadas para descrever os três parâmetros que caracterizam os gráficos de controle de Shewhart (MESSINA, 1987):

$$LSC = \mu_Y + k\sigma_{\bar{Y}}$$

$$LM = \mu_Y$$

$$LIC = \mu_Y - k\sigma_{\bar{Y}}$$

onde LSC é o limite superior de controle, LM é a linha média ou a média de controle, LIC é o limite inferior do processo, e k é a distância dos limites de controle até a LM, a qual é expressa como um múltiplo do desvio-padrão, sendo o valor de k mais usual igual a três.

Considere o gráfico Xbarra de Shewhart, supondo que causas aleatórias não alteram a média de Y , sendo H_0 e H_1 hipóteses mutuamente exclusivas dadas por:

$$H_0 : \mu_Y = \mu_0 \text{ (processo sob controle);}$$

$$H_1 : \mu_Y \neq \mu_0 \text{ (processo fora de controle);}$$

onde μ_0 é a média de controle.

Se a hipótese H_0 for verdadeira e rejeitada, comete-se o erro tipo I ou erro α . Por outro lado, se H_0 for falsa e aceita, comete-se erro tipo II ou β (COSTA et al., 2004). No CEQ, o alarme falso (α) representa a probabilidade de erroneamente considerar-se o processo fora de controle quando ele está sob controle (H_0 verdadeira). A consequência prática é de intervir no processo

quando ele está isento de causas especiais. Já a falta de detecção (β) representa a probabilidade de erroneamente considerar-se o processo sob controle, quando ele está fora de controle (H_1 verdadeira). A consequência prática é de não intervir no processo quando ele está sob influência de causas especiais. O alarme verdadeiro ou poder do gráfico ($Pd = 1 - \beta$) é definido como a probabilidade de detecção, ou seja, de rejeitar H_0 falsa (COSTA et al., 2004).

Estatisticamente, pode-se verificar a eficiência de um determinado gráfico de controle, com base nas estimativas dos erros devidos ao alarme falso (α) e à falta de detecção (β). Quando a hipótese H_1 é verdadeira (processo fora de controle), o ideal seria que o primeiro ponto plotado se situasse fora dos limites de controle. Contudo isso nem sempre ocorre, principalmente se o deslocamento sofrido pela média for pequeno e se deslocamento da média de controle (μ_0) que se quer detectar como variação especial em número de desvios-padrão da variável sob controle (σ_0) dos valores de Y (δ) for grande. Esse deslocamento se refere à média fora de controle da hipótese H_1 ($\mu_1 = \mu_0 + \delta\sigma_0$) que se quer determinar como significativa, sendo:

$$\delta = \frac{\mu_1 - \mu_0}{\sigma_0}$$

A medida de eficiência mais usual dos gráficos de controle é o número médio de amostras até o sinal (NMA), que segue uma distribuição geométrica de parâmetro p , independentemente de se tratar de alarme falso ou verdadeiro. Se a hipótese H_0 for verdadeira ($\delta = 0$), então $p = \alpha$. Se a hipótese H_1 for verdadeira ($\delta > 0$), então $p = Pd$.

Como a média da distribuição geométrica de parâmetro p é igual a $1/p$, então têm-se:

$$NMA_{\delta=0} = \frac{1}{\alpha}$$

$$NMA_{\delta>0} = \frac{1}{Pd}$$

Assim, um gráfico de controle bem planejado é aquele que apresenta, em termos teóricos, um baixo alarme falso ($\alpha \leq 0,10$ ou $\alpha \leq 0,05$ ou $\alpha \leq 0,01$), um alto poder ($Pd \geq 0,90$ ou $Pd \geq 0,95$ ou $Pd \geq 0,99$), um alto $NMA_{\delta=0}$ e um baixo $NMA_{\delta>0}$, em função do valor de δ pré-estabelecido. Nesse caso, recomenda-se escolher valores de $\delta \geq 3$, no início do controle da qualidade, para $2 \leq \delta < 3$ e $1 < \delta < 2$, nas fases intermediárias e, para $\delta \leq 1$, quando o controle da qualidade já se encontra numa fase mais adiantada, em termos de redução da variabilidade do processo. Assim, no início, são controladas grandes variações devidas às causas especiais e, com o passar do tempo, busca-se detectar efeitos cada vez menores dessas possíveis causas, se houver necessidade.

Portanto, a primeira escolha é de um gráfico de controle com baixa incidência de alarme falso, que trará confiabilidade no monitoramento do processo, já que os pontos fora de controle não serão, na grande maioria das vezes, devidos às causas aleatórias. Caso haja a necessidade de aumentar a qualidade do processo e, conseqüentemente, de diminuir a sua variabilidade, será importante aumentar a sensibilidade do gráfico pelo aumento do seu poder em detectar novas causas especiais que eram detectadas anteriormente como causas aleatórias. No entanto, o aumento do Pd não poderá acontecer em função do aumento do α , mas mantendo-o em níveis baixos e aceitáveis.

Usualmente os gráficos de controle de Shewhart costumam ser pouco sensíveis a pequenos desvios no processo, porque utilizam apenas a informação do último ponto plotado, ao invés de toda a seqüência de pontos.

2.1.2 – Gráfico de Controle EWMA

O gráfico de controle EWMA é uma boa alternativa ao gráfico de controle de Shewhart, quando se está interessado em detectar pequenas mudanças no processo (MONTGOMERY, 2004). Ele é baseado em observações individuais

(n=1) ou em amostras que contêm mais de uma unidade (n>1) de tamanhos constantes ou variáveis. No gráfico EWMA, a decisão sobre o processo é baseada na informação de cada amostra com um percentual de peso das anteriores.

Para a construção do gráfico de controle EWMA, devem-se escolher os valores de m (tamanho da amostra), n (número de itens por amostra), k (número de desvios-padrão em torno da LM) e λ (peso da amostra) tal que $0 < \lambda \leq 1$. Os valores da estatística W_i plotados no gráfico EWMA da medida individual são calculados pela seguinte expressão:

$$W_i = \lambda y_i + (1-\lambda)W_{i-1}, \quad \text{para } i = 1, 2, \dots, m$$

À medida que λ tende para um o peso da amostra tende a zero e esse gráfico de controle se aproxima do gráfico de Shewhart, já que:

$$\lim_{\lambda \rightarrow 1} W_i = y_i$$

Caso μ_Y seja conhecida, tem-se: $W_0 = \mu_Y$. Caso contrário, se ela for desconhecida e substituída pelo seu estimador, então tem-se: $W_0 = \bar{Y}$. Para as determinações da LM e dos limites de controle para cada amostra i, são utilizadas as expressões:

$$LSC_i = \mu_Y + k\sigma_Y \sqrt{\left(\frac{\lambda}{2-\lambda}\right) [1 - (1-\lambda)^{2i}]}$$

$$LM = \mu_Y$$

$$LIC_i = \mu_Y - k\sigma_Y \sqrt{\left(\frac{\lambda}{2-\lambda}\right) [1 - (1-\lambda)^{2i}]}$$

Se os valores da estatística W_i plotados permanecerem entre os limites inferior e superior de controle e de forma aleatória, então se tem o processo sob controle estatístico da qualidade. Caso contrário, o processo será considerado fora, apesar das decisões não estarem livres dos erros α e β (COSTA et al., 2004).

2.1.3 – Gráfico de Controle CUSUM

O gráfico de controle CUSUM utiliza informações de diversas amostras para decidir o estado do processo. Ao plotar a soma acumulada dos desvios das médias da característica de qualidade, em relação a um valor-alvo, em sucessivas amostras, as mudanças permanentes na média do processo, mesmo as menores levarão a um valor mensurável para a soma cumulativa dos desvios. Isso permite a detecção de pequenos desvios que passaram despercebidos pelo gráfico Xbarra de Shewhart (COSTA et al., 2004). Por exemplo, se devido ao desgaste das máquinas um processo sai suavemente fora de controle, produzindo resultados abaixo ou acima do valor-alvo, o gráfico CUSUM mostrará, se houver, um aumento ou decréscimo constante na soma cumulativa dos desvios da especificação.

Os gráficos de controle da soma cumulativa foram inicialmente propostos na Inglaterra por Page (1954) e foram estudados por diversos autores. Ewan (1963) fez o esboço para vários esquemas de controle gráfico e o tipo de processo para os quais os gráficos CUSUM são mais apropriados. Goel e Wu (1973) apresentaram um procedimento para o projeto econômico do gráfico CUSUM para controlar a média de um processo com as características de qualidade normalmente distribuídas. A técnica é empregada para determinar os valores ótimos do tamanho da amostra, o intervalo de amostragem e o limite de decisão. Reynolds (1975) apresentou uma aproximação do número de amostras necessário para o gráfico CUSUM sinalizar um ponto fora de controle além de comparar com um procedimento que não requer a suposição de normalidade. Hawkins (1981) apresentou uma técnica para empregar o mesmo procedimento

da CUSUM usada para a média, para controlar a variabilidade. Woodall (1985) apresentou um método para projetar gráficos de controle da qualidade, com base no desempenho estatístico a partir das especificações das regiões dentro e fora de controle.

Montgomery (2004) apresentou os gráficos de controle da soma acumulada, aplicados à média e à variabilidade do processo, e mencionou que é possível projetar procedimentos de CUSUM para outras variáveis estatísticas, tais como amplitude e desvio-padrão, além de estudar variáveis binomiais e de Poisson em gráficos por atributos.

O gráfico CUSUM incorpora diretamente toda a seqüência de informações, demarcando as somas acumuladas dos desvios dos valores da amostra em relação a um valor-alvo. Supondo que amostras de tamanho $n \geq 1$ são coletadas, \bar{Y}_{jx} , é a média da j-ésima amostra e μ_0 é o valor desejado para a média do processo, o gráfico de controle da soma acumulada é formado demarcando a quantidade resultante da fórmula (2.1) junto à amostra i:

$$C_i = \sum_{j=1}^i (\bar{Y}_j - \mu_0) \quad 2.1$$

onde

C_i é chamada soma acumulada incluindo a j-ésima amostra, pois combinam as j-ésimas informações das diversas amostras anteriores. Os gráficos de CUSUM são mais eficientes que os de Shewhart para detectar pequenas mudanças do processo. Além disso, são particularmente eficazes em amostras com observações individuais.

Se o processo permanece sob controle para o valor desejado μ_0 , as somas acumuladas definidas pela equação 2.1 são um percurso aleatório com média zero. Porém, se a média muda para algum valor acima ($\mu_1 > \mu_0$), então uma tendência ascendente se desenvolverá na soma acumulada C_i . Reciprocamente, se a média muda para algum valor abaixo ($\mu_1 < \mu_0$), a soma acumulada C_i terá

uma direção negativa. Por esta razão, se nos pontos demarcados aparecer uma tendência para cima ou para baixo, deve-se considerar isto como uma evidência de que a média do processo mudou e uma busca das causas especiais deve ser realizada.

Há duas maneiras de representar as CUSUMs: a tabular (algorítmico) e a máscara V , sendo a primeira preferida por Montgomery (2004). O gráfico CUSUM tabular é usado com base em observações individuais ($n=1$) ou em amostras que contêm mais de uma unidade ($n>1$) de tamanhos constantes (n) ou variáveis (n_i). Nesse caso, a decisão sobre o processo é baseada na informação acumulada de diversas amostras e não apenas na última delas. Acumulando-se dessa forma a pequena evidência que cada amostra fornece sobre o processo, consegue-se maior rapidez na sinalização de pequenos desajustes.

Na CUSUM tabular, são estabelecidas a $S_H(i)$ (CUSUM unilateral superior) e a $S_L(i)$ (CUSUM unilateral inferior), até a amostra i . Assim, após o cálculo da LM e dos limites de controle, deve-se representar os m valores de $S_H(i)$ e $S_L(i)$ no gráfico e verificar se o processo pode ser considerado sob controle ou se existem pontos fora dos limites. Se o processo permanecer sob controle, então os valores das CUSUMs tabulares ficarão em torno de zero. Caso contrário, se eles deslocarem para cima ou para baixo, então ocorrerão valores cada vez mais positivos ou negativos, com tendência à detecção de causas especiais, para a média.

Os valores de $S_H(i)$ e $S_L(i)$ acumulam desvios que são maiores do que K (valor de referência), com ambas as quantidades reajustadas para zero, caso se tornarem negativos, sendo os valores iniciais de $S_H(0)$ e $S_L(0)$ iguais a zero. Se pelo menos um $S_H(i)$ ou $S_L(i)$ exceder o valor de H (intervalo de decisão), então o processo será considerado como fora de controle, sendo: $LIC=-H$, $LM=0$ e $LSC=H$. Para a construção do gráfico de controle CUSUM tabular, os valores de $S_H(i)$ recebem sinais positivos e, os de $S_L(i)$, negativos.

Na literatura, são apresentados alguns trabalhos e revisões que relatam à importância do gráfico de controle CUSUM tabular dentro do CEQ, dentre os

quais são citados os seguintes: King (1995), Montgomery e Runger (2003), Costa et al. (2004), Montgomery (2004) e Ribeiro Júnior et al. (2006).

As CUSUMs tabulares da medida individual ($n=1$) em relação à média de controle (μ_0), são calculadas por meio das seguintes expressões:

$$S_H(i) = \text{máximo} [0; y_i - (\mu_0 + K) + S_H(i-1)], \text{ para } i = 1, 2, \dots, m;$$

$$S_L(i) = \text{máximo} [0; (\mu_0 - K) - y_i + S_L(i-1)], \text{ para } i = 1, 2, \dots, m.$$

Os valores de K e H obtidos em função de σ_0 são dados por:

$$K = k^* \sigma_0;$$

$$H = h^* \sigma_0, \text{ em que:}$$

$$k^* = \frac{\delta}{2} = \text{valor padronizado de K};$$

$$h^* = \text{valor padronizado de H}.$$

O valor padronizado de K (k^*) é geralmente escolhido como metade do valor do deslocamento (δ) para o qual se está interessado. Portanto o valor da média que corresponde ao estado de fora de controle é dado por: $\mu_0 \pm \delta \sigma_0$. Os valores mais usuais de k^* estão entre 0,25 ($\delta=0,5$) e 1,5 ($\delta=3$). Já os valores mais usuais de h^* se encontram entre 3 e 6.

2.2 – Redes Neurais Artificiais

Rede neural artificial (RNA) é uma técnica de inteligência artificial que tenta simular de uma maneira simplificada, o funcionamento do cérebro humano. Ela é capaz de reconhecer padrões, extrair regularidades e detectar relações subjacentes em um conjunto de dados aparentemente desconexo. Além disso, ela apresenta habilidade de lidar com dados difíceis de serem analisados e de prever sistemas não lineares (MELLO, 2004).

De acordo com Haykin (2000) uma RNA é um processador distribuído de forma paralela e constituído de unidades de processamento simples, que tem a propensão natural de armazenar conhecimento experimental e de torná-lo disponível para o uso. Portanto, as RNAs apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento por meio da experiência.

As RNAs foram desenvolvidas originalmente na década de 40, pelo neurofisiologista Warren McCulloch, do MIT, e pelo matemático Walter Pitts, da Universidade de Illinois, os quais fizeram uma analogia entre as células nervosas vivas e o processo eletrônico. O trabalho consistiu num modelo de resistores variáveis e amplificadores que representam conexões sinápticas de um neurônio biológico (TAFNER, 2008).

Desde então diversos modelos de RNAs têm surgidos com o propósito de desenvolver e aplicar esta tecnologia. Algumas destas propostas tendem a aperfeiçoar mecanismos internos da rede neural para aplicação na indústria e negócios, outras procuram aproximá-las ainda mais dos modelos biológicos originais (TAFNER, 2008).

A RNA é formada a partir da combinação de diversos neurônios organizados em camadas: de entrada, intermediária (oculta) e outra de saída. As entradas simulam uma área de captação de estímulos, que podem ser conectadas em muitos neurônios, e resultarem em uma série de saídas. Na camada intermediária é feita a maior parte do processamento da rede por meio de conexões ponderadas. Estas conexões comparadas com o sistema biológico representam o contato dos dendritos com outros neurônios, formando assim as sinapses. A função da conexão em si é transformar o sinal de saída de um neurônio em um sinal de entrada do outro, ou ainda, orientar o sinal de saída para o mundo externo ou real. As diferentes possibilidades de conexões entre as camadas dos neurônios podem gerar muitas estruturas diferentes (TAFNER, 2008).

Na prática, as RNAs consistem de camadas de elementos chamados de neurônios artificiais, que são uma estrutura lógico-matemática que procura

simular o comportamento e as funções de um neurônio biológico. Cada camada é interligada a outra por meio de conexões, chamadas de sinapses artificiais, as quais permitem que informações possam fluir entre as várias camadas dos neurônios.

O neurônio artificial adaptado de Haykin (2001) (Figura 2) representa uma região onde as informações são processadas. Seus três elementos básicos são os pesos sinápticos e as funções de soma e de ativação. As conexões entre os neurônios, denominadas de pesos sinápticos, são responsáveis pelo armazenamento das informações. Além disso, elas definem o efeito que a saída de um neurônio exerce sobre a entrada do seguinte. Os pesos sinápticos são de grande importância para uma rede neural, pois determinam toda a manipulação dos valores da rede.

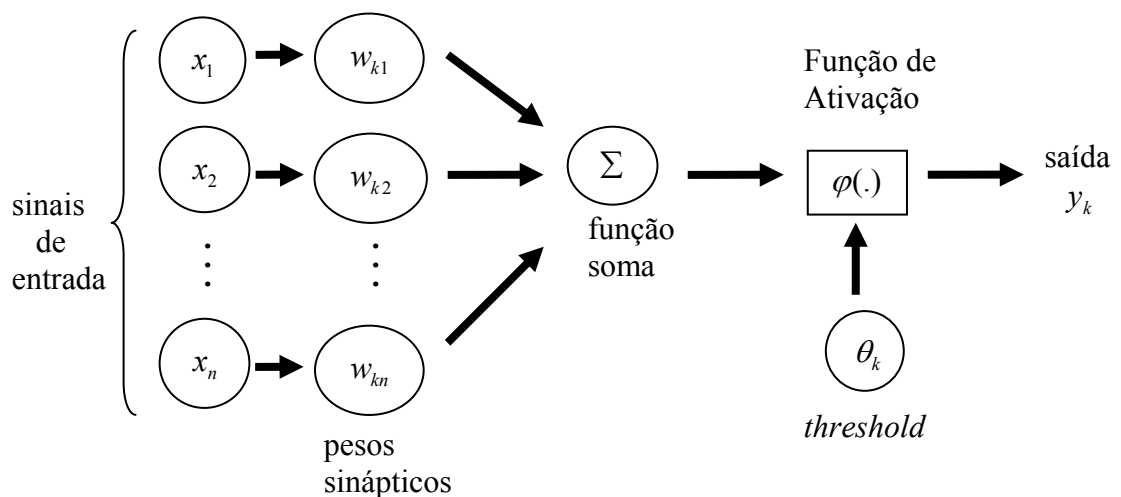


Figura 2. Estrutura básica de um neurônio artificial

Na Figura 2, x_j é o j -ésimo elemento do vetor de entrada, w_{kj} é peso conectando a j -ésima entrada ao neurônio k , Σ é a função soma, $y_k = \varphi(\cdot)$ é a saída gerada pelo neurônio k , θ_k é o *threshold* aplicado ao neurônio k , para $j = 1, 2, \dots, n$.

A função soma representada na equação 2.2 processa os estímulos (x_j) ponderados pelos respectivos pesos (w_{kj}), onde v_k é a saída gerada por cada neurônio da camada anterior:

$$v_k = \sum_j w_{kj} x_j \quad 2.2$$

Já a função de ativação representada na equação 2.3, também chamada de função de transferência é limitada à amplitude do intervalo do sinal de saída do neurônio para algum valor finito, geralmente no intervalo normalizado de 0 a 1 ou de -1 a 1:

$$y_k = f(v_k) \quad 2.3$$

onde: y_k são estímulos dos sinais de saída.

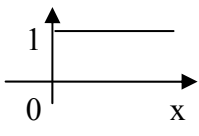
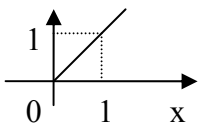
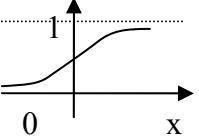
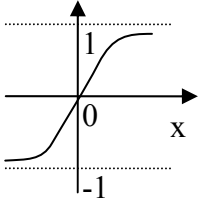
Além dos elementos básicos já citados, o neurônio pode ainda apresentar um *viés* que tem o efeito de aumentar ou diminuir a entrada líquida da função de ativação (HAYKIN, 2001). O *viés* é um meio alternativo de representar o limiar (*threshold*) de um neurônio, ele é tomado fora do corpo do neurônio e conectado usando uma entrada adicional.

Dentre as principais funções de ativação utilizadas, as básicas são: degrau, linear, sigmóide e tangente hiperbólica. A escolha do tipo da função de ativação na concepção de uma rede neural é de extrema importância, uma vez que é usual normalizar os dados de entrada da rede para posterior aplicação. O processo de normalização visa adequar os dados a serem utilizados entre um limite de valores mínimos e máximos, os quais serão manipulados no processo de definição do menor erro de saída.

A função de ativação $\varphi(\cdot)$ é também conhecida como função restritiva, já que limita o intervalo possível da saída (y) a um valor finito. A definição da função de ativação a ser utilizada pela rede neural é de extrema importância. Existem diversas funções matemáticas utilizáveis, entretanto, as funções do tipo sigmóide são as de maior utilização no momento, pois possuem a importante característica de serem diferenciáveis, que é uma propriedade valiosa na definição dos pesos. Esta característica permite a construção dos algoritmos de aprendizagem e memorização, pois dada uma equação, podem ser determinados os seus pontos de máximo e mínimo por meio da resolução das equações

diferenciais encontradas. As principais funções de ativação podem ser resumidas conforme apresentado na Tabela 1.

Tabela 1. Tipos de funções de ativação utilizadas em RNA

| Tipo | Representação gráfica | Função |
|----------------------|---|---|
| Degrau |  | $f(x) = \begin{cases} 1 & \text{se } x \geq 0 \\ 0 & \text{se } x < 0 \end{cases}$ |
| Linear |  | $f(x) = \begin{cases} 1 & \text{se } x \geq \frac{1}{2} \\ x & \text{se } -\frac{1}{2} < x < \frac{1}{2} \\ 0 & \text{se } x \leq -\frac{1}{2} \end{cases}$ |
| Sigmóide |  | $f(x) = \frac{1}{1 + \exp(-\alpha x)}$ |
| Tangente Hiperbólico |  | $f(x) = \frac{e^{\alpha x} - e^{-\alpha x}}{e^{\alpha x} + e^{-\alpha x}}$ |

A função degrau é o tipo mais simples de função de ativação. Sua resposta pode assumir dois valores: 0 ou 1. Normalmente é utilizada para criar neurônios que tomem decisões binárias, como nos classificadores. É ilimitada, porém não é derivável. Já a função linear também não é limitada. Neurônios com esta função de propagação podem ser utilizados como aproximadores lineares. A função sigmóide assume valores em um intervalo contínuo entre 0 e 1. É uma função de ativação muito adotada em redes neurais em virtude de ser contínua, monotônica, não linear e facilmente diferenciável em qualquer ponto. A função tangente hiperbólica mapeia a entrada dos neurônios no intervalo de -1 a 1. Possui as mesmas características e emprego da função sigmóide, possibilitando que as saídas sejam simétricas.

Os diferentes tipos de conexões entre os neurônios de uma rede determinam a topologia (arquitetura) da mesma. As principais topologias de rede são as alimentadas adiante (*feed-forward*) e as recorrentes.

Nas redes alimentadas adiante (*feed-forward*), os neurônios estão dispostos em camadas conectadas por pesos unidirecionais na direção entrada-saída, ou seja, as conexões ocorrem apenas entre as camadas diferentes e subseqüentes. Esta estrutura é totalmente conectada uma vez que todas as saídas dos neurônios de uma camada estão conectadas com as entradas de todos os neurônios da camada seguinte.

Já nas redes recorrentes, um neurônio pode receber entradas de qualquer outra camada. Destas fazem parte as redes com realimentação, nas quais os neurônios de entrada recebem sinais vindos diretamente dos neurônios de saída.

A rede neural deve ter a capacidade de generalização, ou seja, ela deve ser capaz não apenas de classificar as entradas para as quais ela recebe treinamento, mas também de generalizar e classificar entradas que não tenham sido apresentadas. Isto é possível graças a um processo de aprendizagem ao qual a rede é submetida. Esta propriedade permite que a rede encontre respostas corretas mesmo quando os dados disponíveis para as entradas estão incompletos. O desenvolvimento de uma rede neural, apesar de ser um processo de tentativa e erro, envolve a escolha da topologia, da função de transferência e do algoritmo de aprendizagem.

A propriedade mais importante das redes neurais é a habilidade em aprender de seu ambiente e com isso de melhorar o seu desempenho. Isto é feito por meio de um processo iterativo de ajustes aplicados aos seus pesos, conhecido como treinamento. O aprendizado ocorre quando a rede neural atinge a solução generalizada para uma classe de problemas, o que acontece internamente por meio do ajuste dos pesos sinápticos das conexões, durante a exposição dos exemplos, em resposta à quantidade de erros gerados pela rede. Portanto, a rede neural é capaz de modificar-se em função da necessidade de aprender a informação que lhe foi apresentada (TAFNER, 2008).

As RNAs são treinadas para aprenderem em função dos dados de entrada. Assim como o cérebro humano, elas aprendem a partir das experiências e não por meio da programação. Por este motivo, deve-se tomar bastante cuidado com a formação do conjunto de treinamento, que deve ser gerado a partir de dados históricos, ou seja, a partir de experiências e de fatos ocorridos no passado.

Como já mencionado anteriormente, a rede deve ser capaz de generalizar. Mas, ao mesmo tempo, deve-se tomar cuidado para que não aconteça um número excessivo de treinamentos (*overfitting*) e, conseqüentemente, a memorização dos dados. Se uma RNA é submetida a um número excessivo de treinamentos, ela perde a capacidade de reconhecer padrões fora do conjunto de treinamento. Para evitar esta situação, deve-se ter um conjunto de teste com dados diferentes do conjunto de treinamento, para que a rede seja capaz de classificá-los corretamente, mostrando flexibilidade e capacidade de generalização.

De acordo com Haykin (2001), os principais modelos de aprendizagem são: supervisionada, não supervisionada e híbrida. A aprendizagem supervisionada (aprendizagem com professor) é caracterizada pela utilização de um agente externo que indica à rede a resposta desejada para o padrão de entrada. O ajuste dos pesos ocorre quando o sistema compara a saída da rede com a resposta desejada e previamente conhecida. Esse modelo de aprendizado caracteriza-se pela disponibilidade de um conjunto de treinamento composto por dados de entrada previamente classificados. Os estímulos de entrada, disponíveis no conjunto de treinamento, são apresentados à rede, a qual calcula uma resposta utilizando como parâmetros, os valores dos pesos atuais. A partir daí, faz-se uma comparação entre as respostas oferecida pela rede e a desejada àqueles estímulos. Com base na similaridade entre elas, os pesos são ajustados. Este procedimento ocorre até que os pesos possibilitem a rede classificar o mais corretamente possível as entradas de treinamento e, conseqüentemente, as entradas apresentadas durante a fase de teste.

Entre os principais modelos de redes neurais existentes, destacam-se o *perceptron* proposto por F.Rosenblatt em 1957 e o de multicamadas propostos por P. Werbos, D. Parker, D.Rumelhart em 1974.

O *perceptron* é uma forma simples de uma rede neural utilizada para a classificação de padrões. É constituído por apenas uma camada de neurônios com pesos e *bias*, incluindo o caso particular de apenas um neurônio, limitado a classificar padrões em duas classes. O *perceptron* adota uma soma ponderada das suas entradas ou sinapses, enviando o resultado de uma unidade positiva se esta soma for superior a um determinado limiar. Caso contrário o valor será zero ou uma unidade negativa, dependendo da função de ativação escolhida na modelagem da rede. Pode-se notar com isso que as funções de ativação do *perceptron* podem ser a degrau, para casos onde a resposta deve ser 0 ou +1, ou a degrau simétrico, para situações onde a saída deve assumir os valores +1 ou -1.

Tipicamente as entradas e os pesos das conexões, neste modelo, assumem valores reais, que podem ser positivos ou negativos. Se uma entrada correspondente a uma característica, tender a disparar o peso sináptico associado a esta entrada, terá seu valor como sendo positivo. Pode-se então dizer que tal característica tem função excitatória sobre o neurônio. Se o oposto ocorrer, ou seja, se o neurônio for inibido por uma determinada característica, o valor do peso sináptico associado a esta entrada será negativo.

O processo de atualização da matriz de pesos atinge o seu ótimo quando a redução do gradiente do erro quadrático entre as saídas desejadas e as obtidas pelo *perceptron* atinge seu mínimo. O *perceptron*, depois de treinado, separa linearmente duas coleções por meio de um hiperplano contido no hiperespaço destes conjuntos. Este hiperplano é chamado de superfície de decisão. A aprendizagem do ponto de vista geométrico, pode então ser resumida a um processo de determinados deste hiperplano no espaço multidimensional das entradas.

Já o *perceptron* multicamada (*backpropagation* ou retropropagação), possui uma ou mais camadas de neurônios intermediários entre as camadas de entrada e saída da RNA. Uma das grandes vantagens apresentada por esse modelo é que ele possui um conjunto de equações bem definidas e explícitas para a correção dos pesos da rede. Ele realiza a retropropagação do erro gerado na comparação entre a saída e a desejada, com o objetivo de minimizar o erro total

da saída gerada pela rede. O treinamento de uma rede com o uso deste algoritmo envolve três etapas: a propagação dos dados da camada de entrada para a de saída da rede, o cálculo e a retropropagação do erro gerado pela rede e o ajuste dos pesos (FAUSETT, 1994).

O algoritmo *backpropagation* é baseado na regra delta de Widrow-Hoff, desenvolvida para ajustes dos pesos sinápticos na entrada de cada neurônio (elementos de processamento), tendo como referência o erro existente entre a saída real e a desejada do neurônio, quando um dado vetor é apresentado à sua entrada. A correção do peso é feita utilizando-se o método de otimização de mínimos quadrados, com o objetivo de encontrar um valor para o peso que minimize o erro na saída da rede.

Não é recomendada a utilização de um grande número de camadas escondidas. A principal camada responsável pela informação (camada de saída), precisa de erro cometido pela rede. Já as camadas ocultas recebem as estimativas do erro gerado na camada de saída. De uma maneira geral e para a maioria dos problemas, utiliza-se de uma a três camadas escondidas.

Para determinar o momento em que o treinamento da RNA deve ser encerrado, existem vários métodos. Uma boa determinação destes critérios é fundamental para um bom treinamento e, conseqüentemente, para uma boa generalização. Os critérios de parada mais utilizados são: número de ciclos e erro.

O número de ciclos de treinamento representa o número de vezes em que o conjunto de treinamento é apresentado à rede. De acordo com Lawrence e Giles (2000), um número excessivo de ciclos pode levar à rede, a perda do poder de generalização (*overfitting*). Um pequeno número de ciclos, a não obter o melhor desempenho (*underfitting*).

O critério de parada por erro consiste em encerrar o treinamento quando o erro médio quadrático (MSE) ficar abaixo de um valor pré-definido. Um MSE muito pequeno não implica necessariamente numa boa generalização. A definição do melhor valor depende diretamente do problema.

Devido ao rápido desenvolvimento dos métodos e ferramentas relacionadas à RNA, esta tem gerado um enorme interesse em soluções de problemas ligados à manufatura (FRANÇA, 2005). Nathan et al. (1999) conduziram estudos, com RNAs para monitorar a potência e a temperatura na região de corte em um processo de retificação cilíndrica. Neste estudo, as RNAs apresentaram sucesso na predição de momento de início da queima.

As RNAs são utilizadas em áreas relacionadas à manufatura, tais como: planejamento e controle de processos, dimensionamento de sistemas, tomada de decisão operacional, gerenciamento de robôs, monitoramento e outras.

A principal limitação das RNAs é a de não ter capacidade de explicação (DANDOLINI, 1997). Os dados entram nas redes e uma previsão sai, mas o tipo de relacionamento entre as variáveis não é revelado, assim como os detalhes de como elas se relacionam com os dados para se chegar às conclusões (FRANCIS, 2001). Por este motivo, as RNAs são recomendadas para serem aplicadas em áreas de conhecimento, que não necessitam de modelos precisos da realidade física do problema ou onde os resultados do modelo são mais importantes que a compreensão de seu funcionamento.

2.3 – Regressão Logística

Este método, assim como as regressões lineares ou não lineares, estuda a relação de uma variável dependente em função de uma ou mais variáveis independentes. A diferença se deve ao fato de que na regressão logística as variáveis dependentes estão dispostas em categorias, enquanto na regressão linear estas variáveis consistem de dados contínuos. Outra diferença é que na regressão logística, a resposta é expressa por meio de uma probabilidade de ocorrência, enquanto na regressão linear obtém-se um valor numérico (HOSMER e LEMESHOW, 1989).

A regressão logística encontra amplas aplicações em situações na qual o principal objetivo é identificar o grupo ao qual um objeto (indivíduo, produto ou serviço) pertence. Aplicações potenciais incluem prever o sucesso ou o fracasso

de um novo produto, decidir se um estudante deve ser aceito em uma faculdade, classificar estudantes quanto a grupos vocacionais, determinar a categoria de risco de crédito de uma pessoa ou prever se uma empresa terá sucesso ou não. Em cada caso, os objetos recaem em grupos, e é desejado que a pertinência a um deles possa ser prevista ou explicada por um conjunto de variáveis independentes selecionado pelo pesquisador (HAIR JUNIOR et al., 2005).

Na regressão logística as variáveis dependentes podem estar dispostas em duas ou mais categorias. Nesta abordagem existem três procedimentos distintos para manipular os dados, sendo denominados de regressões logísticas binária, ordinária e nominal.

Quando a variável dependente Y é binária, ela aceita apenas dois níveis, como sim ($Y = 1$) ou não ($Y = 0$). Já quando ela é ordinária, têm-se três ou mais níveis com uma ordenação natural como por exemplo em pequeno, médio e grande, ou classificações como ruim, regular, bom e excelente. No caso da nominal, ela tem mais de três níveis sem nenhuma ordenação. Um exemplo é a classificação de algum objeto em azul, preto, amarelo e vermelho, ou a previsão do tempo como ensolarado, nublado e chuvoso. Em todos os casos a variável dependente Y é estudada em função de uma ou mais variáveis independentes X_s . A regressão logística é definida a seguir:

$$g(P_i) = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \dots + \beta_p x_{pi} + e_i$$

onde:

$g(P_i)$ é função de ligação, para $i = 1, 2, \dots, n$;

P_i é probabilidade de $Y = 1$ na i -ésima observação;

β_0 é constante da regressão;

β_j é coeficientes de regressão, para $j = 1, 2, \dots, p$;

x_{pi} é valor da p -ésima variável independente X na i -ésima observação;

e_i é erro aleatório não observável associado a i -ésima observação.

A função de ligação neste procedimento garante que a resposta seja expressa como uma probabilidade, pois esta função trabalha em um intervalo de 0 e 1. Existem basicamente três tipos de função de ligação: logit, probit (normit) e gompit. Estas funções são os inversos das funções de distribuição logística, normal e Gompertz, respectivamente (Tabela 2).

Tabela 2. Funções de ligação mais comumente utilizadas

| Nome da Função de Ligação | Função de Ligação | Distribuição | Função de Distribuição |
|---------------------------|-------------------------------|--------------|-----------------------------|
| Logit | $p(P_i) = \ln(P_i/(1 - P_i))$ | Logística | $p(y_i) = 1/(1 + e^{-y_i})$ |
| Normit (probit) | $p(P_i) = \phi^{-1}(P_i)$ | Normal | $p(y_i) = \phi(y_i)$ |
| Gompit | $p(P_i) = \ln(-\ln(1 - P_i))$ | Gompertz | $p(y_i) = 1 - e^{-e^{y_i}}$ |

Em regressão logística há uma série de gráficos, testes de ajuste e outras medidas para averiguar a validade do modelo. Estas estatísticas permitem identificar as variáveis que não se ajustam bem ou que têm forte influência sobre a estimação dos parâmetros.

O teste de *Hosmer-Lemeshow* avalia a qualidade de ajuste, comparando as frequências observadas às esperadas, por meio da distribuição de qui-quadrado (HOSMER e LEMESHOW, 1989).

No modelo de regressão logística binária com uma única variável independente e com função de distribuição logística, utiliza-se as probabilidades de um valor observado pertencer a uma das duas populações em que se quer diferenciar. A probabilidade de um elemento amostral pertencer à população 1 é estimada por:

$$\hat{p}(Y = 1) = \text{prob}(1 / X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}} = \frac{e^{(\beta_0 + \beta_1 x)}}{1 + e^{(\beta_0 + \beta_1 x)}}$$

Conseqüentemente, a probabilidade desse elemento pertencer à população 2 é estimada por:

$$\hat{p}(Y = 0) = 1 - \hat{p}(Y = 1)$$

Em linhas gerais, o modelo logit é escrito da seguinte forma: seja Y uma variável dependente que assume o valor 1, se o elemento amostrado pertencer a população 1; e assume o valor 0 se pertencer à população 2. O modelo de regressão linear simples, para o logaritmo da razão entre as probabilidades de Y assumir os valores 1 e 0, é dado por:

$$\ln\left(\frac{p(Y = 1 | X)}{p(Y = 0 | X)}\right) = \beta_0 + \beta_1 x$$

A regressão logística fornece estimativas das probabilidades de Y assumir o valor 1 em função de cada valor de X e como a probabilidade varia de 0 a 1, então pode-se utilizar essas estimativas para classificar as observações em $Y = 0$ (população 1) e $Y = 1$ (população 2). O procedimento prevê a estimativa da probabilidade de que o evento pertencerá ou não à população 1. Se a probabilidade prevista for menor que 0,5, então a previsão será que pertence. Caso contrário será que pertence à população 2 (HAIR JUNIOR et al., 2005).

2.4 – Análise Discriminante

Um problema comum nas áreas da análise de dados é quando há dois ou mais grupos de objetos formando amostras de duas ou mais populações, para as quais um número de características foi medida e deseja-se classificar objetos similares desconhecidos baseados no mesmo conjunto de características. Um método estatístico que tem sido aplicado para esse propósito é o da análise discriminante (FERGUSON, 1994).

Para sua aplicação, é necessário que os grupos para os quais cada elemento amostral possa ser classificado, sejam pré-definidos, ou seja, conhecidos *a priori* por meio de várias características. Este conhecimento permite a elaboração de uma função matemática chamada de regra de classificação ou discriminação, que é utilizada para classificar novos elementos

amostrais nos grupos já existentes (MINGOTI, 2005). Estas funções são construídas de modo que os escores dos elementos de cada grupo se concentrem em torno do escore médio, fazendo com que a sobreposição dos escores dos elementos de diferentes grupos seja minimizada (LACHENBRUCH, 1975). A análise discriminante implica na estimativa das densidades de probabilidades específicas nas diferentes populações. Todo processo de tomada de decisão traz consigo um possível erro de decisão. O objetivo, portanto, é o de construir uma regra de classificação que minimize o número de classificações incorretas.

A análise discriminante é útil quando o pesquisador está interessado em compreender diferenças de grupos ou de classificar corretamente objetos em grupos ou classes. Portanto, a análise discriminante pode ser considerada um tipo de análise de perfil ou uma técnica preditiva analítica (HAIR JUNIOR et al., 2005).

Para aplicar a análise discriminante, o pesquisador deve primeiramente especificar quais variáveis respostas serão avaliadas, o tamanho da amostra para a estimação das funções discriminantes e a divisão da amostra para fins de validação. A análise discriminante é muito sensível à proporção entre o tamanho da amostra e o número de variáveis. Muitos estudos sugerem a proporção de 20 observações para cada uma delas. No caso da divisão da amostra para a validação, vários procedimentos têm sido sugeridos. O mais popular envolve desenvolver a função discriminante em um grupo e então testá-la em um segundo grupo (HAIR JUNIOR et al., 2005).

Uma função discriminante linear tem a seguinte forma:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \dots + \beta_p X_{pi}$$

onde Y é o escore discriminante e β_j , para $j = 1, 2, \dots, p$ são os coeficientes da função nas variáveis Y_j . A função retorna um valor Y , para um novo padrão m_j , que é um vetor com as medidas. O valor de Y funcionará como escore de classificação.

No caso em que temos apenas uma variável resposta Y , a função discriminante linear se reduz a:

$$Y_i = \beta_0 + \beta_1 X_{1i}$$

Para classificar um conjunto de observações em duas populações conhecendo-se a distribuição de probabilidades das características medidas nos elementos amostrais de cada população, é possível utilizar o princípio da máxima verossimilhança (CASELLA e BERGER, 2002) para construir uma regra de classificação que minimize a chance em classificar um elemento amostral incorretamente.

Considere as populações 1 e 2 que têm distribuições normais com medias μ_1 e μ_2 e variância comum σ^2 . Neste caso, para cada valor da variável resposta Y , é calculada a razão entre as duas distribuições de probabilidades, chamada razão de verossimilhança entre duas populações e definida por:

$$\lambda(x) = \frac{f_1(x)}{f_2(x)}$$

em que $f_1(x)$ é a função densidade de probabilidade da população 1 e $f_2(x)$ é a função densidade de probabilidade da população 2. Se a função de distribuição de probabilidade for normal, a razão de verossimilhança entre as duas populações torna-se:

$$\lambda(x) = \exp\left\{\frac{-1}{2}\left[\left(\frac{x - \mu_1}{\sigma}\right)^2 - \left(\frac{x - \mu_2}{\sigma}\right)^2\right]\right\}$$

Quando $\lambda(x) > 1$, o valor da função densidade da população 1 calculada para o respectivo valor x é maior do que aquele obtido usando-se a distribuição da população 2. Assim pelo princípio da maior probabilidade (máxima verossimilhança), se $\lambda(x) > 1$, será razoável classificar o dado como vindo da

população 1. Por outro lado, se $\lambda(x) < 1$, será razoável classificá-lo como pertencente à população 2 (MINGOTI, 2005).

Para o uso da análise discriminante, algumas pressuposições devem ser observadas. Segundo HAIR et al. (2005), os resultados são afetados, principalmente, no caso da violação da normalidade multivariada das variáveis respostas. A sua ausência pode causar problemas de estimação da função discriminante. Já a homogeneidade das matrizes de variância e covariância para todos os grupos, ou seja, a variabilidade dentro dos grupos deve ser a mesma.

3 – MATERIAIS e MÉTODOS

3.1 – Simulação dos Dados

Com base nas definições apresentadas em revisão de literatura planejou-se um estudo por simulação de dados objetivando-se avaliar a capacidade das metodologias RL, AD e RNA serem utilizadas no CEQ como alternativas aos gráficos de controle na caracterização de um processo em sob ou fora de controle.

As simulações foram realizadas com o software MatLab para processos independentes e, R, para processos autocorrelacionados. Os dados da variável aleatória normal Y foram simulados para 540 observações individuais ($n=1$) com parâmetros, média $\mu_0 = 1.000$ e desvio-padrão $\sigma_0 = 100$. Destas 540 observações, 432 foram usadas para o treinamento da RNA. Essa escolha foi feita com base em estudos anteriores de forma que a rede não perdesse a sua capacidade de generalizar, uma vez que se o número de amostras fosse muito grande, a rede apenas memorizaria o conjunto de dados treinados (Diagrama 1). As 108 amostras restantes foram usadas para testar a capacidade da RNA e também dos outros métodos propostos: regressão logística e análise discriminante, além dos gráficos de controle de Shewhart, EWMA e CUSUM.

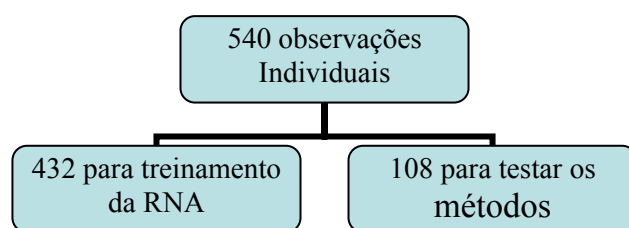


Diagrama 1. Dados simulados

Para o treinamento e teste da eficiência da RNA e dos outros métodos propostos, foram definidas diferentes combinações no conjunto de dados com o objetivo de se ter, além dos pontos sob controle de qualidade com média $\mu_0 = 1.000$, também pontos fora de controle com média $\mu_1 = \mu_0 + \delta\sigma_0$, para valores de δ iguais a 1, 2 e 3, aos quais foram inseridos aleatoriamente aos dados gerados no

meio (posição 0) e no final (posição 1) do processo de geração aleatória das observações. Essa estratégia foi aplicada a quatro conjuntos de dados, um sem a presença ($\rho_i=0$) e os outros três com as seguintes autocorrelações de primeira ordem (0,3; 0,6 e 0,9). No total criaram-se 24 amostras com 540 observações individuais cada, sendo seis sem e 18 com a presença da primeira autocorrelação (Diagrama 2).

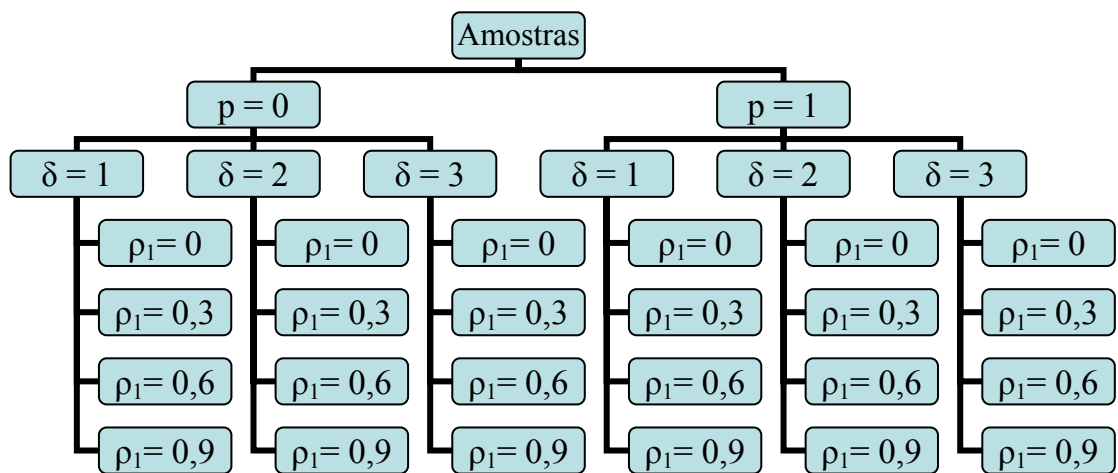


Diagrama 2. Configuração das 24 amostras.

Portanto, foram realizados dois treinamentos. No primeiro, as primeiras 144 das 432 observações estavam dentro de controle estatístico ($\mu_0 = 1.000$ e $\sigma_0 = 100$), seguidas de 144 observações fora de controle estatístico ($\mu_1 = \mu_0 + \delta\sigma_0$ e $\sigma_0 = 100$) para valores de δ iguais a 1, 2 e 3 e de mais 144 observações sob controle estatístico (Figura 3). No segundo, foram 288 observações sob controle estatístico seguido de 144 observações fora de controle estatístico (Figura 4).

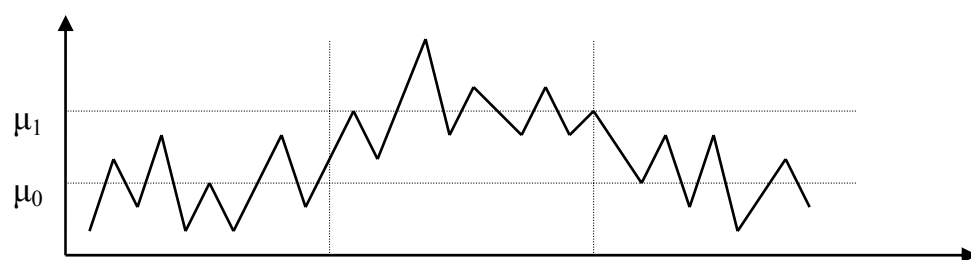


Figura 3. Primeiro treinamento onde as primeiras observações foram simuladas com média μ_0 seguidas de observações com médias μ_1 e μ_0 .

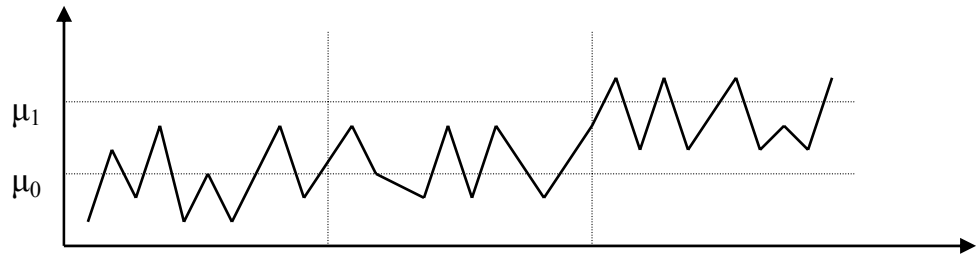


Figura 4. Segundo treinamento onde as primeiras observações foram simuladas com média μ_0 seguidas de observações com média μ_1 .

Já para testar a eficiência de cada um dos métodos com as 108 observações destinadas para este fim, também foi sob duas situações. A primeira em que os pontos fora de controle estatístico estavam no meio do processo: 36 observações sob seguidas de 36 fora e de 36 observações sob. Na segunda situação, os pontos fora de controle ficaram no final do processo: 72 observações sob seguidas de mais 36 fora de controle estatístico.

Os dados sem autocorrelação (independentes) foram simulados a partir da distribuição normal. Já para os dados autocorrelacionados, a simulação foi baseada no processo inverso do modelo auto-regressivo (AR) de séries temporais. Como exemplo deste procedimento, considere o modelo AR (1) para uma variável aleatória Y , com a constante ξ , coeficiente auto-regressivo ϕ_1 e erro aleatório e_i , dado por:

$$y_i = \xi + \phi_1 y_{i-1} + e_i.$$

Quando se tem os valores de y_i , esta equação é utilizada com o objetivo de encontrar os valores de ξ e ϕ_1 que minimizam a soma de quadrados dos erros (e_i), que é distribuído normal e independentemente com média zero e desvio padrão σ_E . O processo chamado anteriormente de “inverso”, consiste na simulação do erro $e_i \sim N(0, \sigma_E)$ e em seguida na atribuição de valores para as constantes ξ e ϕ_1 , de forma a atingir os parâmetros especificados aos mesmos.

3.2 – Gráficos de Controle

Nas construções de cada tipo de gráfico de controle, foram utilizadas constantes pré-determinadas, com objetivo de tentar obter melhor eficiência em cada um dos três tipos propostos. Em todos eles, a linha média (LM) foi igual a $\mu_0 = 1.000$. Para o gráfico de controle Xbarra de Shewhart, foram utilizadas as constantes $k = 1,96$ e $k = 3$ e os valores de δ iguais a 1, 2 e 3. Para o gráfico EWMA foram testadas 36 combinações dos valores de k (1,96 e 3), λ (0,05; 0,20 e 0,40), posição (meio e final) e δ (1, 2 e 3). Para o gráfico CUSUM tubular foram comparados quatro valores de k^* (0,05; 1; 1,5 e 2). A análise de regressão foi feita ao nível de 5% de probabilidade no software Minitab para os alarmes falso e verdadeiro em cada um dos três tipos de gráfico de controle em função das suas respectivas constantes.

O objetivo foi identificar os valores das constantes que proporcionam menor e maior incidência de alarmes falso e verdadeiro, respectivamente. Posteriormente, foram estimados o número médio de amostras para a detecção de um sinal falso ($NMA_{\delta=0}$) ou verdadeiro ($NMA_{\delta>0}$), dados, respectivamente, por:

$$N\hat{M}A_{\delta=0} = \frac{1}{\hat{\alpha}}$$

$$N\hat{M}A_{\delta>0} = \frac{1}{\hat{P}d}$$

Para obtenção das estimativas de α e Pd , $NMA_{\delta=0}$ e $NMA_{\delta>0}$, foram simuladas quatro amostras com 108 observações individuais. Um sem a presença ($\rho_1 = 0$) e três com presença de autocorrelações de primeira ordem (0,3; 0,6 e 0,9). Para as amostras com a presença de autocorrelação foram ajustados modelos de séries temporais do tipo ARIMA de acordo com a metodologia de Box e Jenkins, construído com os ruídos aproximadamente independentes (MONTGOMERY, 2004).

Os gráficos de controle e ajuste do modelo de series temporais ARIMA foram feitos no software Minitab, para as 24 amostras com 108 observações

individuais simuladas. De posse desses resultados, foram calculados os alarmes falso e verdadeiro para a comparação com os métodos propostos.

3.3 – Redes Neurais Artificiais

As RNAs utilizadas neste trabalho foram construídas no software MatLab, que contém funções específicas e pacotes matemáticos para elaboração de redes neurais. Diversas estruturas foram testadas para obter a configuração que fornecesse os melhores resultados para determinar se o processo está sob ou fora de controle.

A rede neural escolhida para o treinamento e validação dos dados do presente trabalho foi a *perceptrons* de múltiplas camadas (algoritmo *backpropagation*) com treinamento por retropropagação, o que minimiza o erro gerado na comparação entre a saída da rede e a desejada. Esta estrutura caracteriza-se por uma rede de múltiplas camadas, com realimentação de entradas. A escolha foi baseada nas ótimas características que este modelo de RNA possui em relação à classificação de padrões. No MatLab existem várias funções de treinamento para implementar este algoritmo.

Neste trabalho foram testadas 576 arquiteturas de RNAs, cada arquitetura foi construída com a combinação de algumas características importantes na construção da RNA, conforme mostradas na Tabela 3.

Tabela 3. Descrição e características das RNAs.

| Característica | Descrição |
|--|--|
| Funções de treinamento | <i>traingdx, traingda, trainlm e trainbr</i> |
| Número de camadas | 3 e 4 |
| Número de neurônios por camada | 1, 5, 7, 11, 14, 17, 20 e 27 |
| Número de dados p/ treinamento e validação | 180, 540 e 1620 |
| Porcentagem utilizada p/ treinamento | 40%, 60% e 80% |

Para escolha de apenas uma das arquiteturas testadas, foi avaliado o desempenho de acordo com o menor erro médio quadrático (MSE), dado pela equação 3.1 onde y_{esp} é o valor esperado, e y_{obs} é o valor observado na saída da RNA, como segue:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_{esp} - y_{obs})^2 \quad 3.1$$

A fim de relacionar as características das RNAs com o MSE foi estimada uma equação de regressão linear múltipla do mesmo em função dos números de camadas, neurônios e dados e da porcentagem para cada função de treinamento (Tabela 3), cujos coeficientes foram testados pelo teste t a 5% de probabilidade.

A função de ativação (transferência) entre as camadas foi a sigmóide, uma vez que os dados de entrada e saída da rede foram normalizados entre os valores 0 e 1. No MatLab a função de transferência *tansig* implementa esta função de ativação.

O número de ciclos de treinamento foi fixado em 1.250 épocas, para que o número de vezes que o conjunto de treinamento fosse apresentado à rede não se tornasse excessivo, o que poderia levar à perda do poder de generalização (*overfitting*). Por outro lado, não ser demais de tal forma a não atingir o seu melhor desempenho (*underfitting*).

Com os parâmetros da RNA escolhidos, foram realizados o treinamento e a validação da rede. Para avaliar o desempenho da RNA, foram calculados os alarmes falso e verdadeiro nas mesma 108 amostras utilizadas pelos outros métodos (AD e RL), para os 24 conjuntos de dados com observações individuais.

3.4 – Regressão Logística

A regressão logística binária procurou estabelecer uma relação de causa e efeito entre a classificação, agora denominado Y, do processo (0 = sob e 1 = fora) em função dos valores observados da característica de qualidade, agora

denominada de variável de controle X , por meio da criação de uma variável contínua não observável ou função de ligação dada por:

$$P(y_i = 1) = g(p_i) = \beta_0 + \beta_1 x_i + e_i$$

Desse modo, o objetivo foi de modelar a probabilidade de sucesso p_i em função dos níveis de X , ou seja, de estimar p_i da mudança de estado ($Y = 0$ para $Y = 1$) para cada x_i . A mudança de estado pode ser denominada como uma quebra de tolerância (T) do elemento amostral para um dado nível x_i . Assim, T será uma v.a.c para cada nível x_i . Portanto, cada elemento amostral mudará de estado, ou seja, terá sucesso, quando a sua tolerância t_i for menor do que o efeito do nível x_i , sendo:

$$p_i = P(T \leq x_i) = F(x_i)$$

Se T segue distribuição logística com os parâmetros de μ_T e σ_T , então será adotado o modelo logit dado por:

$$g(p_i) = \text{logit}(p_i) = \ln\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_i + e_i.$$

A equação de regressão ajustada do modelo logit é dada por:

$$g(\hat{p}_i) = \text{logit}(\hat{p}_i) = \ln\left(\frac{\hat{p}_i}{1-\hat{p}_i}\right) = \hat{\beta}_0 + \hat{\beta}_1 x_i,$$

em que:

$$\hat{p}_i = \frac{e^{\text{logit}(\hat{p}_i)}}{1 + e^{\text{logit}(\hat{p}_i)}}$$

Para o modelo logit, o método usado na estimação do vetor β é o da máxima verossimilhança, cuja solução do logaritmo da função de

verossimilhança é obtida geralmente por processos iterativos do tipo Newton-Raphson. Maiores detalhes, dentro do contexto dos modelos lineares generalizados, podem ser consultados em Demétrio (2002).

Para a avaliação da significância do coeficiente β_1 , testaram-se as hipóteses $H_0: \beta_1 = 0$ vs $H_a: \beta_1 \neq 0$ pelo teste z, dado por:

$$z_{\text{cal}} = \frac{\hat{\beta}_1}{s(\hat{\beta}_1)}$$

Se $|z_{\text{cal}}| \geq z_{\text{tab}}$ ou se p-valor $\leq \alpha$, rejeita-se H_0 .

Para a avaliação da qualidade do ajustamento da equação de regressão, utilizou-se o teste de *Hosmer-Lemeshow* para k classes, sendo dado por:

$$G_{\text{HL}}^2 = \sum_{i=1}^k \frac{(FO_i - FE_i)^2}{FE_i \times \left(1 - \frac{FE_i}{r_i}\right)}$$

em que:

FO_i = frequência observada para o nível $y_i = 0$ ou $y_i = 1$;

$FE_i = \hat{p}_i \times r_i$;

r_i = número de observações por x_i .

Se $G_{\text{HL}}^2 \geq \chi_{\text{tab}}^2$ com $k-2$ gl ou se p-valor $\leq \alpha$, rejeita-se H_0 (bom ajuste dos dados ao modelo proposto).

O modelo logit foi ajustado no software Minitab. As amostras com probabilidades de fracasso foram classificadas como sob controle e as de sucesso como fora de controle.

Para comparar regressão logística com os demais métodos obteve-se o percentual de classificação correta. Conseqüentemente, foram calculados os alarmes falso e verdadeiro nos 24 conjuntos de dados.

3.5 – Análise Discriminante

A análise discriminante foi utilizada para estabelecer funções de discriminação nas populações 1 (sob controle, $Y = 0$) e 2 (fora de controle, $Y = 1$) definidas a priori com base nos valores simulados das características de qualidade, em função dos valores observados da característica de qualidade.

No software Minitab foram estimadas as funções para as duas populações, nos 24 conjuntos de dados. Foram classificadas como pertencentes à população 1, amostras que tinham maior probabilidade de pertencerem a essa população. Da mesma forma foram classificadas as amostras pertencentes à população 2.

Após a classificação, foi comparado o resultado obtido pela análise discriminante com o resultado esperado. Assim foi possível calcular os alarmes falso e verdadeiro.

3.6 – Medidas de Comparação

Após aplicar cada um dos métodos propostos (gráficos de controle, análise discriminante, regressão logística e RNAs), foram realizadas comparações baseadas nas incidências dos alarmes falso e verdadeiro sobre a classificação dos valores observados em dentro e fora de controle estatístico.

Neste estudo, foi realizada análise de variância de um experimento em duas situações, a primeira a variável resposta é o alarme falso e a segunda a variável resposta é o alarme verdadeiro, as duas situações com quatro fatores (método, delta, posição e autocorrelação) em um delineamento inteiramente casualizado com três repetições por tratamento, cujo teste F será aplicado aos efeitos principais e às interações duplas.

Posteriormente, de acordo com as significâncias das interações, será aplicado o teste de Duncan aos métodos, cujos níveis foram os referentes àqueles que são eficientes em termos das incidências dos alarmes falso e verdadeiro e realizada análise de regressão em função do delta (1, 2 e 3), posição (0 e 1) e

autocorrelação (0; 0,3; 0,6 e 0,9), cujos coeficientes foram testados pelo teste t. Todas as análises foram feitas no software SAEG, a 5% de probabilidade.

4 – RESULTADOS e DISCUSSÃO

4.1 – Gráficos de Controle

4.1.1 - Xbarra de Shewhart

O valor da constante k que proporcionou melhores resultados para os alarmes falso e verdadeiro foi o de 1,96, de acordo com as equações de regressão estimadas (Tabela 4). Para o alarme falso, apenas a constante k influenciou ($P < 0,05$). Para o alarme verdadeiro as constantes δ e k influenciaram ($P < 0,05$), positiva e negativamente, respectivamente.

Tabela 4. Equações de regressão ajustadas em função de k e δ para os alarmes falso e verdadeiro e os respectivos coeficientes de determinação

| Alarme | Equação de regressão | R^2 |
|------------|--|-------|
| Falso | $\hat{\alpha} = 6,68 - 2,17 * k$ | 0,78 |
| Verdadeiro | $\hat{Pd} = 46,1 + 28,8 * \delta - 26,8 * k$ | 0,97 |

*Significativo pelo teste t ($P < 0,05$), $1,96 \leq k \leq 3$, $1 \leq \delta \leq 3$.

De acordo com as baixas estimativas de Pd para $\delta \leq 2$ (Tabela 4), concluiu-se que o gráfico Xbarra foi pouco sensível em detectar diferenças pequenas em torno da média de controle, principalmente para $k = 3$. Por outro lado, os percentuais de α foram pequenos, independentemente dos valores de δ e k (Tabela 4). Portanto, a constante $k = 1,96$ conferiu maior eficiência ao gráfico Xbarra, em termos de alarmes falso e verdadeiro. Conseqüentemente, ela será utilizada na comparação do gráfico de controle Xbarra de Shewhart com os demais métodos propostos no trabalho.

4.1.2 – EWMA

Para o alarme falso a posição (p) e a constante λ , além da interação entre elas, influenciaram ($P < 0,05$) na sua incidência. Já o alarme verdadeiro foi influenciado ($P < 0,05$) pelo δ , λ e interação $\delta \times \lambda$ (Tabela 5).

Tabela 5. Equações de regressão ajustadas em função de p , δ e λ para os alarmes falso e verdadeiro e os respectivos coeficientes de determinação

| Alarme | Equação estimada | R ² |
|------------|---|----------------|
| Falso | $\hat{\alpha} = 0,593 - 0,410 * p - 1,480 * \lambda + 1,060 * p \times \lambda$ | 0,82 |
| verdadeiro | $\hat{P}d = 1,02 + 0,014 * \delta - 1,73 * \lambda + 0,634 * \delta \times \lambda$ | 0,71 |

*Significativo pelo teste t ($P < 0,05$), $0,05 \leq \lambda \leq 0,40$, $1 \leq \delta \leq 3$, $p=0$ e $p=1$

De acordo com as estimativas de α e Pd , a posição 0 (pontos fora de controle no meio do conjunto de dados) forneceu bons percentuais de alarmes verdadeiros (81,48%, 95,56% e 100%) para $\lambda = 0,20$ e δ igual a 1, 2 e 3, respectivamente. Porém, a incidência do alarme falso foi alta. Para $\lambda = 0,40$, os alarmes verdadeiros de 59,56%, 86,32% e 100% para δ igual a 1, 2 e 3, respectivamente e falso de 1,20% foram menores.

No caso da posição 1 (pontos fora de controle no final do conjunto de dados), os percentuais dos alarmes verdadeiros foram semelhantes. Já os dos falsos foram menores. Portanto, em função da interação $p \times \lambda$, serão estudados os valores λ iguais a 0,20 e 0,40 e $k = 1,96$, cuja constante não influenciou ($P > 0,05$) nos alarmes, para a comparação com os demais métodos propostos neste trabalho.

4.1.3 – CUSUM Tabular

De acordo com as equações de regressão ajustadas (Tabela 6), apenas a posição (p) e o δ influenciaram ($P < 0,05$) nas incidências dos alarmes falso e verdadeiro. Portanto, como as constantes k^* e h^* não causaram efeito ($P > 0,05$),

foram utilizados $k^* = 1$ e $h^* = 5$ (valores mais usuais), para a comparação do gráfico de controle CUSUM tabular com os demais métodos.

Tabela 6. Equações de regressão ajustadas em função de p e δ para os alarmes falso e verdadeiro e os respectivos coeficientes de determinação

| Alarme | Equação estimada | R^2 |
|------------|---|-------|
| Falso | $\hat{\alpha} = 60,80 - 35,40 * p$ | 0,97 |
| Verdadeiro | $\hat{Pd} = 87,9 + 3,92 * \delta + 7,85 * p - 2,67 * p \times \delta$ | 0,92 |

*Significativo pelo teste t ($P < 0,05$), $0,05 \leq \lambda \leq 0,40$, $1 \leq \delta \leq 3$, $p=0$ e $p=1$

Apesar das altas estimativas do Pd (acima de 90%), as configurações testadas do gráfico CUSUM tabular se mostraram bastantes sensíveis ao alarme falso, com estimativas entre 91,82 e 99,66% (Tabela 6).

4.2 – Rede Neural Artificial

A menor estimativa do erro médio quadrático estimado ($M\hat{S}E = 0,003$) ocorreu para a arquitetura com três camadas, sendo uma intermediária, 540 dados utilizados para treinamento e validação e com 80% dos dados utilizados para o treinamento, segundo a função de ativação *traingdx*. De acordo com a equação de regressão ajustada para essa função, obteve-se:

$$M\hat{S}E = -0,729 + 0,353 * nc + 0,000083 * n - 0,0521 * p \quad R^2 = 0,912$$

*significativo pelo teste t ($P < 0,05$)

onde nc é o número de camadas, n o número de dados utilizados para treinamento e validação e p é a porcentagem de dados utilizados para treinamento.

Esta arquitetura da RNA será utilizada na comparação com os demais métodos propostos neste trabalho (anexos I e II).

4.3 – Alarmes Falso e Verdadeiro

A comparação entre os métodos foi feita no esquema fatorial 7x3x2x4 referentes aos níveis dos respectivos fatores: método (m), delta (δ), posição (p) para pontos fora de controle e autocorrelação de primeira ordem (ρ_1). Além dos métodos RNA, RL e AD foram testados os seguintes gráficos de controle: Xbarra de Shewhart ($k = 1,96$), EWMA 1 ($k = 1,96$ e $\lambda = 0,20$), EWMA 2 ($k = 1,96$ e $\lambda = 0,40$), CUSUM tabular ($k^* = 1$ e $h^* = 5$).

4.3.1 – Alarme Falso

Pela análise de variância para o alarme falso (Tabela 7) verificou-se que o mesmo sofreu efeito significativo ($P < 0,05$) de todos os fatores e das interações duplas $\rho_1 \times m$, $\delta \times m$, $p \times m$ e $\delta \times p$. Isso implicou que o melhor método dependeu das combinações dos níveis de ρ_1 , δ e p mais apropriadas.

Tabela 7. Análise de Variância para o alarme falso

| FV | GL | SQ | QM | F | p-valor |
|------------------------|-----|----------|---------|---------|---------|
| m | 6 | 16512,42 | 2752,07 | 165,97* | 0,0000 |
| δ | 2 | 431,05 | 215,53 | 13,00* | 0,0000 |
| p | 1 | 2140,76 | 2140,76 | 129,10* | 0,0000 |
| ρ_1 | 3 | 2665,86 | 888,62 | 53,59* | 0,0000 |
| $\rho_1 \times m$ | 18 | 1399,87 | 77,77 | 4,69* | 0,0000 |
| $\delta \times m$ | 12 | 1815,04 | 151,25 | 9,12* | 0,0000 |
| $p \times m$ | 6 | 12421,56 | 2070,26 | 124,85* | 0,0000 |
| $\delta \times p$ | 2 | 148,49 | 74,24 | 4,48* | 0,0136 |
| $\rho_1 \times \delta$ | 6 | 156,40 | 26,07 | 1,57 | 0,1622 |
| $\rho_1 \times p$ | 3 | 11,00 | 3,67 | 0,22 | >0,50 |
| Resíduo | 108 | 1790,85 | 16,58 | | |

* Significativo pelo teste F ($P < 0,05$)

Portanto, com o objetivo de comparar as médias de α entre os métodos procederam-se os desdobramentos das interações duplas significativas ($P < 0,05$), por meio da aplicação do teste de Duncan dentro de cada posição, delta e primeira autocorrelação.

Para pontos fora de controle situados ao meio ($p = 0$) do conjunto de dados, quatro métodos (RNA, GC de Shewhart, GC EWMA 2 e RL) apresentaram as menores ($P < 0,05$) médias, sendo portanto, indicados nessa condição. Para pontos fora de controle situados ao final ($p = 1$), cinco métodos (RNA, GC de Shewhart, GC EWMA 1, GC EWMA 2 e RL) apresentaram as menores ($P < 0,05$) médias (Tabela 8).

Tabela 8. Médias dos alarmes falsos dos métodos para pontos fora de controle estatístico posicionados ao meio (0) e ao final (1) do conjunto de dados

| Método | $p = 0$ | $p = 1$ |
|------------------|---------------------------|---------------------------|
| RNA | <u>3,21 c</u> | <u>2,81 c</u> |
| GC de Shewhart | <u>0,67 c</u> | <u>0,54 c</u> |
| GC EWMA 1 | 8,58 b | <u>2,43 c</u> |
| GC EWMA 2 | <u>2,28 c</u> | <u>1,24 c</u> |
| GC CUSUM Tabular | 54,93 a | 6,43 b |
| RL | <u>2,18 c</u> | <u>2,40 c</u> |
| AD | 9,02 b | 15,05 a |

Médias seguidas pela mesma letra, na coluna, não diferem entre si pelo teste de Duncan ($P > 0,05$)

Portanto, para ambas as posições destacaram-se os seguintes métodos: RNA, GC de Shewhart, GC EWMA 2 ($\lambda = 0,20$) e RL. Isso implicou que a RNA, a RL e o aumento de λ no GC EWMA promoveram, como o GC de Shewhart, baixa incidência do alarme falso, o que possibilita usá-los no monitoramento de processos que apresentam correções, quando necessárias, difíceis, caras e demoradas.

Para o valor de $\delta = 1$, cinco métodos (RNA, GC de Shewhart, GC EWMA 1, GC EWMA 2 e RL) apresentaram as menores ($P < 0,05$) médias. Outros quatro métodos (RNA, GC de Shewhart, GC EWMA 2 e RL) apresentaram ($P < 0,05$)

para $\delta = 2$. Já para o valor de $\delta = 3$, cinco métodos (RNA, GC de Shewhart, GC EWMA 2, RL, e AD) apresentaram as menores ($P < 0,05$) médias (Tabela 9).

Tabela 9. Médias dos alarmes falsos dos métodos para δ iguais a 1, 2 e 3

| Método | $\delta = 1$ | $\delta = 2$ | $\delta = 3$ |
|------------------|--------------------------------|--------------------------------|--------------------------------|
| RNA | <u>4,07 c</u> | <u>2,32 c d</u> | <u>0,94 c</u> |
| GC de Shewhart | <u>0,75 c</u> | <u>0,50 d</u> | <u>0,57 c</u> |
| GC EWMA 1 | <u>4,04 c</u> | 6,05 b c | 6,42 b |
| GC EWMA 2 | <u>1,80 c</u> | <u>1,78 c d</u> | <u>1,69 c</u> |
| GC CUSUM Tabular | 30,06 a | 31,60 a | 30,39 a |
| RL | <u>3,51 c</u> | <u>2,32 c d</u> | <u>1,05 c</u> |
| AD | 24,54 b | 31,60 b | <u>2,17 c</u> |

Médias seguidas pela mesma letra, na coluna, não diferem entre si pelo teste de Duncan ($P > 0,05$).

Novamente, os métodos RNA, RL, GC de Shewhart e EWMA 2 ($\lambda = 0,40$) apresentaram-se eficientes em minimizar as estimativas dos alarmes falso, para valores de δ iguais a 1, 2 e 3.

Portanto, os métodos RNA, RL e EWMA 2 se mostraram eficientes e capazes de substituírem o GC de Shewhart, no monitoramento de processos que visam detectar pequenas ou grandes diferenças em relação à média de controle, em termos da incidência do alarme falso.

Para o conjunto de dados sem a presença da primeira autocorrelação ($\rho_1 = 0$) dois métodos (GC de Shewhart e GC EWMA 2) apresentaram as menores ($P < 0,05$) médias. Para o conjunto de dados com $\rho_1 = 0,3$, três métodos (GC de Shewhart, GC EWMA 2 e RL) não apresentaram alarmes falsos. Entre os que apresentaram, RNA, GC EWMA 1 e AD acarretaram as menores ($P < 0,05$) estimativas. Para o conjunto de dados com $\rho_1 = 0,6$ cinco métodos (RNA, GC de Shewhart, GC EWMA 1, GC EWMA 2 e RL) apresentaram as menores ($P < 0,05$) médias. Os mesmos cinco métodos também apresentaram as menores ($P < 0,05$) médias para $\rho_1 = 0,9$ (Tabela 10).

Tabela 10. Médias dos alarmes falsos dos métodos para autocorrelações iguais a 0; 0,3; 0,6 e 0,9

| Método | $\rho_1 = 0$ | $\rho_1 = 0,3$ | $\rho_1 = 0,6$ | $\rho_1 = 0,9$ |
|------------------|--------------------------------|----------------------------------|----------------------------------|----------------------------------|
| RNA | 8,65 c | <u>0,54 b</u> | <u>0,51 c</u> | <u>2,33 c</u> |
| GC de Shewhart | <u>2,43 d</u> | <u>0 b</u> | <u>0 c</u> | <u>0 c</u> |
| GC EWMA 1 | 15,83 b | <u>2,01 b</u> | <u>2,06 c</u> | <u>2,11 c</u> |
| GC EWMA 2 | <u>7,03 c d</u> | <u>0 b</u> | <u>0 c</u> | <u>0 c</u> |
| GC CUSUM Tabular | 43,31 a | 26,47 a | 26,47 a | 26,47 a |
| RL | 8,68 c | <u>0 b</u> | <u>0 c</u> | <u>0,49 c</u> |
| AD | 16,64 b | <u>3,14 b</u> | 8,16 b | 20,20 b |

Médias seguidas pela mesma letra, na coluna, não diferem entre si pelo teste de Duncan ($P > 0,05$)

Na ausência de autocorrelação ($\rho_1 = 0$) ocorreram mais alarmes falsos em relação à presença das mesmas (0,3, 0,6 e 0,9). De forma contrária à esperada, a violação da pressuposição de independência para a construção dos gráficos de controle, não prejudicou, em média, a incidência dos alarmes falsos. Do mesmo modo, ela favoreceu o monitoramento do processo pelos métodos RNA e RL que se mostraram eficientes e capazes de substituírem o GC de Shewhart, na presença ou não de autocorrelação, em termos da incidência do alarme falso.

4.3.2 – Alarme Verdadeiro

Pela análise de variância para o alarme verdadeiro (Tabela 11), verificou-se que o mesmo sofreu efeito ($P < 0,05$) dos fatores método (m), delta (δ) e autocorrelação (ρ_1) e das interações duplas $\rho_1 \times m$ e $\delta \times m$. Isso implicou que o melhor método também dependeu das combinações apropriadas dos níveis de δ e ρ_1 .

Tabela 11. Análise de Variância para o alarme verdadeiro

| FV | GL | SQ | QM | F | p-valor |
|------------------------|-----|----------|----------|---------|---------|
| m | 6 | 28097,88 | 4682,98 | 82,43* | 0,0000 |
| δ | 2 | 20154,60 | 10077,30 | 177,37* | 0,0000 |
| p | 1 | 2,12 | 2,12 | 0,04 | >0,50 |
| ρ_1 | 3 | 2932,41 | 977,47 | 17,20* | 0,0000 |
| $\rho_1 \times m$ | 18 | 3028,09 | 168,23 | 2,96* | 0,0003 |
| $\delta \times m$ | 12 | 22892,72 | 1907,73 | 33,58* | 0,0000 |
| $p \times m$ | 6 | 147,72 | 24,62 | 0,43 | >0,50 |
| $\delta \times p$ | 2 | 61,25 | 30,62 | 0,54 | >0,50 |
| $\rho_1 \times \delta$ | 6 | 602,80 | 100,47 | 1,77 | 0,1124 |
| $\rho_1 \times p$ | 3 | 136,31 | 45,44 | 0,80 | >0,50 |
| Resíduo | 108 | 6135,87 | 56,81 | | |

* Significativo pelo teste F ($P < 0,05$)

Portanto, com o objetivo de comparar as médias dos alarmes verdadeiros entre os métodos procederam-se os desdobramentos das interações duplas significativas ($P < 0,05$), por meio da aplicação do teste de Duncan dentro de cada delta e autocorrelação.

Para o valor de $\delta = 1$, os métodos RNA, GC CUSUM, GC EWMA 1 e RL apresentaram as maiores ($P < 0,05$) médias. Já os métodos RNA, GC EWMA 1, GC EWMA 2, GC CUSUM, RL e AD apresentaram as maiores ($P < 0,05$) médias para o valor de $\delta = 2$. E para $\delta = 3$, todos os métodos apresentaram médias iguais ($P > 0,05$) entre si. Os métodos RNA, GC EWMA 1, GC CUSUM e RL apresentaram-se eficientes em maximizar as estimativas dos alarmes verdadeiros, para valores de δ iguais a 1, 2 e 3. Já GC EWMA 2 e AD se comportaram de formas iguais ($P > 0,05$) aos melhores somente para δ igual a 2 e 3 (Tabela 12).

Tabela 12. Médias dos alarmes verdadeiros dos métodos para δ iguais a 1, 2 e 3

| Método | $\delta = 1$ | $\delta = 2$ | $\delta = 3$ |
|------------------|--------------------------------|--------------------------------|--------------------------------|
| RNA | <u>90,90 a</u> | <u>96,32 a</u> | <u>98,06 a</u> |
| GC de Shewhart | 5,49 c | 70,24 b | <u>96,11 a</u> |
| GC EWMA 1 | <u>91,04 a</u> | <u>99,44 a</u> | <u>99,83 a</u> |
| GC EWMA 2 | 74,69 b | <u>99,41 a</u> | <u>99,83 a</u> |
| GC CUSUM Tabular | <u>91,74 a</u> | <u>97,40 a</u> | <u>99,79 a</u> |
| RL | <u>85,17 a</u> | <u>93,96 a</u> | <u>97,57 a</u> |
| AD | 69,93 b | <u>94,90 a</u> | <u>94,84 a</u> |

Médias seguidas pela mesma letra, na coluna, não diferem entre si pelo teste de Duncan ($P > 0,05$).

Portanto, os métodos RNA, GC EWMA 1, GC CUSUM e RL se mostraram eficientes e capazes de substituírem o GC de Shewhart, no monitoramento de processos que visam detectar pequenas ou grandes diferenças em relação à média de controle, em termos da incidência do alarme verdadeiro.

Para o conjunto de dados sem a presença da primeira autocorrelação ($\rho_1 = 0$), dois métodos (GC EWMA 1 e GC CUSUM tabular) apresentaram as maiores ($P < 0,05$) médias. Para o conjunto de dados com $\rho_1 = 0,3$ seis métodos apresentaram as maiores ($P < 0,05$) médias. Nesse caso, apenas o GC de Shewhart apresentou a menor ($P < 0,05$) média. O mesmo ocorreu para o conjunto de dados com a presença de autocorrelação $\rho_1 = 0,6$. Para o conjunto de dados com $\rho_1 = 0,9$, cinco métodos (RNA, GC EWMA 1, GC EWMA 2, GC CUSUM e RL) apresentaram as maiores ($P < 0,05$) médias (Tabela 13).

Tabela 13. Médias dos alarmes verdadeiros dos métodos para autocorrelações iguais a 0; 0,3; 0,6 e 0,9

| Método | $\rho = 0$ | $\rho = 0,3$ | $\rho = 0,6$ | $\rho = 0,9$ |
|------------------|------------|--------------|--------------|--------------|
| RNA | 86,48 b | 98,24 a | 97,87 a | 99,78 a |
| GC de Shewhart | 51,11 d | 58,98 b | 59,49 b | 59,54 c |
| GC EWMA 1 | 95,46 a | 97,27 a | 97,27 a | 97,04 a |
| GC EWMA 2 | 86,62 b | 93,70 a | 93,56 a | 91,39 a |
| GC CUSUM Tabular | 97,13 a | 96,25 a | 95,93 a | 95,93 a |
| RL | 69,77 c | 100 a | 100 a | 99,17 a |
| AD | 79,17 b | 91,57 a | 92,82 a | 82,69 b |

Médias seguidas pela mesma letra, na coluna, não diferem entre si pelo teste de Duncan ($P > 0,05$).

Os métodos GC EWMA 1 e GC CUSUM tabular se mostraram mais eficientes com ou sem presença de autocorrelação. Já os métodos RNA, EWMA 2 e RL mostraram mais eficiência apenas com a presença da mesma. Assim, o monitoramento do processo pelos cinco métodos se mostrou capaz de substituir o GC de Shewhart, na presença ou não de autocorrelação, em termos da incidência do alarme verdadeiro.

Para cada método os efeitos de ρ_1 e δ estão apresentados na Tabela 14.

Tabela 14. Estimativas em percentuais dos alarmes verdadeiros

| Método | Equação de regressão | R^2 |
|------------------|---|-------|
| RNA | $P\hat{d} = 72 + 9,01 * \delta + 3,53 * \rho_1 - 1,21 * \rho_1 \times \delta$ | 0,60 |
| GC de Shewhart | $P\hat{d} = -33,3 + 45,3 * \delta$ | 0,90 |
| GC EWMA 1 | $P\hat{d} = 88 + 4,39 * \delta$ | 0,75 |
| GC EWMA 2 | $P\hat{d} = 66,2 + 12,6 * \delta$ | 0,65 |
| GC CUSUM Tabular | $P\hat{d} = 91,1 + 2,92 * \delta - 0,622 * \rho_1 + 0,245 * \rho_1 \times \delta$ | 0,89 |
| RL | $P\hat{d} = 46,5 + 16,2 * \delta + 7,40 * \rho_1 - 2,23 * \rho_1 \times \delta$ | 0,60 |
| AD | $P\hat{d} = 61,6 + 12,5 * \delta$ | 0,35 |

*Significativo pelo teste t ($P < 0,05$), $1 \leq \delta \leq 3$, $0 \leq \rho_1 \leq 0,9$

Os alarmes verdadeiros dos métodos RNA, GC CUSUM tabular e RL dependeram ($P < 0,05$) de ρ_1 e δ e da interação $\rho_1 \times \delta$. Já os demais dependeram ($P < 0,05$) apenas de δ . Nas Figuras 5, 6, 7 e 8, são apresentadas as estimativas dos alarmes verdadeiros em função dos valores de δ para ρ_1 igual a 0; 0,3; 0,6 e 0,9, respectivamente.

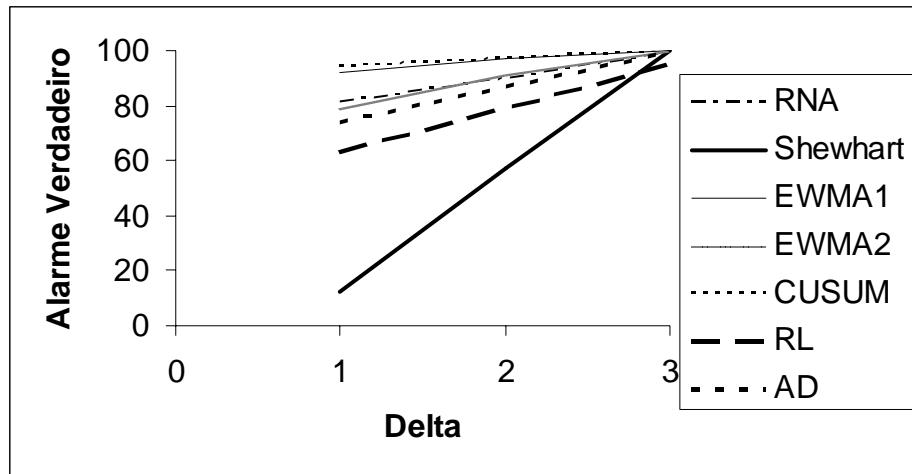


Figura 5. Estimativa do alarme verdadeiro em função de delta sem a presença da primeira autocorrelação

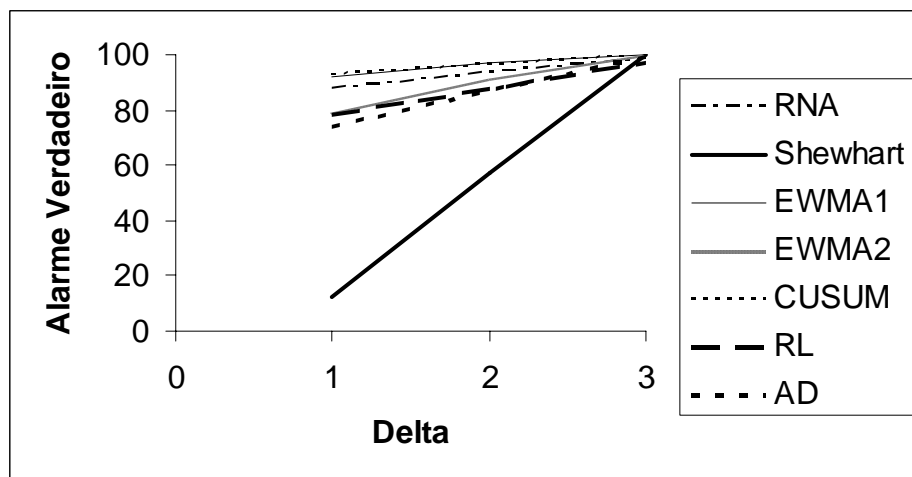


Figura 6. Estimativa do alarme verdadeiro em função de delta com a presença da primeira autocorrelação ($\rho_1 = 0,3$)

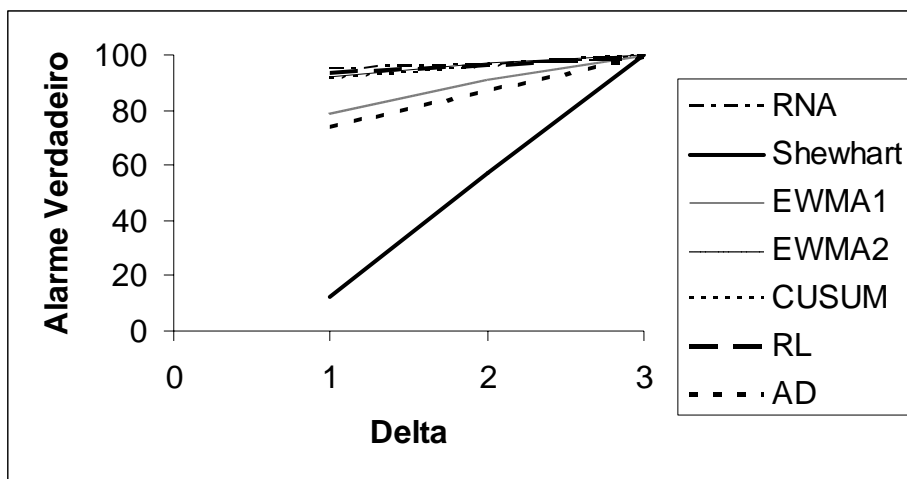


Figura 7. Estimativa do alarme verdadeiro em função de delta com a presença da primeira autocorrelação ($\rho_1 = 0,6$)

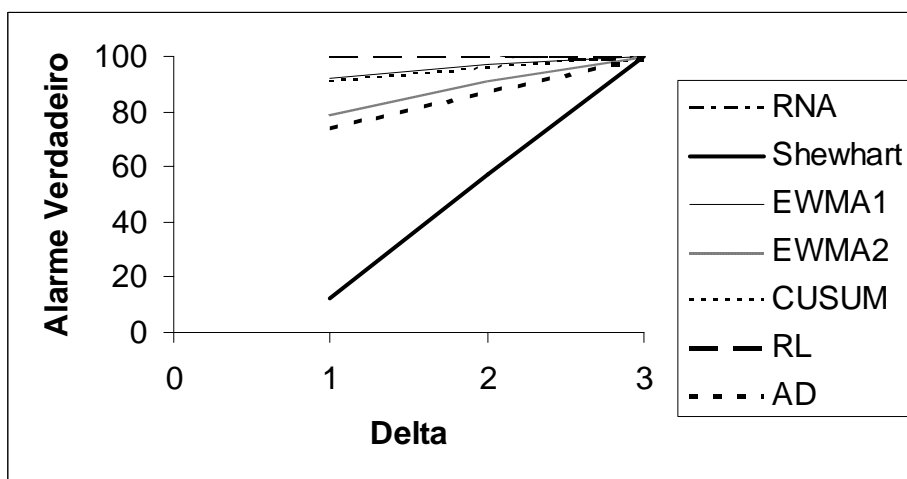


Figura 8. Estimativa do alarme verdadeiro em função de delta com a presença da primeira autocorrelação ($\rho_1 = 0,9$)

Em todos os métodos δ teve efeito ($P < 0,05$) positivo. Assim, quanto maior foi o seu valor maior foi o alarme verdadeiro. Resultados semelhantes foram obtidos por Souza et al. (2008) que conduziram estudos avaliando a eficiência dos gráficos de controle Xbarra, EWMA e CUSUM, sobre a incidência dos alarmes verdadeiros. Além do efeito ($P < 0,05$) de δ , as RNAs foram influenciadas ($P < 0,05$) positivamente por ρ_1 e negativamente pela interação $\rho_1 \times \delta$. O mesmo ocorreu com a RL. O efeito positivo de ρ_1 se destacou na Figura 3 sem a presença de autocorrelação e na Figura 6 com a presença da maior autocorrelação ($\rho_1 =$

0,9). Já o GC CUSUM Tabular foi influenciado ($P < 0,05$) negativamente pela autocorrelação e positivamente pela interação $\rho_1 \times \delta$.

4.3.3 – Alarmes Falso e Verdadeiro

Após determinar os melhores métodos (RNA, GC de Shewhart, GC EWMA 2 e RL) para minimizar a incidência do alarme falso, apenas RNA e RL foram considerados como os melhores em maximizar o alarme verdadeiro, para valores de δ iguais a 1, 2 e 3. Além deles, foram indicados EWMA 2 e GC de Shewhart para valores de δ maiores que 1. A AD foi indicada apenas para $\delta = 3$. Já o GC CUSUM tabular em relação às constantes utilizadas no trabalho, não foi indicado para nenhum valor de δ . Esse método apesar de ótimos resultados em maximizar o alarme verdadeiro promoveu uma grande incidência dos falsos.

Na ausência da primeira autocorrelação ($\rho_1 = 0$), foram indicados os métodos intermediários para aumentar o Pd (RNA, EWMA 2), pelo fato dos melhores (CUSUM tabular e EWMA 1) provocarem alta incidência dos alarmes falsos. Já na presença da primeira autocorrelação (0,3; 0,6 e 0,9), os melhores métodos em minimizar e maximizar os alarmes falso e verdadeiro, respectivamente, foram RNA, EWMA 2 e RL.

Os resultados obtidos coincidem com alguns trabalhos onde foram utilizadas as RNA, RL, AD e os gráficos de controle. Gonçalves (2005) em sua dissertação concluiu que os resultados obtidos pelos modelos de regressão logística e rede neural foram satisfatórios e bastante próximos, sendo o primeiro ligeiramente superior. Já Santos (2008) concluiu que as RNAs foram mais eficientes do que a AD para classificar cerâmicas arqueológicas encontradas em escavações com base na composição química. Balestrassi (2000) destacou que a RNA evita erro em determinar padrão, o que pode ocorrer nos gráficos de controle. Santos (2005) obteve melhor resultado com a RNA em relação à RL na predição da hepatite A. Rodrigues (2006) em sua dissertação de mestrado

comparou o desempenho das RNAs com AD na classificação de padrões no diagnóstico de doenças cardíacas e obteve melhor resultado com as RNAs.

Guimarães e Neto (2006) obtiveram resultados diferentes nas comparações de métodos multivariados com as redes neurais, concluindo que as últimas apresentaram resultados inferiores aos métodos RL e AD sobre as classificações de elementos amostrais baseadas em uma variável dicotômica.

5 – CONCLUSÕES

Os métodos redes neurais artificiais e regressão logística podem substituir, de forma eficiente, os melhores tipos de gráficos de controle, para sinalizarem pontos fora de controle estatístico situados ao meio ou ao final do conjunto de dados, sob diferentes distâncias da média de controle e distribuídos de forma independente ou não. Do mesmo modo, esses métodos têm capacidade de detectarem corretamente pontos que caracterizam o processo como sob controle.

A análise discriminante não se mostrou eficiente em caracterizar corretamente processos univariados sob ou fora de controle estatístico. Apesar do bom desempenho em classificar pontos sob controle, ela gerou muitos alarmes falsos.

O gráfico de controle CUSUM tabular foi muito sensível em detectar alarmes falso e verdadeiro, principalmente quando os pontos fora de controle se situam ao meio do conjunto de dados.

O gráfico de controle de Shewhart é o menos sensível à detecção dos alarmes falso e verdadeiro.

6 – REFERÊNCIAS BIBLIOGRÁFICAS

ALWAN, L.; ROBERTS, H.V. **The Problem of Misplaced Control Limits.** *Applied Statistics*. Milwaukee: Blackwell Publishing, 1995. v. 44

ATIENZA, O.O.; ANG, B.W.; TANG, L.C. Statistical Process Control and Forecasting. *Journal of Quality Science*, v. 2, p. 37-51, 1997.

BALESTRASSI, P.P. **Identificação de padrões em gráficos de Controle Estatístico de Processos, em tempo real, utilizando séries temporais e redes neurais artificiais.** 2000. 170 f. Tese (Doutorado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia da Produção, Universidade Federal de Santa Catarina, Florianópolis, 2000.

CASELLA, G.; BERGER, R.L. **Statistical inference** Belmont: Wadsworth, 2002.

COSTA, A. F.B.; EPPRECHT, E.K.; CARPINETTI, L.C.R. **Controle Estatístico de Qualidade.** São Paulo Atlas, 2004. 334 p.

DANDOLINI, G.A. **Um procedimento para avaliação da saúde financeira de pequenas empresas: estudo de um caso usando redes neurais artificiais.** 1997. Dissertação (Mestrado em Engenharia de Produção), Universidade Federal de Santa Catarina, Florianópolis, 1997.

DEMÉTRIO, C.G.B. **Modelos lineares generalizados em experimentação agrônômica.** ESALQ / USP, Piracicaba, 2002. 113 p.

EWAN, W.D. When & How to Use Cu-Sum Charts. *Technometrics*, v. 5, 1963.

FALTIN, F.W.; MASTRANGELO, C.M.; RUNGER, G.C.; RYAN, T.P. Considerations in the Monitoring of Autocorrelated and Independent Data. **Journal of Quality Technology**, v. 29, p. 131-133, 1997.

FAUSETT, L.V. **Fundamentals of Neural Networks: architecture, algorithms, and applications**. New Jersey: Prentice Hall International, 1994.

FERGUNSON, J. **Introduction to Linear Algebra in Geology**. London: Chapman & Hall, 1994.

FRANÇA, T.V. **Estudo da Predição da Circularidade e Rugosidade de Peças Retificadas Utilizando as Redes Neurais Artificiais**. 2005. Dissertação (Mestrado em Engenharia Industrial), Universidade Estadual Paulista, Bauru, 2005.

FRANCIS, L. The Basics of Neural Networks Demystieild. **Contingencies**, p. 56-61, 2001.

GOEL, A.L.; WU, S.M. Economically Optimum Design of Cusum Charts. **Technometrics**, v. 19, 1973.

GONÇALVES, E.B. **Análise de risco de crédito com o uso de modelos de regressão logística, redes neurais e algoritmos genéticos**. 2005 - Departamento de Administração, Faculdade de Economia, Administração e Contabilidade. São Paulo.

GUIMARÃES, I.A.; NETO, A.C. Reconhecimento de Padrões: Comparação de Métodos Multivariados e Redes Neurais. **Revista Negócios e Tecnologia da Informação**. FGV-EAESP. V 1 2006.

HAIR JUNIOR, J.F.; ANDERSON, R.E.; TATHAN, R.L.; BLAC, W.C. **Análise Multivariada de Dados**. 5 ed. Porto Alegre: Bookman, 2005.

HAYKIN, S. **Redes Neurais, Princípios e Prática**. Porto Alegre: Bookman, 2001.

HAWKINS, D.M.A Cusum for a Scale Parameter. **Journal of Quality Science Technology**, v. 13, 1981.

HOSMER, D.W.; LEMESHOW, S. **Applied Logistic Regression**. 2. ed. USA: John Wiley & Sons, 1989.

JURAN, J.M. **Juran's quality handbook**. 5. ed. New York: McGraw-Hill, 1999.

KING, D.W. **Statistical quality control using the SAS system**. North Carolina: SAS Institut Inc, 1995. 383 p.

LACHENBRUCH, P.A. **Discriminant analysis**. New York: Hafner Press, 1975.

MARQUARDT, D.W. **New Technical and Educational Directions for Managing Product Quality**. USA: The American Statistician, 1984.

MELLO, M.T. **Aplicação de Redes Neurais Artificiais no Processo de Precificação de Ações**. 2004. Monografia (Ciências da Computação), Universidade Federal de Pelotas, Pelotas.

MESSINA, W. **Statistical Quality Control for Manufacturing Manages**. New York: Wiley Series in Engineering Management, 1987.

MINGOTI, S.A. **Análise de dados através de métodos de estatística multivariada: Uma abordagem aplicada**. Belo Horizonte: Editora UFMG, 2005.

MONTGOMERY, D.C. **Introdução ao Controle Estatístico da Qualidade**. 4^a. ed. Rio de Janeiro: LTC, 2004.

MONTGOMERY, D.C.; RUNGER, G.C. **Estatística Aplicada e Probabilidade para Engenheiros**. Rio de Janeiro: LTC, 2003.

PAGE, E.S. Continuous Inspection Schemes. **Biometrika**, v. 41, 1954.

REYNOLDS, M.R.J. Approximations to the Average Run Length in Cumulative Sum Control Charts. **Technometrics**, v. 17, 1975.

RIBEIRO JÚNIOR, J.I.; FARIA, R.O.; TERRA, N.S. **Ferramentas estatísticas básicas da qualidade - guia prático SAS**. Viçosa: Editora UFV, 2006. 157 p.

RODRIGUES, T.B. **Seleção de Variáveis e Classificação de Padrões por Redes Neurais como Auxílio ao Diagnóstico de Doença Cardíaca**. 2006. 73 f. Dissertação (Mestre em Engenharia Elétrica), PUC-RJ, Rio de Janeiro.

SANTOS, A.M. Usando Redes Neurais Artificiais e Regressão Logística na Predição da Hepatite A. **Revista Bras Epidemiol**, v. 8, p. 117, 2005.

SANTOS, J.O. Aplicação da análise discriminante linear e redes neurais artificiais para classificação de cerâmicas arqueológicas com base na composição química. **III Congresso de Pesquisa e Inovação da Rede Norte Nordeste de Educação Tecnológica** 2008.

SOUZA, L.M.; RIBEIRO JÚNIOR, J.I.; REIS, G.M.; IDE, M.S. Eficiência dos gráficos de controle Xbarra, EWMA e CUSUM. **Revista Eletrônica de Engenharia de Produção**, v. 1, 2008.

TAFNER, M.A. O Que são Redes Neurais Artificiais: <http://www.cerebromente.org.br/n05/tecnologia/rna.htm>. 16 junho 2008.

WOODALL, W.H. The Statistical Design of Quality Control Charts. **The Statistician**, v. 34, 1985.

WHEELER, D.J. **Advanced Topics in Statistical Process Control: the power of Shewhart's Charts**. Knoxville: SPC 1995.

ANEXO I

Função de simulação dos dados independentes com pontos fora de controle no meio do conjunto de dados e treinamento e validação das RNAs no software MatLab.

```
clear, close all, clc;

%CRIANDO VARIAVEIS
resultado1 = []; % matriz para ir armazenando o resultado de cada
interação

for j = 1:3
    %dados de treinamento

    media = 1000 + j*100;
    y = [random('normal',1000,100,1,144)
random('normal',media,100,1,144) random('normal',1000,100,1,144)];
    y1 = y(5:432);
    y2 = y(4:431);
    y3 = y(3:430);
    y4 = y(2:429);
    y5 = y(1:428);
    p = [y1; y2; y3; y4; y5];
    t = [zeros(1,140) ones(1,144) zeros(1,144)];

    % normalizacao dos dados de treinamento
    [tn, mint, maxt] = premnmx(t);
    [pn, minp, maxp] = premnmx(p);

    x_train = tn;
    y_train = pn;

% CRIANDO A RNA

    net = newff([minmax(y_train)], [6 4 3 1], {'tansig' 'tansig'
'tansig' 'tansig'}, 'traingdx');

    net.trainParam.epochs = 1250;
    net.trainParam.goal = 0.01;
    net.trainParam.show = 10;

% Treinamento da RNA
    net01 = train(net, y_train, x_train);

    for rep = 1:10
        %dados de validação
        v = [random('normal',1000,100,1,36)
random('normal',media,100,1,36) random('normal',1000,100,1,36)];
        vd = [v(5:108); v(4:107); v(3:106); v(2:105); v(1:104)];
```

```

f = [ zeros(1,32) ones(1,36) zeros(1,36) ]; % variavel para obter
o erro em relação a resposta do conjunto de validação.
% normalizacao dos dados validação
[vn,minv,maxv] = premmx(vd);

% Simulacao da RNA
r = hardlim(sim(net01, vn));
%calculo do erro alfa
fal = [f(1:32) f(69:104)]; %relacionado ao erro alfa
ral = [r(1:32) r(69:104)];
al = 68 - sum(ral==fal); %alfa = 68(numero de pontos sob-
controle) - (o numero de vezes que a rna acertou o resultado(apontou
sob controle quando estava sob controle)
%calculo do beta
fbe = f(33:68); %relacionado ao erro beta
rbe = r(33:68);
be = 36 - sum(rbe==fbe); %beta = 36(numero de pontos fora-
controle) - (o numero de vezes que a rna acertou o resultado(apontou
fora controle quando estava fora controle)

resultado1 = [resultado1; rep j al be]; %[ resultado1 , repeticao
j de 1 a 3 , numero de erros alfa, n de erros beta]
xlswrite(strcat(num2str(j),'dadoA',num2str(rep)),v); %resultado do
valor de v validacao Arquivo vai ter nome j(valor delta) de 1 a 3,
dado A refere a 1 repeticao,n de rep de 1 a 10

end

end
%repteção 2
%CRIANDO VARIAVEIS
resultado2 = []; % matriz para ir armazenando o resultado de cada
interação

for j = 1:3
%dados de treinamento

media = 1000 + j*100;
y = [random('normal',1000,100,1,144)
random('normal',media,100,1,144) random('normal',1000,100,1,144)];
y1 = y(5:432);
y2 = y(4:431);
y3 = y(3:430);
y4 = y(2:429);
y5 = y(1:428);
p = [y1; y2; y3; y4; y5];
t = [zeros(1,140) ones(1,144) zeros(1,144)];

% normalizacao dos dados de treinamento
[tn, mint, maxt] = premmx(t);
[pn, minp, maxp] = premmx(p);

x_train = tn;
y_train = pn;

% CRIANDO A RNA

net = newff([minmax(y_train)], [6 4 3 1], {'tansig' 'tansig'
'tansig' 'tansig'}, 'traingdx');

```



```

net.trainParam.epochs = 1250;
net.trainParam.goal = 0.01;
net.trainParam.show = 10;

% Treinamento da RNA
net01 = train(net, y_train, x_train);

for rep = 1:10
%dados de validação
v = [random('normal',1000,100,1,36)
random('normal',media,100,1,36) random('normal',1000,100,1,36)];
vd = [v(5:108); v(4:107); v(3:106); v(2:105); v(1:104)];
f = [ zeros(1,32) ones(1,36) zeros(1,36) ]; % variavel para obter
o erro em relação a resposta do conjunto de validação.
% normalizacao dos dados validação
[vn,minv,maxv] = premmmx(vd);

% Simulacao da RNA
r = hardlim(sim(net01, vn));
%calculo do erro alfa
fal = [f(1:32) f(69:104)];
ral = [r(1:32) r(69:104)];
al = 68 - sum(ral==fal);
%calculo do beta
fbe = f(33:68);
rbe = r(33:68);
be = 36 - sum(rbe==fbe);

resultado2 = [resultado2; rep j al be];
xlswrite(strcat(num2str(j), 'dadoB', num2str(rep)), v);

end

end

%repteção 3
%CRIANDO VARIAVEIS
resultado3 = []; % matriz para ir armazenando o resultado de cada
interação

for j = 1:3
%dados de treinamento

media = 1000 + j*100;
y = [random('normal',1000,100,1,144)
random('normal',media,100,1,144) random('normal',1000,100,1,144)];
y1 = y(5:432);
y2 = y(4:431);
y3 = y(3:430);
y4 = y(2:429);
y5 = y(1:428);
p = [y1; y2; y3; y4; y5];
t = [zeros(1,140) ones(1,144) zeros(1,144)];

% normalizacao dos dados de treinamento
[tn, mint, maxt] = premmmx(t);
[pn, minp, maxp] = premmmx(p);

```

```

x_train = tn;
y_train = pn;

% CRIANDO A RNA

net = newff([minmax(y_train)], [6 4 3 1], {'tansig' 'tansig'
'tansig' 'tansig'}, 'traingdx');

net.trainParam.epochs = 1250;
net.trainParam.goal = 0.01;
net.trainParam.show = 10;

% Treinamento da RNA
net01 = train(net, y_train, x_train);

for rep = 1:10
%dados de validação
v = [random('normal',1000,100,1,36)
random('normal',media,100,1,36) random('normal',1000,100,1,36)];
vd = [v(5:108); v(4:107); v(3:106); v(2:105); v(1:104)];
f = [ zeros(1,32) ones(1,36) zeros(1,36) ]; % variavel para obter
o erro em relação a resposta do conjunto de validação.
% normalizacao dos dados validação
[vn,minv,maxv] = premmx(vd);

% Simulacao da RNA
r = hardlim(sim(net01, vn));
%calculo do erro alfa
fal = [f(1:32) f(69:104)];
ral = [r(1:32) r(69:104)];
al = 68 - sum(ral==fal);
%calculo do beta
fbe = f(33:68);
rbe = r(33:68);
be = 36 - sum(rbe==fbe);

resultado3 = [resultado3; rep j al be];
xlswrite(strcat(num2str(j), 'dadoC', num2str(rep)), v);

end

end

```

ANEXO II

Função de simulação dos dados independentes com pontos fora de controle no final do conjunto de dados e treinamento e validação das RNAs no software MatLab.

```
clear, close all, clc;

%CRIANDO VARIAVEIS
resultado1 = []; % matriz para ir armazenando o resultado de cada
interação

for j = 1:3
    %dados de treinamento

    media = 1000 + j*100;
    y = [random('normal',1000,100,1,144)
random('normal',1000,100,1,144) random('normal',media,100,1,144)];
    y1 = y(5:432);
    y2 = y(4:431);
    y3 = y(3:430);
    y4 = y(2:429);
    y5 = y(1:428);
    p = [y1; y2; y3; y4; y5];
    t = [zeros(1,140) zeros(1,144) ones(1,144)];

    % normalizacao dos dados de treinamento
    [tn, mint, maxt] = premnmx(t);
    [pn, minp, maxp] = premnmx(p);

    x_train = tn;
    y_train = pn;

% CRIANDO A RNA

    net = newff([minmax(y_train)], [6 4 3 1], {'tansig' 'tansig'
'tansig' 'tansig'}, 'traingdx');

    net.trainParam.epochs = 1250;
    net.trainParam.goal = 0.01;
    net.trainParam.show = 10;

% Treinamento da RNA
    net01 = train(net, y_train, x_train);

    for rep = 1:10
        %dados de validação
        v = [random('normal',1000,100,1,36) random('normal',1000,100,1,36)
random('normal',media,100,1,36)];
        vd = [v(5:108); v(4:107); v(3:106); v(2:105); v(1:104)];
        f = [ zeros(1,32) zeros(1,36) ones(1,36) ]; % variavel para obter
o erro em relação a resposta do conjunto de validação.
        % normalizacao dos dados validação
```

```

[vn,minv,maxv] = premmx(vd);

% Simulacao da RNA
r = hardlim(sim(net01, vn));
%calculo do erro alfa
fal = [f(1:32) f(33:68)]; %relacionado ao erro alfa
ral = [r(1:32) r(33:68)];
al = 68 - sum(ral==fal); %alfa = 68(numero de pontos sob-
controle) - (o numero de vezes que a rna acertou o resultado(apontou
sob controle quando estava sob controle)
%calculo do beta
fbe = f(69:104); %relacionado ao erro beta
rbe = r(69:104);
be = 36 - sum(rbe==fbe); %beta = 36(numero de pontos fora-
controle) - (o numero de vezes que a rna acertou o resultado(apontou
fora controle quando estava fora controle)

resultadol = [resultadol; rep j al be]; %[ resultadol , repeticao
j de 1 a 3 , numero de erros alfa, n de erros beta]
xlswrite(strcat(num2str(j),'dadoA',num2str(rep)),v); %resultado do
valor de v validacao Arquivo vai ter nome j(valor delta) de 1 a 3,
dado A refere a 1 repeticao,n de rep de 1 a 10

end

end

%repteção 2
%CRIANDO VARIAVEIS
resultado2 = []; % matriz para ir armazenando o resultado de cada
interação

for j = 1:3
%dados de treinamento

media = 1000 + j*100;
y = [random('normal',1000,100,1,144)
random('normal',1000,100,1,144) random('normal',media,100,1,144)];
y1 = y(5:432);
y2 = y(4:431);
y3 = y(3:430);
y4 = y(2:429);
y5 = y(1:428);
p = [y1; y2; y3; y4; y5];
t = [zeros(1,140) zeros(1,144) ones(1,144)];

% normalizacao dos dados de treinamento
[tn,mint,maxt] = premmx(t);
[pn,minp,maxp] = premmx(p);

x_train = tn;
y_train = pn;

% CRIANDO A RNA

net = newff([minmax(y_train)], [6 4 3 1],{'tansig' 'tansig'
'tansig' 'tansig'}, 'traingdx');

```

```

        net.trainParam.epochs = 1250;
        net.trainParam.goal = 0.01;
        net.trainParam.show = 10;

% Treinamento da RNA
        net01 = train(net, y_train, x_train);

for rep = 1:10
    %dados de validação
    v = [random('normal',1000,100,1,36) random('normal',1000,100,1,36)
random('normal',media,100,1,36)];
    vd = [v(5:108); v(4:107); v(3:106); v(2:105); v(1:104)];
    f = [ zeros(1,32) zeros(1,36) ones(1,36) ]; % variavel para obter
o erro em relação a resposta do conjunto de validação.
    % normalizacao dos dados validação
    [vn,minv,maxv] = premmmx(vd);

    % Simulacao da RNA
    r = hardlim(sim(net01, vn));
    %calculo do erro alfa
    fal = [f(1:32) f(33:68)];
    ral = [r(1:32) r(33:68)];
    al = 68 - sum(ral==fal);
    %calculo do beta
    fbe = f(69:104);
    rbe = r(69:104);
    be = 36 - sum(rbe==fbe);

    resultado2 = [resultado2; rep j al be];
    xlswrite(strcat(num2str(j), 'dadoB', num2str(rep)), v);

end

end

%repteção 3
%CRIANDO VARIAVEIS
resultado3 = []; % matriz para ir armazenando o resultado de cada
interação

for j = 1:3
    %dados de treinamento

    media = 1000 + j*100;
    y = [random('normal',1000,100,1,144)
random('normal',1000,100,1,144) random('normal',media,100,1,144)];
    y1 = y(5:432);
    y2 = y(4:431);
    y3 = y(3:430);
    y4 = y(2:429);
    y5 = y(1:428);
    p = [y1; y2; y3; y4; y5];
    t = [zeros(1,140) zeros(1,144) ones(1,144)];

    % normalizacao dos dados de treinamento
    [tn, mint, maxt] = premmmx(t);
    [pn, minp, maxp] = premmmx(p);

```

```

x_train = tn;
y_train = pn;

% CRIANDO A RNA

net = newff([minmax(y_train)], [6 4 3 1], {'tansig' 'tansig'
'tansig' 'tansig'}, 'traingdx');

net.trainParam.epochs = 1250;
net.trainParam.goal = 0.01;
net.trainParam.show = 10;

% Treinamento da RNA
net01 = train(net, y_train, x_train);

for rep = 1:10
%dados de validação
v = [random('normal',1000,100,1,36) random('normal',1000,100,1,36)
random('normal',media,100,1,36)];
vd = [v(5:108); v(4:107); v(3:106); v(2:105); v(1:104)];
f = [ zeros(1,32) zeros(1,36) ones(1,36) ]; % variavel para obter
o erro em relação a resposta do conjunto de validação.
% normalizacao dos dados validação
[vn,minv,maxv] = premmx(vd);

% Simulacao da RNA
r = hardlim(sim(net01, vn));
%calculo do erro alfa
fal = [f(1:32) f(33:68)];
ral = [r(1:32) r(33:68)];
al = 68 - sum(ral==fal);
%calculo do beta
fbe = f(69:104);
rbe = r(69:104);
be = 36 - sum(rbe==fbe);

resultado3 = [resultado3; rep j al be];
xlswrite(strcat(num2str(j), 'dadoC', num2str(rep)), v);

end

end

```

ANEXO III

Função de simulação dos dados autocorrelacionados no software R.

```
gera.cor<-function()
{
diretorio.anterior<-getwd()
setwd("D:/1 - Thiago/tese/autocorrelacionados")
diretorio<-getwd()
lixo<-readline(paste("Gravando dados em ",diretorio,"... <ENTER>",sep=""))
rm(lixo,diretorio)
quantos<-as.numeric(readline("Quantos arquivos gerar? Resposta:"))
for (i in 1:quantos)
{
a=matrix(1000 + arima.sim(n = 72, list(ar = c(0.3)),sd = sqrt(100)),72,1)
b=matrix(1100 + arima.sim(n = 36, list(ar = c(0.3)),sd = sqrt(100)),36,1)
y=(rbind(a,b))
nomearquivo<-paste("1_dado_a_",i,".txt",sep="")
write.table(y,nomearquivo,row.names=F,quote=F,dec=".",col.names=F)
rm(a,b,y)
}
cat("Arquivos gerados com sucesso!\n")
setwd(diretorio.anterior)
}
```

ANEXO IV

Função de treinamento e validação das RNAs no software MatLab.

```
clear, close all, clc;

%CRIANDO VARIAVEIS
resultado1 = []; % matriz para ir armazenando o resultado de cada
interação
u= dlmread('D:\1 -
Thiago\tese\autocorrelacionados\final\t3_dado_a_10.txt');
y = u';
    y1 = y(5:432);
    y2 = y(4:431);
    y3 = y(3:430);
    y4 = y(2:429);
    y5 = y(1:428);
    p = [y1; y2; y3; y4; y5];
    t = [zeros(1,140) zeros(1,144) ones(1,144)];

% normalizacao dos dados de treinamento
    [tn, mint, maxt] = premmmx(t);
    [pn, minp, maxp] = premmmx(p);

    x_train = tn;
    y_train = pn;

% CRIANDO A RNA

    net = newff([minmax(y_train)], [6 4 3 1], {'tansig' 'tansig'
'tansig' 'tansig'}, 'traingdx');

    net.trainParam.epochs = 1250;
    net.trainParam.goal = 0.01;
    net.trainParam.show = 10;

% Treinamento da RNA
    net01 = train(net, y_train, x_train);

%dados de validação
    t= dlmread('D:\1 -
Thiago\tese\autocorrelacionados\final\v3_dado_a_10.txt');
    v = t';
    vd = [v(5:108); v(4:107); v(3:106); v(2:105); v(1:104)];
    f = [ zeros(1,32) zeros(1,36) ones(1,36) ]; % variavel para obter
o erro em relação a resposta do conjunto de validação.
    % normalizacao dos dados validação
    [vn, minv, maxv] = premmmx(vd);

    % Simulacao da RNA
    r = hardlim(sim(net01, vn));
    %calcula do erro alfa
```



```

    fal = [f(1:68) ]; %relacionado ao erro alfa
    ral = [r(1:68) ];
    al = 68 - sum(ral==fal); %alfa = 68(numero de pontos sob-
controle) - (o numero de vezes que a rna acertou o resultado(apontou
sob controle quando estava sob controle)
    %calculo do beta
    fbe = f(69:104); %relacionado ao erro beta
    rbe = r(69:104);
    be = 36 - sum(rbe==fbe); %beta = 36(numero de pontos fora-
controle) - (o numero de vezes que a rna acertou o resultado(apontou
fora controle quando estava fora controle)

    resultado1 = [resultado1; al; be]; %[ resultado1 , numero de
erros alfa, n de erros beta]
    xlswrite(strcat('3dadoA10'),v); %delta, dadoA, repeticao

```