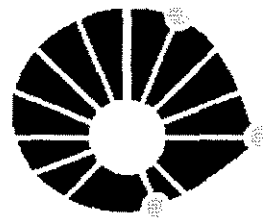


UNICAMP

UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA QUÍMICA

ÁREA DE CONCENTRAÇÃO: ENGENHARIA DE PROCESSOS

APLICAÇÃO DE REDES NEURAIS ARTIFICIAIS E DE QUIMIOMETRIA NA MODELAGEM DO PROCESSO DE CRAQUEAMENTO CATALÍTICO FLUIDO

Autor : Wagner Roberto de Oliveira Pimentel
Orientador : Prof. Dr. Antonio Carlos Luz Lisboa

Tese de Doutorado apresentada à Faculdade de Engenharia Química como parte dos requisitos exigidos para a obtenção do título de Doutor em Engenharia Química.

Campinas - São Paulo

Março de 2005

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

P649a

Pimentel, Wagner Roberto de Oliveira
Aplicação de redes neurais artificiais e de quimiometria na modelagem do processo de craqueamento catalítico fluido / Wagner Roberto de Oliveira Pimentel.-- Campinas, SP: [s.n.], 2005.

Orientador: Antonio Carlos Luz Lisboa.
Tese (Doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Química.

1. Redes Neurais (Computação). 2. Craqueamento catalítico. 3. Modelagem de dados. 4. Otimização matemática. Simulação (Computadores). I. Lisboa, Antonio Carlos Luz. II. Universidade Estadual de Campinas. Faculdade de Engenharia Química. III. Título.

Titulo em Inglês: Application of artificial neural networks and chemometrics in the modeling of fluid catalytic cracking process.
Palavras-chave em Inglês: Artificial neural network, Catalytic cracking, Data modelling, Mathematical optimization e Computer simulation
Área de concentração: Engenharia de Processos
Titulação: Doutor em Engenharia Química
Banca examinadora: Waldir Martignoni, Ronei Jesus Poppi, Liliane Maria Ferrareso Lona e Rubens Maciel Filho
Data da defesa: 18/03/2005

Tese de Doutorado defendida por Wagner Roberto de Oliveira Pimentel e aprovada em 18 de março de 2005 pela banca examinadora constituída pelos doutores:

Prof. Dr. Antonio Carlos Luz Lisbôa

FEQ – UNICAMP

Dr. Waldir Martignoni

CENPES – PETROBRAS

Prof. Dr. Ronei Jesus Poppi

IQ – UNICAMP

Prof. Dra. Liliane Maria Ferrareso Lona

FEQ – UNICAMP

Prof. Dr. Rubens Maciel Filho

FEQ – UNICAMP

Este exemplar corresponde à versão final da Tese de Doutorado em Engenharia Química, defendida pelo Msc. Eng. Wagner Roberto de Oliveira Pimentel e aprovada pela banca examinadora em 18 de Março de 2005.

Prof. Dr. Antonio Carlos Luz Lisboa

AGRADECIMENTOS

A minha esposa Luciana e ao meu filho Lucas pelo apoio, amor, carinho e paciência nessa jornada.

A meus pais, Doralice e João, meus irmãos, Welber, Wangles e Welma e a todos das famílias Félix de Oliveira e Pimentel.

Ao meu orientador Prof. Dr. Antonio Carlos Luz Lisbôa, por tudo o que aprendi nesses anos de pós-graduação e por ter fornecido um ambiente propício para a pesquisa de excelência.

Aos professores Ana Frattini Fileti, Fernando José Von Zuben e Ronei Jesus Poppi, pelas valiosas contribuições dadas nas áreas abordadas neste trabalho.

Aos amigos do LDPSP: Andre, Kiki, Manoel, Mardonny e Paulo Porto. Pelo companheirismo, pelas conversas e pelos momentos de "paciência" que tivemos no laboratório.

Aos amigos Urso, Francisleo, Alvaro, Baco, Daniel, Luis, Protásio, Andre Garcia, Junior, Cadu, Agivaldo, Osvaldo e a comunidade Alagoana na FEQ: Edler, Sergio, Karla, Wanda, Ronaldo, Emerson, Elaine, Luciana e Felipe. E a todos aqueles que fizeram parte desta minha jornada na pós-graduação.

A PETROBRAS, pelos dados da planta piloto e da unidade industrial de FCC. Em especial aos engenheiros: Waldir Martignoni (planta piloto), Geraldo Marcio Diniz Santos (RLAM), Daniel Ruy Ribeiro Moreira (RLAM) e Jorge Antonio Santos Freitas (RLAM).

A FAPESP, pelo fundamental suporte financeiro dispensado à execução deste trabalho de pesquisa.

RESUMO

O craqueamento catalítico fluido (FCC) é um dos mais importantes processos de refino da atualidade que produz, dentre outros produtos, gasolina e GLP. Trata-se de um processo que apresenta grande dificuldade de ser modelado fenomenologicamente. Dentro desse contexto surgem as redes neurais artificiais (RNA) como ferramenta de modelagem, visto que as RNA são capazes de "aprender" o que ocorre no processo por meio de um conjunto limitado de dados e apresentam um menor tempo de processamento se comparado aos modelos fenomenológicos. O objetivo principal deste trabalho é desenvolver modelos empíricos, baseados em RNA e na quimiometria, capazes de relacionar as variáveis de entrada com as variáveis de saída do processo de craqueamento catalítico fluido (planta piloto e unidade industrial). Os dados experimentais foram obtidos na unidade piloto de FCC da Petrobrás localizada na usina de xisto em São Mateus do Sul - PR e os dados industriais foram obtidos da unidade da RLAM localizada em São Francisco do Conde – BA. Para uma boa performance das redes foi utilizada a técnica de análise dos componentes principais (PCA) para um pré-processamento dos dados e em seguida foram usadas redes MLP com os seguintes algoritmos de treinamento supervisionado: Método de Broyden-Fletcher-Goldfarb-Shanno (BFGS), Método do Gradiente Conjugado Escalonado (SCG) e Levenberg-Marquardt (LM). Também foram estudados métodos para aumentar a capacidade de predição da rede (generalização). Bons resultados para a planta piloto e para unidade industrial foram obtidos com o método LM utilizando a regularização Bayesiana e com o modelo que utiliza os escores obtidos da PCA como entrada da rede (também com o método LM utilizando a regularização). Os modelos empíricos obtidos podem ser usados para otimização de riser de FCC visto que os modelos apresentam uma elevada capacidade de predição.

Palavras-chave: Redes Neurais Artificiais, Craqueamento Catalítico Fluido, Quimiometria, Modelagem, Otimização, Simulação.

ABSTRACT

The fluidized bed catalytic cracking process is one of the most important refining processes. It produces, among other distillates, gasoline and liquefied petroleum gas (LPG). It is very difficult to model it by fundamental balances. On the other hand, artificial neural networks (ANN) offer convenient tools to describe complex processes. They are able to learn what is going on with in the process through a limited amount of information, requiring less computing time than phenomenological modeling. The main objective of this work was to develop empirical models - based on ANNs and chemometrics - able to relate input and output variables of the FCC process, using data from a pilot and from an industrial plant. Experimental data were obtained from the Petrobras FCC pilot plant located in São Mateus do Sul, Paraná, and from the Petrobras Landulpho Alves Refinery FCC industrial plant located in São Francisco do Conde, Bahia. The principal component analysis (PCA) technique was initially used to preprocess the data. Artificial neural networks were then employed with the following supervising training algorithms: Broyden-Fletcher-Godfarb-Shanno (BFGS), Scaled Conjugated Gradient (SCG) and Levenberg-Marquardt (LM). Methods devised to increase the artificial network prediction power were also used. Good results were obtained for the pilot and industrial plant with the LM algorithm, using Bayesian regularization, with the model that uses the scores produced by the PCA as inputs (also with LM using regularization). The developed empirical models may be used to optimize FCC risers, for they showed excellent prediction capacity.

Keywords: Artificial Neural Networks, FCC, Chemometrics, Modeling, Optimization, Simulation.

SUMÁRIO

- Agradecimentos iii

- Abstract iv

- Resumo v

- Sumário vi

- Lista de Figuras x

- Lista de Tabelas xvi

- Nomenclatura xix

- Capítulo 1: INTRODUÇÃO 01

    1.1 Introdução. 01

    1.2 Objetivos do trabalho. 04

    1.3 Organização do texto 05

- Capítulo 2: DESCRIÇÃO DO PROCESSO DE CRAQUEAMENTO CATALÍTICO FLUIDO 07

    2.1 Introdução. 07

    2.2 Descrição do Processo 08

    2.3 Os dados da unidade de FCC da SIX – Petrobras 13

    2.4 Os dados da unidade industrial de FCC da RLAM – Petrobras 16

    2.5 Variáveis de entrada 17

    2.6 Variáveis de saída 18

        2.6.1 Conversão 18

        2.6.2 Rendimentos dos Produtos 19

    2.7 Considerações finais sobre o capítulo 20

- Capítulo 3: REDES NEURAIS ARTIFICIAIS 21

3.1 Introdução. 21

3.2 Arquitetura de Rede. 23

3.3 Processo de Aprendizagem. 24

3.4 Modelos de Relacionamento da Rede Neural com o seu Ambiente. 25

3.5 O *Perceptron* 26

3.6 O Perceptron de Múltiplas Camadas (MLP) 28

3.6.1 Função de Ativação 29

3.6.2 O Algoritmo de Retropropagação (*backpropagation*) 30

3.6.3 Algoritmos de otimização para treinamento supervisionado 38

3.6.3.1 Método de Levenberg-Marquardt 39

3.6.3.2 Método de Broyden-Fletcher-Goldfarb-Shanno 41

3.6.3.3 Método do Gradiente conjugado escalonado 43

3.7 Definindo a rede MLP para o estudo da modelagem do FCC 48

3.7.1 Modo de Treinamento 48

3.7.2 Número de Camadas e de neurônios (topologia da rede) 49

3.7.3 Amostras: divisão para treinamento, validação e teste 50

3.7.4 Generalização 51

3.7.4.1 Parada Antecipada 51

3.7.4.2 Regularização Bayesiana 52

3.8 Considerações finais sobre o capítulo 56

- Capítulo 4: QUIMIOMETRIA 57

4.1 Introdução. 57

4.2 Pré-processamentos 58

4.2.1 Dados centrados na média 58

4.2.2 Escalamento pela variância 59

4.2.3 Autoescalamento 59

4.3 Análise dos Componentes Principais 60

4.3.1 Introdução 60

4.3.2 O método PCA 62

4.3.3 Estatística T^2 de Hotelling para PCA 65

4.3.4 Regressão dos Componentes Principais 66

4.4 Mínimos Quadrados Parciais (PLS) 68

4.4.1 Introdução 68

4.4.2 O método PLS 69

4.4.3 Validação Cruzada 71

4.5 PCR e PLS com relações internas não lineares 72

4.6 Considerações finais sobre o capítulo. 72

- Capítulo 5: RESULTADOS E DISCUSSÕES 73

5.1 Modelagem Global – Dados da SIX 73

5.1.1 Análise Exploratória dos Dados 73

5.1.2 Modelos de Regressão Linear (PCR e PLS) 82

5.1.2.1 PCR 82

5.1.2.1.1 Conjunto 1 82

5.1.2.1.2 Conjunto 2 84

5.1.2.1.3 Discussão sobre a PCR 85

5.1.2.2 PLS 88

5.1.2.2.1 Conjunto 1 88

5.1.2.2.2 Conjunto 2 90

5.1.2.2.3 Discussão sobre a PLS 91

5.1.3 Redes Neurais com Parada Antecipada 94

5.1.3.1 Conjunto 1 94

5.1.3.2 Conjunto 2 98

5.1.3.3 Discussão sobre redes neurais artificiais 101

5.1.4 Redes Neurais com Regularização 102

5.1.5 PLS com Redes Neurais 108

5.2 Modelagem Individual – Dados da SIX 109

5.2.1 Redes Neurais com Regularização 110

5.2.2 PLS com Redes Neurais 110

5.2.3 PCR com Redes Neurais 111

5.2.4 Discussão sobre os modelos individuais 113

5.3 Modelagem Global – Dados da RALM 116

5.3.1 Análise Exploratória dos Dados 116

5.3.2 Resultados da Modelagem Global 123

5.4 Modelagem Individual – Dados da RLAM 131

- Capítulo 6: CONCLUSÕES 137

6.1- Conclusões Gerais 137

6.2- Conclusões sobre os resultados obtidos para SIX 138

6.3- Conclusões sobre os resultados obtidos para RLAM 139

- SUGESTÕES PARA TRABALHOS FUTUROS 141

- REFERÊNCIAS BIBLIOGRÁFICAS 143

- APÊNDICE A – Resultados da modelagem individual para os dados da RLAM 155

LISTA DE FIGURAS

Figura 2.1: Conversor UOP 9

Figura 2.2: esquema da unidade multipropósito (UMP) de FCC da SIX – Petrobras 14

Figura 2.3: Diagrama do circuito regenerador – *riser– stripper*, da unidade multipropósito de FCC – U-144 da SIX/PETROBRÁS 15

Figura 3.1: Esquema básico de um neurônio biológico e de um artificial 22

Figura 3.2: Exemplos de arquiteturas de redes neurais artificiais 24

Figura 3.3: Unidade de saída 27

Figura 3.4: Arquitetura *perceptron* de múltiplas camadas com uma camada oculta 29

Figura 3.5: Rede neural com 2 camadas. Notação abreviada 32

Figura 4.1: Representação gráfica da análise dos componentes principais 64

Figura 5.1: Gráfico das componentes principais em função da variância acumulada para os dados da SIX com 84 amostras 74

Figura 5.2: Gráfico dos pesos em PC1, PC2, PC3 e PC4 76

Figura 5.3: Gráfico dos escores em PC1, PC2, PC3, PC4 e PC5 77

Figura 5.4: Gráfico de T^2 em função das 84 amostras 78

Figura 5.5: Gráfico das componentes principais em função da variância acumulada para os dados da SIX com 82 amostras 79

Figura 5.6: Gráficos dos pesos em PC1, PC2 e PC3 80

Figura 5.7: Gráficos dos escores em PC1, PC2 e PC3 81

Figura 5.8: Gráfico de T^2 em função das 82 amostras 82

Figura 5.9: Gráfico de *RMSE* (treinamento) em função das variáveis para diferentes números de componentes principais – PCR para o conjunto 1 83

Figura 5.10: Gráfico de *RMSE* (validação) em função das variáveis para diferentes números de componentes principais – PCR para o conjunto 1 83

Figura 5.11: Gráfico de *RMSE* (treinamento) em função das variáveis para diferentes números de componentes principais – PCR para o conjunto 2 84

Figura 5.12: Gráfico de *RMSE* (validação) em função das variáveis para diferentes números de componentes principais – PCR para o conjunto 2 85

Figura 5.13: Gráfico de *RMSE* (treinamento) em função das variáveis para diferentes números de componentes principais – PLS para o conjunto 1 89

Figura 5.14: Gráfico de *RMSE* (validação) em função das variáveis para diferentes números de componentes principais – PLS para o conjunto 1 90

Figura 5.15: Gráfico de *RMSE* (treinamento) em função das variáveis para diferentes números de componentes principais – PLS para o conjunto 2 92

Figura 5.16: Gráfico de *RMSE* (validação) em função das variáveis para diferentes números de componentes principais – PLS para o conjunto 2 92

Figura 5.17: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método LM – conjunto 1 95

Figura 5.18: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método BFGS – conjunto 1 96

Figura 5.19: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método SCG – conjunto 1 96

Figura 5.20: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método LM – conjunto 2 98

Figura 5.21: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método BFGS – conjunto 2 99

Figura 5.22: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método SCG – conjunto 2 99

Figura 5.23: Desvios para a variável de saída conversão em função das amostras para o método LM com parada antecipada. 103

Figura 5.24: Desvios para a variável de saída rendimento em gasolina em função das amostras para o método LM com parada antecipada. 103

Figura 5.25: Desvios para a variável de saída rendimento em GLP em função das amostras para o método LM com parada antecipada. 103

Figura 5.26: Desvios para a variável de saída rendimento em gás combustível em função das amostras para o método LM com parada antecipada. 104

Figura 5.27: Desvios para a variável de saída rendimento em LCO em função das amostras para o método LM com parada antecipada. 104

Figura 5.28: Desvios para a variável de saída rendimento em óleo decantado em função das amostras para o método LM com parada antecipada. 104

Figura 5.29: Desvios para a variável de saída rendimento em coque em função das amostras para o método LM com parada antecipada. 105

Figura 5.30: Desvios para a variável de saída conversão em função das amostras para o método LM com regularização Bayesiana. 106

Figura 5.31: Desvios para a variável de saída rendimento em gasolina em função das amostras para o método LM com regularização Bayesiana. 106

Figura 5.32: Desvios para a variável de saída rendimento em GLP em função das amostras para o método LM com regularização Bayesiana. 107

Figura 5.33: Desvios para a variável de saída rendimento em gás combustível em função das amostras para o método LM com regularização Bayesiana. 107

Figura 5.34: Desvios para a variável de saída rendimento em LCO em função das amostras para o método LM com regularização Bayesiana. 107

Figura 5.35: Desvios para a variável de saída rendimento em óleo decantado em função das amostras para o método LM com regularização Bayesiana. 108

Figura 5.36: Desvios para a variável de saída rendimento em coque em função das amostras para o método LM com regularização Bayesiana. 108

Figura 5.37: Esquema simplificado da modelagem de FCC utilizando redes neurais com PCA nos dados de entrada 112

Figura 5.38: Desvios para a variável de saída conversão em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX. 114

Figura 5.39: Desvios para a variável de saída rendimento em gasolina em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX. 114

Figura 5.40: Desvios para a variável de saída rendimento em GLP em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX. 114

Figura 5.41: Desvios para a variável de saída rendimento em gás combustível em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX. 115

Figura 5.42: Desvios para a variável de saída rendimento em LCO em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX. 115

Figura 5.43: Desvios para a variável de saída rendimento em óleo decantado em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX. 115

Figura 5.44: Desvios para a variável de saída rendimento em coque em função das amostras para o método PCR-RNA com parada antecipada na modelagem individual dos dados da SIX. 116

Figura 5.45: Gráficos dos escores em PC1, PC2, PC3 e PC4 - Dados 1. 118

Figura 5.46: Gráficos dos pesos. PC1, PC2 e PC3 *versus* variáveis de saída - Dados 1. 119

Figura 5.47: Gráfico de T^2 em função das amostras – Dados 1. 120

Figura 5.48: Gráficos dos escores em PC1, PC2, PC3 e PC4 - Dados 2. 121

Figura 5.49: Gráficos dos pesos. PC1 *versus* variável de saída - Dados 2. 122

Figura 5.50: Gráfico de T^2 em função das amostras – Dados 2 123

Figura 5.51: Gráficos dos escores em PC1, PC2, PC3 e PC4 - Dados 3 124

Figura 5.52: Gráfico de T^2 em função das amostras – Dados 3 125

Figura 5.53: Gráficos dos pesos. PC1 *versus* variável de saída - Dados 3 126

Figura 5.54 – Desvios para a variável de saída vazão de gás combustível em função das amostras para o modelo global PCR-RNA com dados da RLAM 129

Figura 5.55 – Desvios para a variável de saída vazão de gás ácido em função das amostras para o modelo global PCR-RNA com dados da RLAM 129

Figura 5.56 – Desvios para a variável de saída vazão de propeno em função das amostras para o modelo global PCR-RNA com dados da RLAM 129

Figura 5.57 – Desvios para a variável de saída vazão de propano em função das amostras para o modelo global PCR-RNA com dados da RLAM 130

Figura 5.58 – Desvios para a variável de saída vazão de GLP em função das amostras para o modelo global PCR-RNA com dados da RLAM 130

Figura 5.59 – Desvios para a variável de saída vazão de gasolina em função das amostras para o modelo global PCR-RNA com dados da RLAM 130

Figura 5.60 – Desvios para a variável de saída vazão de LCO em função das amostras para o modelo global PCR-RNA com dados da RLAM 131

Figura 5.61 – Desvios para a variável de saída vazão de óleo decantado em função das amostras para o modelo global PCR-RNA com dados da RLAM 131

Figura 5.62 - Desvios para a variável de saída vazão de gás combustível (VGC) em função das amostras para o modelo RNA-VC com dados da RLAM 134

Figura 5.63: Desvios para a variável de saída vazão de gás ácido (VGA) em função das amostras para o modelo RNA-RB com dados da RLAM 134

Figura 5.64: Desvios para a variável de saída vazão de propeno (VPE) em função das amostras para o modelo PCA-RNA com dados da RLAM 134

Figura 5.65: Desvios para a variável de saída vazão de propano (VPA) em função das amostras para o modelo PCA-RNA com dados da RLAM 135

Figura 5.66: Desvios para a variável de saída vazão de gás liquefeito do petróleo (VGLP) em função das amostras para o modelo PCA-RNA com dados da RLAM 135

Figura 5.67: Desvios para a variável de saída vazão de gasolina (VGAS) em função das amostras para o modelo RNA-VC com dados da RLAM 135

Figura 5.68: Desvios para a variável de saída vazão de óleo leve de reciclo (LCO) em função das amostras para o modelo PCA-RNA com dados da RLAM 136

LISTA DE TABELAS

Tabela 2.1: Variáveis de entrada para estudo da UMP da SIX. 15

Tabela 2.2: Variáveis de entrada e saída da RLAM usadas nos modelos empíricos estudados 17

Tabela 4.1: Autovalores e variâncias explicadas para o conjunto fictício da Figura 4.1 65

Tabela 5.1: Porcentagem de variância descrita pelas componentes principais para a Matriz de dados **X** (84 amostras e 7 variáveis). 74

Tabela 5.2: Pesos de cada variável nas cinco componentes principais usadas 75

Tabela 5.3: Porcentagem de variância descrita pelas componentes principais para a matriz de dados **X** modificada (82 amostras e 5 variáveis). 78

Tabela 5.4: *RMSE* na etapa de validação para cada variável estudada em função dos modelos de PCR estudados 86

Tabela 5.5: Análise de regressão entre as respostas dos modelos de PCR e as saídas desejadas para cada variável de saída 86

Tabela 5.6: Porcentagem de variância capturada pelo modelo PLS – Conjunto1 89

Tabela 5.7: Porcentagem de variância capturada pelo modelo PLS – Conjunto 2 91

Tabela 5.8: *RMSE* na etapa de validação para cada variável estudada em função dos modelos de PLS estudados 93

Tabela 5.9: Análise de regressão entre as respostas dos modelos de PLS e as saídas desejadas para cada variável de saída 93

Tabela 5.10: *RMSE* globais para os métodos LM, BFGS e SCG – Conjunto 1 95

Tabela 5.11: Melhores resultados para o *RMSE* de validação para as variáveis dependentes nos diferentes métodos considerados – conjunto 1 97

Tabela 5.12: Análise de regressão entre as respostas das redes e as saídas desejadas para cada variável de saída – conjunto 1 97

Tabela 5.13: *RMSE* globais para os métodos LM, BFGS e SCG – Conjunto 2 100

Tabela 5.14: Melhores resultados para o *RMSE* de validação para as variáveis dependentes nos diferentes métodos considerados – Conjunto 2 100

Tabela 5.15: Análise de regressão entre as respostas das redes e as saídas desejadas para cada variável de saída – conjunto 2 101

Tabela 5.13: *RMSE* dos conjuntos 1 e 2 nas etapas de treinamento e validação para o método LM com regularização 105

Tabela 5.14: *RMSE* dos conjuntos 1 e 2 nas etapas de treinamento e validação para o método PLS com redes neurais artificiais 109

Tabela 5.15: *RMSE* para o conjunto 2 nas etapas de treinamento e validação para o modelo de redes neurais usando o método LM com regularização – modelagem individual com os dados da SIX 110

Tabela 5.16: *RMSE* para o conjunto 2 nas etapas de treinamento e validação para o método PLS com redes neurais artificiais – modelagem individual com os dados da SIX 111

Tabela 5.17: *RMSE* para o conjunto 2 nas etapas de treinamento e validação para o método PCA-RNA – modelagem individual com os dados da SIX. 112

Tabela 5.18: *RMSE* de validação e treinamento para as topologias de redes que tiveram os menores valores – conjunto 2 113

Tabela 5.19: *RMSE* globais para treinamento e validação de diferentes modelos empíricos globais – Dados da RLAM 127

Tabela 5.20: *RMSE* de cada variável de saída no conjunto de teste obtidos por diferentes

modelos empíricos – Dados da RLAM 128

Tabela 5.21: *RMSE* globais para treinamento e validação de diferentes modelos empíricos globais – Dados da RLAM 132

Tabela 5.22: *RMSE* de cada variável de saída no conjunto de teste obtidos por diferentes modelos empíricos – Dados da RLAM 133

Tabela A.1: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGC, VGA, VPE e VPA. Método LM com validação cruzada. 155

Tabela A.2: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGLP, VGAS, VLCO e VOD. Método LM com validação cruzada 156

Tabela A.3: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGC, VGA, VPE e VPA. Método LM com regularização Bayesiana 156

Tabela A.4: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGLP, VGAS, VLCO e VOD. Método LM com regularização Bayesiana 157

Tabela A.5: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGC, VGA, VPE e VPA. Método PCA-RNA. 157

Tabela A.6: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGLP, VGAS, VLCO e VOD. Método PCA-RNA. 158

NOMENCLATURA

-Latinas

E – Matriz dos resíduos

$e_j(n)$ – sinal de erro da unidade de saída j na iteração n

$u_j(n)$ – ativação da unidade j na iteração n; sinal a ser aplicado à não-linearidade

$f_j(.)$ – função de ativação associada à unidade j

k – número de componentes principais

M – número de camadas

m – número de linhas na matriz de dados linhas e n colunas

n – n-ésimo vetor de entrada (iteração)

n – número de colunas na matriz

N – número de amostras (padrões de treinamento)

P – matriz dos pesos (*loadings*)

S – matriz de dados de saída (saídas desejadas)

$s_j(n)$ – j-ésimo elemento do vetor de saídas

T – matriz dos escores

$w_{i,j}(n)$ – peso sináptico conectando a saída da unidade i à entrada da unidade j na iteração n

X – matriz de dados de entrada (amostras de treinamento)

$x_i(n)$ – i-ésimo elemento do vetor de entradas

$y_j(n)$ – sinal de saída da unidade j na iteração n

Z – Matriz de covariância

-Gregas

α – taxa de aprendizagem

λ - autovalores

-Subscritos

i, j – índices referentes a diferentes neurônios da rede

-Abreviaturas

API – densidade da carga

CAT – catalisador (atividade)

CP – componentes principais

FCC – *Fluid catalytic cracking*

GLP – gás liquefeito do petróleo

LCO – *light cyrcle oil*

MLP – *multilayer perceptron* (*perceptron* de múltiplas camadas)

NIPALS – *non iterative partial least square*

PCA – *Principal component analysis*

PCR – *Principal component regression*

PLS – *Partial least square*

RGAS – rendimento em gasolina

RGC – rendimento em gás combustível

RLAM – refinaria Landulpho Alves

RLCO – rendimento em óleo leve de reciclo

RMSE – root mean square error prediction

RNA – redes neurais artificiais

ROD – rendimento em óleo decantado

SIX – unidade de negócios industrialização do xisto

SVD – decomposição em valores singulares

TCA – temperatura da carga

TCER – temperatura da fase densa do regenerador

TRX – temperatura de reação

UMP – unidade multipropósito

VCA – vazão de carga

VPA – vazão de propano

VPE – vazão de propeno

VvL – vazão de vapor de *lift*

VvD – vazão de vapor de dispersão

CAPÍTULO 1

INTRODUÇÃO

1.1 Introdução

O craqueamento catalítico fluido, conhecido - mesmo no Brasil - pelas iniciais FCC (*fluid catalytic cracking*), foi desenvolvido na década de 50 como um avanço em relação ao processo então existente de craqueamento térmico. Ambos os processos procuram transformar compostos pesados, derivados da destilação do petróleo, em compostos leves, tais como gás liquefeito de petróleo (GLP) e gasolina. O processo de FCC, além de aumentar a produção de gasolina, produz uma gasolina de maior qualidade que a gasolina de destilação atmosférica. Atualmente no Brasil, a gasolina efluente das refinarias é uma mistura da gasolina proveniente da destilação com a gasolina do FCC. O processo de FCC utiliza um catalisador conhecido como zeolito. O craqueamento ocorre durante o contato desse catalisador com a carga vaporizada num reator de transporte vertical conhecido como *riser* com um tempo de residência de alguns segundos.

Em geral, uma unidade de FCC é constituída de três seções: conversão (onde ocorre as reações de craqueamento catalítico fluido), fracionamento e recuperação de gases (onde ocorre a separação dos diversos subprodutos obtidos da seção de conversão). Os produtos obtidos por esse processo são: gasolina, gás liquefeito do petróleo (GLP), gasóleo leve de reciclo (cuja retificação produz diesel de FCC) e óleo decantado (aproveitado, em mistura, como óleo combustível).

A seção de conversão consiste de dois reatores interconectados: um de leito de arraste (*riser*) e outro de leito fluidizado gás-sólido (regenerador). O reator *riser*, onde quase todas as reações de craqueamento endotérmico e deposição de coque no catalisador ocorrem, e o reator regenerador, onde ar é usado para queimar o coque acumulado no catalisador. O calor produzido é carregado pelo catalisador do regenerador para o *riser*.

Assim, além de reativar o catalisador, o regenerador fornece o calor requerido pelas reações endotérmicas de craqueamento. A complexidade do processo de FCC provém das fortes interações entre as variáveis operacionais do *riser* e do regenerador. Além disso, há uma grande incerteza quanto a cinética das reações de craqueamento e desativação do catalisador - pela deposição do coque no catalisador - e quanto a cinética da queima de coque no regenerador (KUMAN *et al.*, 1995).

A região de condições operacionais economicamente atrativas é determinada por diversos parâmetros tais como: temperatura, pressão, qualidade da carga, catalisador, tempo de residência e da distribuição desejada de produtos requisitados. Segundo MICHALOPOULOS *et al.* (2001), na prática, a otimização de uma dada unidade de FCC para a faixa de produtos desejados é, em geral, realizada por tentativa e erro e/ou pela experiência dos operadores.

A modelagem do processo pode ser usada para descobrir o caminho ótimo para um movimento seguro da planta de um estado para outro de forma que minimize a perda de produtos durante a mudança. Os modelos podem ser divididos em duas categorias principais: a) modelos fenomenológicos; b) modelos empíricos.

Os modelos fenomenológicos representam uma relação funcional entre as variáveis que explica as relações de causa e efeito entre as entradas e saídas do processo. Os modelos fenomenológicos podem ser desenvolvidos a partir dos princípios fundamentais, tais como as leis da conservação da massa, energia e momentum, e princípios da engenharia química (BORGES, 2001). Os modelos fenomenológicos são capazes de explicar o fundamento físico do sistema e, no caso do processo de FCC, podem ser encontrados na literatura (HANU e SOHRAB, 1997; KUMAN *et al.*, 1995; JACOB *et al.*, 1976; CHRISTENSEN *et al.*, 1999; DEWACHTERE, *et al.*, 1997; THEOLOGOS *et al.*, 1997; VAN LANDEGHEM et al., 1996).

Modelos fenomenológicos do processo de FCC podem ser usados para descobrir o caminho ótimo para uma mudança de condições operacionais que minimize a perda de produtos desejáveis durante a mudança. No entanto, devido a complexidade das unidades de FCC é muito difícil obter modelos fenomenológicos precisos. Em geral, torna-se necessário fazer simplificações por falta de dados e/ou parâmetros do modelo.

Na prática, a otimização de unidades de FCC para a faixa de produtos desejados geralmente é realizada por tentativa e erro. A desvantagem da abordagem fenomenológica no processo de FCC é que a transição de um estado ao outro deve ser gradual e não é sempre bem sucedida, por causa das complexas interações entre os dois reatores. Em conseqüência, poderia levar à perda de produção e conseqüentemente afetar os lucros (KUMAN *et al.*, 1995).

Os modelos empíricos são constituídos por equações (por exemplo, polinômios), ou banco de dados, que necessitam de informações do processo para estabelecer relações de entrada-saída. Não apresentam, porém, nenhuma equação fundamental que descreva o fenômeno físico-químico existente, mas fornecem uma descrição da relação dinâmica entre as variáveis de entrada e saída. Um tipo de modelagem empírica é feita por redes neurais artificiais (RNA). As RNA são modelos relativamente simples. Devido a sua estrutura não-linear altamente interconectada, possuem capacidade de auto-organização, possibilitando o aprendizado diretamente a partir de dados do processo. Com isto sua aplicação torna-se uma boa alternativa na modelagem de qualquer processo químico não-linear. Estes modelos possuem as vantagens de precisão, pequeno tempo para o desenvolvimento e poder de adaptação às mudanças nas condições do processo. O fato de apresentarem uma velocidade de execução elevada torna-as bastante indicadas para otimização de processos em tempo real (BORGES, 2001).

As redes neurais *perceptron* de múltiplas camadas (MLP – *MultiLayer Perceptron*) com o algoritmo *backpropagation* têm sido usadas com sucesso na modelagem de processos químicos (HOSKINS e HIMMELBLAU, 1988; BULSARI, 1995). No processo de FCC temos aplicações em identificação e controle do processo (VIEIRA, 2001; SANTOS *et al.*, 2000; ALARADI e ROHANI, 2002) e na modelagem (BOLLAS *et al.*, 2003; MICHALOPOULOS *et al.*, 2001; BATISTA, 1996; MCGREAVY *et al.*, 1994).

Existem diversos métodos para o treinamento de redes MLP com o *backpropagation* cujas propriedades de convergência são comprovadamente superiores às do algoritmo clássico quando aplicados a uma grande variedade de problemas de treinamento. Tais algoritmos utilizam técnicas de otimização numérica como os métodos do gradiente conjugado, quase-Newton e Levenberg-Marquardt (DEMUTH e BEALE, 2002).

Um outro tipo de modelagem empírica é a que utiliza técnicas de estatística multivariada. Estes métodos estatísticos de regressão podem ser utilizados para construção de modelos de predição entrada-saída, que podem ser utilizados para aproximar relações complexas envolvendo regiões limitadas das variáveis estudadas (BAFFI *et al.*, 1999). Esta afirmação é baseada na suposição de que relações não-lineares podem ser aproximadas localmente por modelos lineares. As técnicas de regressão de componentes principais (PCR) e de mínimos quadrados parciais (PLS) têm-se mostrado como ferramentas poderosas na solução de problemas em que os dados são ruidosos e altamente correlacionados e ainda, quando apenas um número limitado de amostras está disponível para a construção de modelos (OLIVEIRA-ESQUERRE, 2003).

Diversos trabalhos que utilizam a metodologia combinada de redes neurais com estatística multivariada (quimiometria) em processos químicos podem ser encontrados na literatura (ADEBIYI e CORRIPIO, 2003; MALTHOUSE et al., 1996; JIA *et al.*, 1998; KOURTI E MACGREGOR, 1995). No entanto, até o momento, não foi encontrada uma aplicação desta abordagem ao processo de FCC.

1.2 Objetivos do Trabalho

O presente trabalho teve como propósito mostrar metodologias empregadas para modelar o craqueamento catalítico fluido via redes neurais artificiais e métodos estatísticos multivariados (quimiometria), visando obter um modelo que melhor represente o fenômeno estudado.

Os principais objetivos deste trabalho foram:

- Aplicação de modelos baseados em redes neurais artificiais utilizando redes do tipo MLP com diferentes algoritmos de otimização dos pesos da rede;
- Uso de ferramentas da quimiometria (PCR e PLS) para gerar modelos lineares e não-lineares (combinados com redes neurais) para a planta piloto de FCC;
- Aplicação da análise dos componentes principais como ferramenta para detecção de amostras com comportamento anômalo e para redução da dimensionalidade do problema estudado;

- Uso de diferentes métodos para melhorar a capacidade de generalização das redes neurais estudadas;
- Comparação dos resultados obtidos por cada modelo visando obter o modelo com maior poder de predição para a planta piloto de FCC;
- Aplicar a metodologia desenvolvida em uma unidade industrial de craqueamento catalítico fluido.

Os dados experimentais foram gerados na planta piloto de FCC da Petrobras, localizada na Usina de Xisto. Essa planta apresenta as condições necessárias para gerar o conjunto de dados para o treinamento e posterior validação da rede neural. Os dados experimentais são referentes ao reator *riser*.

Com base nos resultados obtidos na planta piloto, as técnicas citadas anteriormente foram utilizadas para modelar a unidade industrial de FCC da Petrobras da refinaria Landulpho Alves (RLAM), a qual forneceu um conjunto de dados da produção diária referente ao período de 01/01/2002 a 01/05/2003.

O benefício imediato esperado é ter uma ferramenta para auxiliar na realização dos testes que serão executados na planta piloto: planejamento e análise. O modelo empírico também poderá ser usado como ferramenta que irá auxiliar na validação de modelos fenomenológicos do processo e como ferramenta para otimização de *riser* de FCC que é objetivo final da Petrobras dentro do projeto de aperfeiçoamento das unidades da companhia.

1.3 Organização do Texto

Este trabalho foi dividido em seis capítulos. No primeiro capítulo foi feita uma introdução concisa dos propósitos do trabalho. Os capítulos 2, 3 e 4 apresentam os assuntos de interesse da tese (craqueamento catalítico, redes neurais artificiais e estatística multivariada).

No segundo capítulo é feita uma descrição do processo de craqueamento catalítico fluido considerando a planta piloto localizada em São Mateus do Sul – PR. Também são

consideradas as variáveis operacionais do processo e quais dessas variáveis foram utilizadas para obtenção dos modelos (planta piloto e unidade industrial).

No capítulo três é feito um estudo sobre redes neurais artificiais em que é dada uma maior atenção às redes *perceptron* de múltiplas camadas e aos algoritmos de otimização estudados e aplicados nas modelagens. O capítulo é finalizado mostrando as metodologias usadas para definir a rede MLP (forma de treinamento, número de camadas e de neurônios ocultos, divisão das amostras e generalização).

No capítulo quatro é apresentado um estudo referente a estatística multivariada. É feita uma breve introdução ao assunto e, em seguida, são apresentados os modelos de regressão que serão aplicados ao conjunto de dados bem como a metodologia usada.

No capítulo cinco são mostrados todos os resultados obtidos utilizando as metodologias descritas nos capítulos anteriores e algumas discussões acerca dos mesmos também são feitas.

O capítulo seis apresenta as conclusões deste trabalho com base nos resultados descritos no capítulo cinco. A tese termina com as sugestões para trabalhos futuros, referências bibliográficas e apêndices.

CAPÍTULO 2

DESCRIÇÃO DO PROCESSO DE CRAQUEAMENTO CATALÍTICO FLUIDO

2.1 Introdução

O craqueamento catalítico é um processo de refino que visa aumentar a produção de gasolina e GLP de uma refinaria, pela conversão de cortes pesados provenientes da destilação do petróleo (gasóleo e resíduos) em frações mais leves. É um processo largamente utilizado em todo o mundo, uma vez que a demanda de gasolina em vários países é superior a dos óleos combustíveis. O craqueamento catalítico corrige a produção de gasolina e GLP, suplementando a diferença entre a quantidade obtida diretamente do petróleo e a requerida pela refinaria de modo a atender ao mercado de sua área de influência.

O craqueamento catalítico fluido ou FCC (*Fluid Catalytic Cracking*) é hoje um processo largamente difundido em todo mundo, devido principalmente a dois fatores. O primeiro deles consiste no fato de contribuir eficazmente com a refinaria no sentido de ajustar sua produção às reais necessidades do mercado consumidor local, devido à sua grande flexibilidade operacional.

O segundo fator que tornou o processo consagrado está ligado ao aspecto econômico. Transformando frações residuais, de baixo valor comercial, em derivados nobres de alto valor, tais como gasolina e GLP, o craqueamento catalítico aumenta em muito os lucros da refinaria, devido a sua extraordinária rentabilidade.

Em geral, uma unidade de craqueamento catalítico fluido é constituída de três seções: conversão (onde ocorre as reações de craqueamento catalítico), fracionamento (onde ocorre a separação dos diversos subprodutos obtidos da seção de conversão) e recuperação de gases. Os produtos obtidos por esse processo são: gasolina, gás liquefeito do petróleo (GLP), gasóleo leve de reciclo (cuja retificação produz diesel de FCC) e óleo decantado (aproveitado, em mistura, como óleo combustível). Aqui será considerada

apenas a seção de conversão, pois os dados experimentais fornecidos são referentes ao *riser*.

2.2 Descrição do processo (ABADIE, 1997)

O petróleo admitido em uma refinaria é enviado à destilação atmosférica, onde são separadas as frações mais leves. O produto de fundo da torre de destilação atmosférica é destilado sob vácuo. O gasóleo de vácuo, a fração mais leve da destilação a vácuo, é tratado na unidade de craqueamento catalítico (FCC), onde é transformado em produtos de maior valor comercial. Assim, uma unidade de craqueamento catalítico fluido, além de adequar a produção de derivados à demanda do mercado, aumenta a lucratividade de uma refinaria de petróleo.

A Figura 2.1 mostra um tipo de arranjo de conversor mostrando a interação entre todos os equipamentos. Este figura será usada como exemplo para descrição do processo a seguir.

A carga proveniente da destilação a vácuo (gasóleo), após penetrar na unidade, é aquecida com produtos quentes que saem do processo através de permutadores do sistema de pré-aquecimento, e é encaminhada à base do *riser*. Neste ponto recebe uma grande quantidade de catalisador em alta temperatura (650°C – 750°C), o que provoca a instantânea vaporização do óleo, fluidizando o catalisador.

O *riser* é uma tubulação de grande diâmetro, por onde sobe a mistura de catalisador e vapores de hidrocarbonetos. As moléculas vaporizadas penetram nos poros do catalisador, onde ocorrem efetivamente as reações de craqueamento enquanto, progressivamente, o coque vai se depositando na superfície dos sólidos. A velocidade de escoamento ao longo do *riser* é bastante elevada, fazendo com que o tempo efetivo da reação seja muito pequeno (1 - 4 s), suficiente, entretanto para que todas as reações desejadas ocorram, formando os produtos. A parte final do *riser* é colocada no interior do vaso de separação.

O vaso de separação é destinado a propiciar um espaço físico para que ocorra a separação inercial entre as partículas do catalisador e os gases provenientes do craqueamento. Esta separação é feita pela diminuição súbita da velocidade dos vapores em

ascensão e pelo aumento do diâmetro do equipamento. A temperatura dos gases é aproximadamente a mesma da saída do *riser*, situando-se entre 490°C – 550°C, conforme o tipo da carga, o catalisador e o interesse na maximização de um determinado produto (GLP ou gasolina).

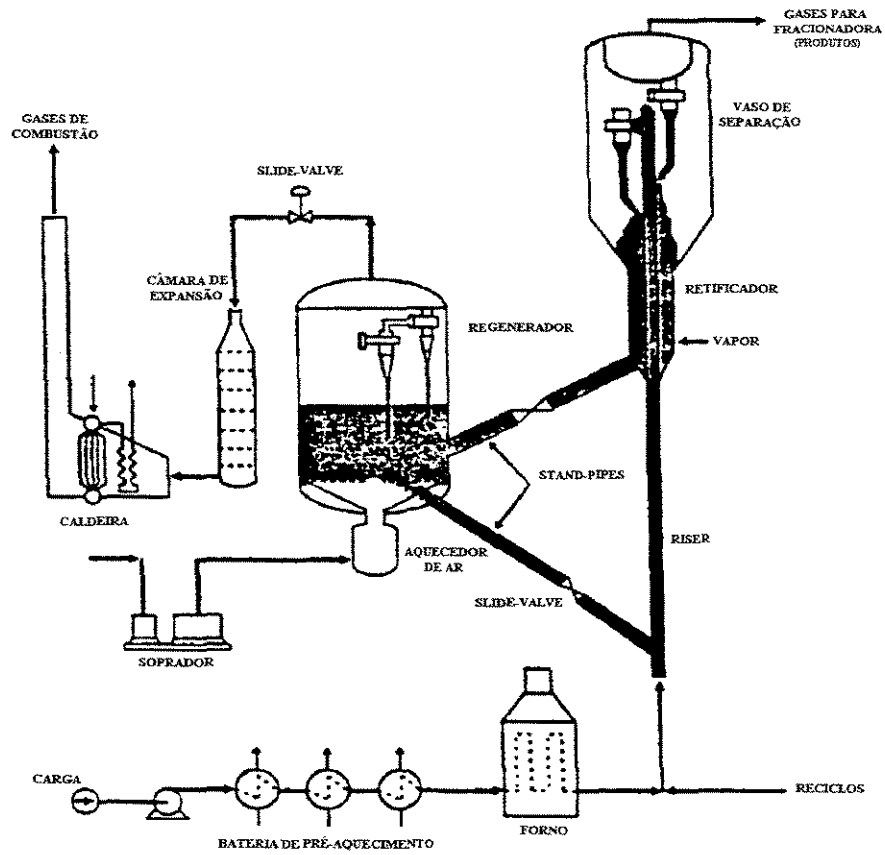


Figura 2.1: Conversor UOP.

As partículas finas de catalisador que sobem junto com a corrente gasosa (vapores de hidrocarbonetos craqueados, vapor d'água e gases inertes), são retiradas por equipamentos fixos denominados ciclones, e devolvidas ao leito de catalisador. Tais equipamentos, sem qualquer peça móvel, têm sua atuação baseada unicamente na ação da força centrífuga. A grande vazão de gases produzidos no craqueamento, carreando as partículas sólidas, tem como único caminho para deixar o vaso de separação a passagem obrigatória pelo conjunto de ciclones. A entrada desses gases nesses equipamentos é feita

de forma tangencial e em alta velocidade por causa de uma área reduzida de passagem. Devido a isso, intensifica-se a ação da força centrífuga, arremessando contra as paredes internas dos ciclones as partículas sólidas (finos do catalisador), enquanto a corrente gasosa, com um teor de pó bastante reduzido, sai pela parte superior de cada equipamento, sendo reunida num local de coleta, denominada câmara plena. O efluente gasoso, após passar por essa câmara, segue através de uma linha de transferência para a seção de fracionamento onde, por meio de uma torre de destilação, há uma separação preliminar dos produtos.

Imediatamente após a saída do *riser*, as partículas de catalisador, recobertas por coque, sob ação do próprio peso tendem a cair ao fundo do vaso de separação. Contudo, vapores de hidrocarbonetos ocupam os poros de catalisador e os espaços entre as partículas. De modo a recuperar parte destes produtos, principalmente os gases que estão alojados nos espaços interparticulares, o catalisador gasto passa pelo retificador ou *stripper*. Os internos desse equipamento, colocado imediatamente abaixo do vaso de separação, consistem de uma série de defletores convergentes-divergentes ou de defletores alternados conhecidos como chicanas. Após a chicana mais inferior, é colocado um anel com vários furos, por onde é injetado vapor d'água para a retificação.

A contínua descida do catalisador gasto através das chicanas, recebendo em contracorrente o fluxo ascendente de vapor proveniente do fundo do retificador, permite a recuperação de uma quantidade considerável de hidrocarbonetos, evitando assim que eles sejam enviados ao regenerador e queimados. O vapor d'água que faz a operação de retificação mistura-se com os gases de craqueamento no vaso de separação, seguindo com eles para a seção de fracionamento. O catalisador gasto retificado sai pelo fundo do *stripper* e por meio de um duto de grande porte, denominado *stand-pipe*, é transferido ao regenerador.

A função do regenerador é queimar os depósitos carbonosos alojados na superfície do catalisador, transformando-os em gases de combustão, enquanto, devido a essa eliminação, reativa-se as partículas. Essa combustão ocorre devido à alta temperatura de chegada do catalisador ao regenerador (490°C – 550°C), à presença do coque depositado e à grande vazão de ar injetada pela parte inferior do regenerador.

A queima do coque causa não só a regeneração do catalisador, mas também uma intensa liberação de calor, elevando a temperatura do catalisador regenerado para cerca de 650°C – 740°C. Essa grande geração de energia provinda da combustão do coque é a maior fonte de calor para o processo, uma vez que, devido à contínua remoção de catalisador regenerado levado à base do *riser*, a energia carreada por ele é suficiente para não só aquecer e vaporizar a carga, como também suprir todas as necessidades térmicas das reações de craqueamento, permitindo que, no vaso de separação, a temperatura de saída do catalisador possa atingir ainda 490°C – 550°C.

O ar requerido para a queima é fornecido por um soprador de ar de grande capacidade conhecido como *blower*, e é injetado no regenerador através de um distribuidor localizado no fundo do vaso chamado de *pipe-grid*. Este pode ser de vários formatos, em função da concepção do regenerador e do conversor como um todo.

A passagem de ar através da massa de catalisador no interior do regenerador causa a formação de bolhas, produzindo um efeito semelhante à de um líquido em ebulição. O íntimo contato entre o ar, progressivamente transformado em gases de combustão, e os sólidos, permite a formação de um leito fluidizado, ou seja, o conjunto gases-partículas tende a se comportar como se fosse um líquido. Essa região onde predomina a massa de sólidos é conhecida como fase densa. Acima do leito há uma outra região onde predomina agora os gases de combustão, existindo porém uma grande quantidade de partículas arrastadas, conhecida como fase diluída. Esses finos de catalisador arrastados são quase totalmente recuperados pelo conjunto de ciclones do regenerador, normalmente de duplo estágio (dois ciclones em série). Os gases de combustão, inertes e finos não recuperados deixam o segundo estágio de cada conjunto de ciclones e alcançam a câmara plena do regenerador, que serve não só como coletora dos gases, mas também como ponto de sustentação dos ciclones.

A composição volumétrica desses gases tomada em base seca é aproximadamente a seguinte: N_2 = 80%, CO = 10%, CO_2 =10%. A temperatura de saída dos gases pode atingir valores superiores a 730°C, sendo extremamente elevado seu conteúdo energético. Para o aproveitamento desta grande quantidade de calor, os gases são dirigidos a um equipamento conhecido como caldeira de CO.

Dentro da caldeira, os gases recebem uma quantidade adicional estequiométrica de ar e, por meio de um conjunto auxiliar de maçaricos, o CO é transformado quase integralmente em CO_2. Esta reação faz com que os gases atinjam temperaturas próximas a 1000°C no interior da caldeira. Toda essa energia é utilizada na geração de uma grande quantidade de vapor d'água em alta pressão, que, uma vez produzido, poderá ser consumido no acionamento das grandes máquinas da unidade (*blower* e compressores de gás) e o excedente será fornecido às demais unidades da refinaria. A caldeira de CO é extremamente importante, tanto pela sua grande produção de energia na forma de vapor em alta pressão, quanto pelo fato de eliminar CO dos gases, contribuindo assim como agente antipoluidor. Depois da passagem pela caldeira, os gases são lançados na atmosfera por uma chaminé de grande altura.

De forma a compatibilizar a pressão de trabalho do regenerador (2,0 kgf/cm^2 – 4,0 kgf/cm^2) e da caldeira, os gases devem passar por um sistema redutor de pressão. Este é constituído de um par de válvulas corrediças paralelas (*slide-valves*) e de uma torre com vários pratos perfurados conhecida como câmara de orifícios ou câmara de expansão. As *slide-valves* têm por função causar uma perda de carga na corrente de gases de combustão, ao mesmo tempo em que fazem o controle da pressão do regenerador e indiretamente o diferencial de pressão entre o reator e o regenerador.

Um pequeno forno aquecedor de ar, parte integrante da linha de injeção de ar para o distribuidor, é um equipamento complementar ao conversor. Somente utilizado por ocasião da partida da unidade, sua função é aquecer o ar e fornecer o calor necessário para elevar a temperatura da fase densa do regenerador ao ponto em que se possa iniciar a combustão do coque na superfície do catalisador.

O arranjo relativo entre o *riser*, vaso de separação e o regenerador faz com que haja vários tipos de conversores de FCC. As maiores projetistas mundiais do ramo são a UOP, KELLOGG, EXXON, AMOCO, TEXACO e SHELL, sendo que as duas primeiras estão destacadamente à frente das demais.

2.3 Os dados da planta piloto de FCC da SIX – Petrobras

Localizada em São Mateus do Sul – PR, a planta piloto multipropósito de craqueamento catalítico fluido da Unidade de Negócios Industrialização do Xisto (SIX), da Petrobras, oferece condições ideais para realizar um estudo das variáveis operacionais do processo podendo obter dados experimentais numa ampla faixa de operação. A Figura 2.2 mostra um esquema simplificado da planta piloto multipropósito (UMP) de FCC da SIX que representa, de uma maneira simplificada, um *scale down* de uma unidade industrial.

A Figura 2.3 mostra o reator em detalhes. Na Figura 2.3 foram omitidos os dados geométricos do equipamento por motivos do acordo de sigilo industrial assinado com a Petrobras.

Segundo BOLLAS *et al.* (2003): "Os modelos de planta piloto são muitas vezes utilizados para desenvolver predições precisas e modelos de otimização. A justificativa é que a operação das unidades de planta piloto de FCC pode ser facilmente adaptada a uma ampla faixa de condições (propriedades da carga de alimentação, condições de operações do catalisador, etc.)".

O projeto de aperfeiçoamento de *riser* de FCC contempla 11 variáveis de entrada a serem estudadas entre os limites operacionais da unidade e que foram consideradas pela Petrobras de importância relevante para a tecnologia de FCC, conforme listado na Tabela 2.1. O objetivo do projeto de aperfeiçoamento é a otimização dos *risers* existentes na companhia, considerando uma perspectiva de avanço tecnológico. Dentro desse contexto, o projeto visa obter uma quantidade considerável de dados experimentais que possam ser usados como parâmetros de avaliação no desenvolvimento de modelos matemáticos de simulação multidimensional.

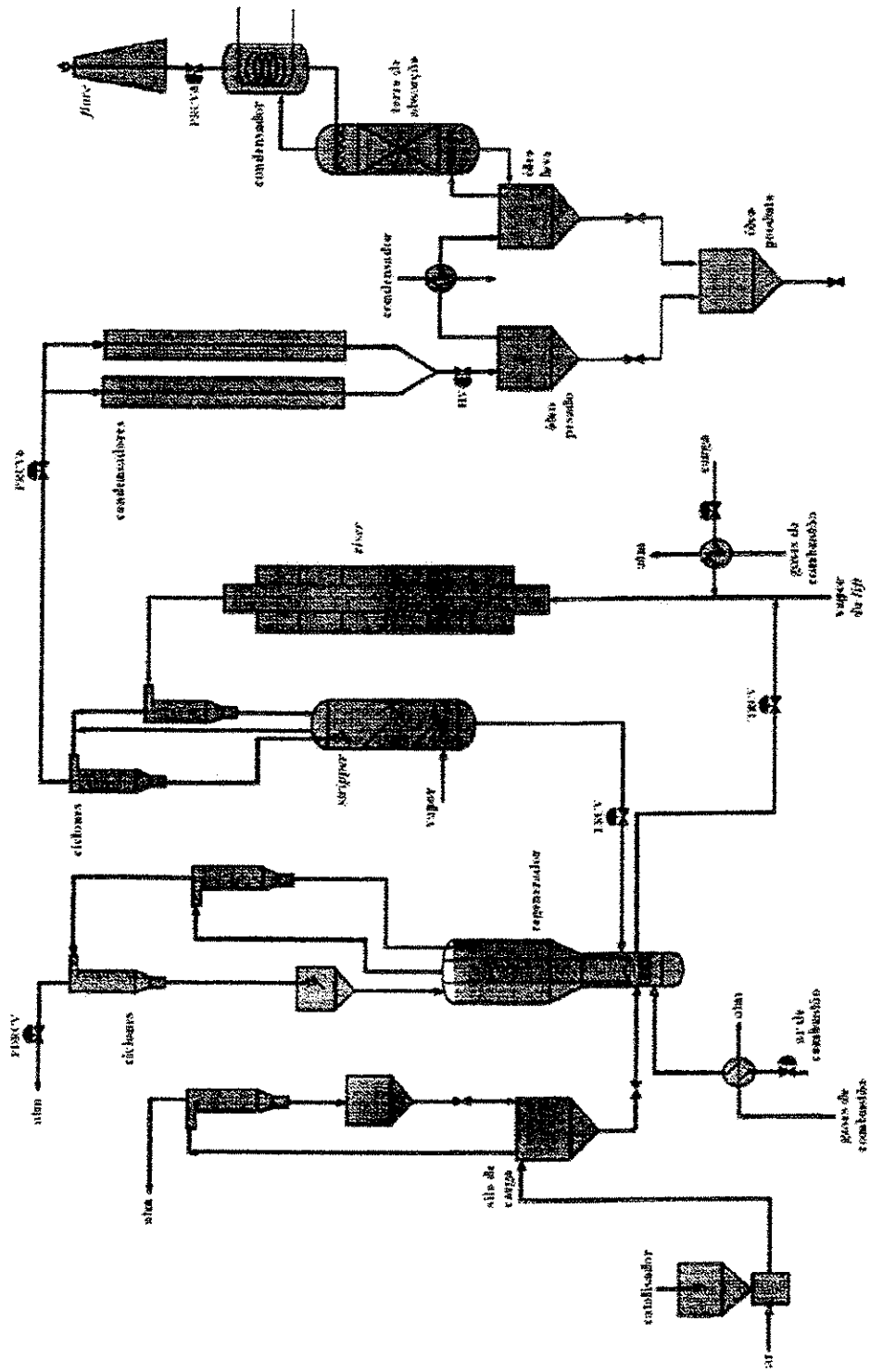


Figura 2.2: Esquema da unidade multipropósito (UMP) de FCC da SIX - Petrobras.

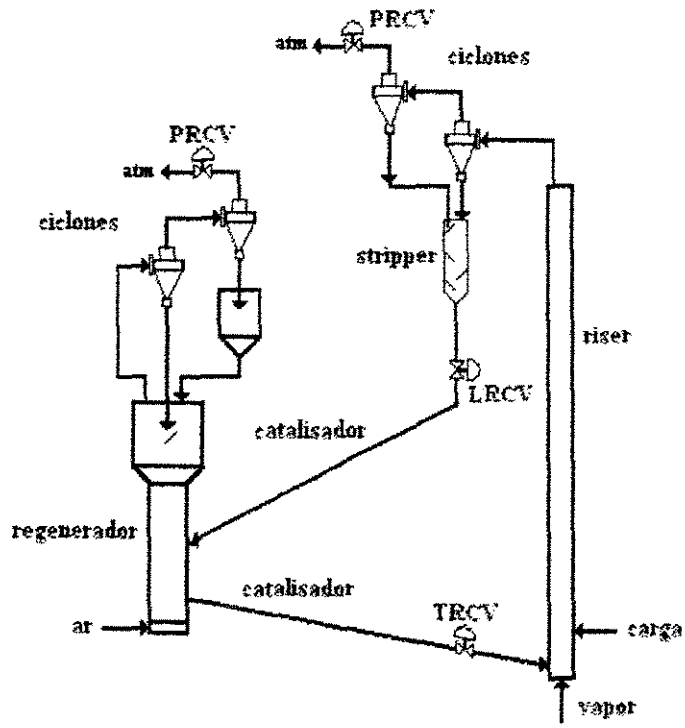


Figura 2.3: Diagrama do circuito regenerador – *riser*– *stripper*, da unidade multipropósito de FCC – U-144 da SIX/PETROBRÁS.

Tabela 2.1: Variáveis de entrada e saída para estudo da UMP da SIX.

ORDEM	VARIÁVEIS DE ENTRADA	SIGLA	VARIÁVEIS DE SAÍDA	SIGLA
1	Vazão da carga	VCA	Conversão global	CONV
2	Temperatura da carga	TCA	Rendimento em gasolina	RGAS
3	Elevação do *riser*	H	Rendimento em GLP	RGLP
4	Temperatura de reação	TRX	Rendimento em gás combustível	RGC
5	Temperatura da fase densa do regenerador	TCER	Rendimento em LCO	RLCO
6	Vazão do vapor de *lift*	VvL	Rendimento em óleo decantado	ROD
7	Vazão do vapor de Dispersão	VvD	Rendimento em coque	RCOQ
8	Vazão de vapor de *Stripper*	VvS		
9	Pressão	P		
10	Tipo de carga	CER		
11	Tipo de Catalisador	CAT		

Inicialmente foram priorizadas três variáveis (consideradas principais pela Petrobras) para serem estudadas no processo: elevação do *riser* (H), temperatura de reação (TRX) e temperatura da fase densa do regenerador (TCER). Essas três variáveis foram modificadas dentro dos limites operacionais da UMP, em três níveis resultando em 27 condições experimentais. As variáveis vazão de vapor de dispersão (VvD), vazão de vapor de *lift* (VvL), temperatura de carga (TCA) e vazão de carga (TCA) foram modificadas numa faixa mais reduzida de operação. O objetivo foi repetir as condições operacionais citadas anteriormente. Já as variáveis tipo de catalisador (CAT), tipo de carga (CAR), pressão (P) e vazão de vapor de *stripper* (VvS) permaneceram fixas de acordo com os valores fornecidos pela SIX – Petrobras e por isso não fizeram parte dos modelos desenvolvidos para a SIX.

As faixas de operação aplicadas nos experimentos foram definidas com base na busca de máximos rendimentos dos produtos mais desejados que são a gasolina e o GLP. No entanto, foram determinados também os rendimentos de outros produtos do processo tais como LCO, GC, OD e COQUE, que serão usados também neste trabalho. Foram utilizados dados de 91 experimentos.

2.4 Os dados da unidade industrial de FCC da RLAM – Petrobras

Localizada na rodovia BA – 523, km 4, São Francisco do Conde – BA, a refinaria Landulpho Alves (RLAM) forneceu os dados da unidade U-39 para treinamento, validação e teste dos modelos que foram empregados. O conjunto de dados refere-se a produção diária no período de 01/01/2002 a 01/05/2003. Excluindo as lacunas existentes no conjunto de dados fornecido, restaram 328 amostras.

A Tabela 2.2 mostras as variáveis de entrada e saída fornecidas pela RLAM. Pode-se observar que os dados da RLAM apresentam mais variáveis de entrada em relação aos dados da SIX pois, no caso da RLAM, houve modificações nas características da carga e há medidas sobre na atividade do catalisador. Também observa-se que há mais vazões nos vapores de retificação, dispersão e arraste em relação a planta piloto da SIX, aumentando a complexidade do modelo a ser obtido.

Tabela 2.2: Variáveis de entrada e saída da RLAM usadas nos modelos empíricos estudados.

ORDEM	VARIÁVEIS DE ENTRADA	SIGLA	VARIÁVEIS DE SAÍDA	SIGLA
1	Vazão de carga	VCA	gás combustível	VGC
2	Temperatura da carga	TCA	gás ácido	VGA
3	Temperatura de reação	TRX	propeno	VPE
4	Temperatura da fase densa	TCER	propano	VPA
5	Densidade da carga	API	GLP	VGLP
6	Vazão de vapor do retificador (topo)	VvRt	gasolina	VGAS
7	Vazão de vapor do retificador (meio)	VvRm	LCO	VLCO
8	Vazão de vapor do retificador (base)	VvRb	óleo decantado	VOD
9	Vazão de vapor de dispersão (superior)	VvDs		
10	Vazão de vapor de dispersão (inferior)	VvDi		
11	Vazão de vapor de arraste (radial)	VvAr		
12	Vazão de vapor de arraste (axial)	VvAa		
13	Catalisador (atividade)	CAT		

2.5 Variáveis de entrada

A elevação do *riser* (H) é uma variável relacionada a posição do bico injetor de carga no *riser* e foi variada visando o estudo do tempo de contato para que o mesmo ficasse dentro de um intervalo que fosse condizente com os tempos de contato existentes nas unidades industriais da Petrobras.

A temperatura de reação (TRX) é a variável de maior importância no craqueamento catalítico e a mais usada para o controle da conversão. A temperatura de reação ideal, para uma determinada carga, é aquela na qual se obtém a maior conversão possível (DECROOCQ, 1984). A temperatura que tomamos como referência como sendo a do reator é a temperatura de saída do *riser*. Esta é a menor temperatura reinante na zona de reação, pois o processo é endotérmico e a temperatura varia ao longo do riser, não sendo de fato aquela em que ocorrem as reações.

A temperatura da fase densa do regenerador (TCER) é uma variável de extrema importância para eficiência da regeneração do catalisador. Quanto mais alta for esta temperatura, menor será o teor de coque no catalisador regenerado e maior será sua atividade (DECROOCQ, 1984). Por outro lado, temperaturas excessivamente altas podem causar desativação térmica, principalmente se os teores de metais depositados forem também elevados.

A temperatura da carga (TCA) é uma das mais importantes variáveis operacionais do conversor. Ela é utilizada normalmente para acertar o balanço térmico, e em especial, a temperatura do regenerador. A temperatura que a carga atinge ao ser injetada na base do *riser* é fornecida pela bateria de pré-aquecimento e pelo forno, oscilando numa faixa entre 100°C e 400°C.

A vazão da carga (VCA) é definida em função do plano de operação da refinaria, na qual leva-se em consideração vários outros fatores alheios à unidade de craqueamento. Estocagens de gasóleo e de resíduos, paradas, reduções e maximizações de carga nas unidades de destilação da refinaria são fatores que influenciam a vazão de carga a ser processada. Em última análise, ela é definida de modo a atender o mercado consumidor da área de influência da refinaria. Assim como a elevação do riser, a vazão de carga fresca é um variável que influenciará bastante o tempo de contato.

2.6 Variáveis de saída (ABADIE, 1997)

2.6.1 Conversão

A conversão em base mássica é definida conforme feito na indústria de refino de petróleo:

$$\textbf{Conversão} = \left(100 - \left(\%LCO + \%OD\right)\right) \tag{2.1}$$

Onde $\%LCO$ e $\%OD$ representam as porcentagens mássicas de óleo leve de reciclo (LCO) e óleo decantando (OD), respectivamente.

Esta definição de conversão mássica fornece o percentual da carga que é transformada em gás combustível (GC), gás liquefeito do petróleo (GLP), nafta de craqueamento (gasolina) e coque, mas não informa nada sobre a produção específica de gasolina e GLP, que é de interesse. Por isso, a conversão é usada apenas como elemento de comparação no acompanhamento do desempenho da unidade.

2.6.2 Rendimentos dos Produtos

O rendimento dos produtos é, na realidade, o grande objetivo a ser alcançado na operação da unidade. Alcançar o máximo rendimento de uma determinada fração, com uma certa qualidade de carga e com uma atividade característica do catalisador é o desafio a ser atingido pela refinaria e normalmente lhe proporciona uma boa rentabilidade.

O FCC pode ser operado visando a maximização de dois produtos: nafta de craqueamento (gasolina) ou GLP. Caso exista na refinaria uma unidade de tratamento de diesel instável (LCO), a maximização dessa fração pode também tornar-se econômica.

A maximização dessas frações irá depender, evidentemente, da severidade operacional, em função da carga processada. Uma vez mantida constante a carga e o catalisador, pode-se direcionar o perfil dos produtos atuando nos outros fatores que influenciam o craqueamento catalítico.

Para a maximização de GLP, deve-se operar o conversor com bastante severidade, ou seja, alta temperatura do reator, elevada razão catalisador/óleo e tempo de contato alto.

Se é desejado maximizar a produção de gasolina (nafta), o conversor deve operar com condições de média severidade, tais como, temperatura moderada no reator, média razão catalisador/óleo e médio tempo de contato.

Naturalmente, quando é desejável aumentar a produção de LCO no processo, deve-se operar com baixas condições de severidade, ou seja, menores temperaturas de reação, baixa razão catalisador/óleo e pouco tempo de contato. No entanto, operando dessa forma, tem-se baixas conversões, o que poderia prejudicar a rentabilidade do processo.

A decisão de maximizar gasolina, GLP ou LCO é determinada não só por fatores econômicos, mas também pela necessidade de abastecimento local, limitações da unidade, etc.

2.7 Considerações finais sobre o capítulo

O objetivo deste capítulo foi apresentar o processo de craqueamento catalítico fluido de forma geral e mostrar as variáveis operacionais que serão consideradas no estudo tanto para a SIX quanto para a RLAM. Os dados da SIX apresentaram menos variáveis de entrada em relação aos dados da RLAM. As variáveis de saída para SIX estão em termos de rendimentos enquanto que os dados da RLAM estão em termos das vazões diárias.

Para os dois conjuntos de dados foram feitos dois tipos de modelagem: 1) O modelo global no qual todas as variáveis de saída são fornecidas pelo modelo e 2) os modelos individuais onde há uma rede para cada variável de saída.

Nos capítulos seguintes os dados passarão por uma análise estatística em busca de amostras anômalas (*outliers*) e serão pré-processados para que possam ser aplicados aos métodos empíricos de modelagem.

CAPÍTULO 3

REDES NEURAIS ARTIFICIAIS

3.1 Introdução

Redes neurais artificiais (RNA) são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura de organismos inteligentes e que adquirem conhecimento através da experiência.

De acordo com HAYKIN (2001), uma rede neural artificial é um processador maciçamente paralelo distribuído, constituído de unidades de processamento simples, que têm propensão natural para armazenar conhecimentos experimentais e torná-los disponíveis para o uso, que se assemelha ao cérebro em dois aspectos:

1. O conhecimento é adquirido pela rede por meio de dados do ambiente, num processo de aprendizagem. Processo de treinamento é chamado de "Algoritmo de Aprendizagem", que tem como finalidade ajustar os pesos sinápticos da rede de uma forma ordenada para alcançar um objetivo desejado.
2. As conexões entre os neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido.

Uma rede neural artificial (RNA) é um sistema de processamento de informação que possui algumas características de desempenho em comum com as redes neurais biológicas. Os modelos neurais artificiais têm como principal fonte de inspiração as redes neurais biológicas. A Figura 3.1 apresenta um modelo simplificado de um neurônio biológico e um modelo básico de um neurônio artificial.

Na Figura 3.1, o neurônio biológico apresenta alguns setores de entrada (dendritos) das informações que chegam na forma de pulsos elétricos, uma região onde as informações são processadas (soma), e a saída (axônio) que transmite a informação processada para os dendritos de outros neurônios. O espaço existente entre o axônio de um neurônio e os

dendritos de outros neurônios é chamado de sinapse ou região sináptica. Na mesma figura temos o neurônio artificial o qual foi criado tomando por base o neurônio biológico. O princípio de funcionamento do neurônio artificial é semelhante ao biológico, porém rudimentar. As informações provenientes do meio exterior (ou de outros neurônios anteriores) são multiplicadas pelos respectivos pesos simulando a sinapse que ocorre no neurônio biológico. Após a multiplicação pelos pesos, as informações ponderadas são somadas e esta soma é modificada por uma função de ativação (ou função de transferência) que gera o sinal de saída do neurônio artificial que será transmitido a outro neurônio (ou para o meio exterior).

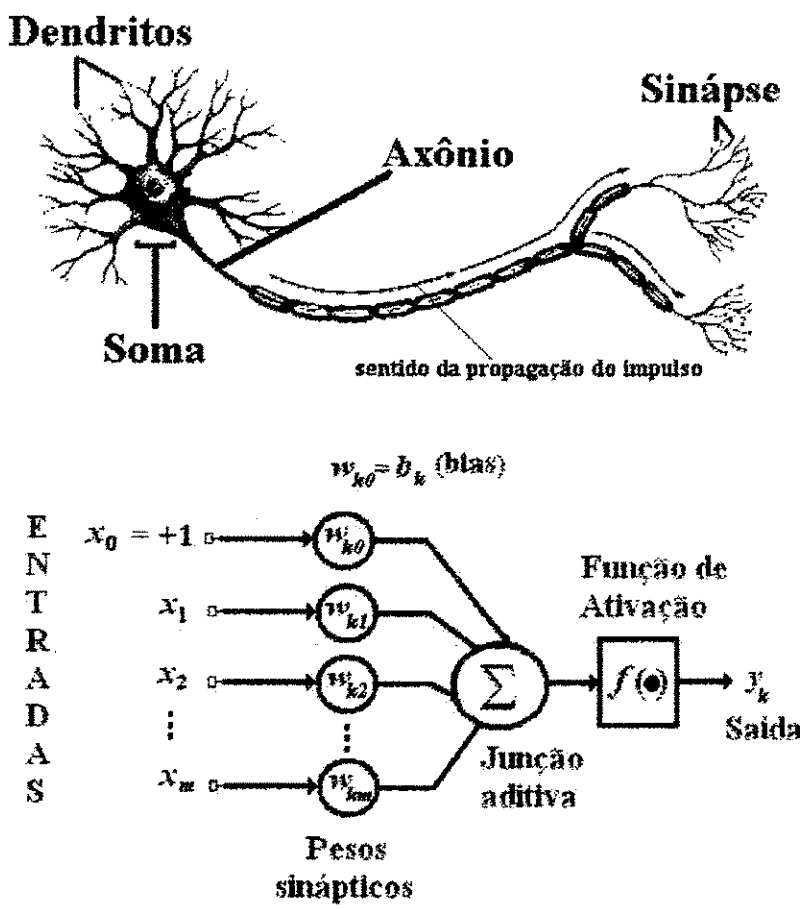


Figura 3.1: Esquema básico de um neurônio biológico e de um artificial

Uma rede neural funciona com vários neurônios, trabalhando como descrito anteriormente, organizados em grupos (ou camadas) seqüenciados. Os dados são alimentados na camada de entrada e a resposta da rede ao seu efeito é observada na saída. Podem existir uma ou mais camadas, denominadas de camada oculta, entre a entrada e a saída da rede. O seu número é dado pelas características de cada sistema.

3.2 Arquitetura da Rede

A arquitetura de uma rede neural artificial depende diretamente do problema que será tratado pela rede. Como parte da definição da arquitetura da rede tem-se: quantidades de camadas, números de neurônios em cada camada e tipo de conexão entre os neurônios (BRAGA, 2000).

Quanto ao número de camadas, pode-se ter:

1. redes de camada única. A forma mais simples de uma rede em camadas surge quando tem-se uma camada de entrada que se projeta para a camada de saída, mas não vice-versa, como mostrado na Figura 3.2 (a) e (d);
2. redes com múltiplas camadas. Redes com múltiplas camadas, ver Figura 3.2 (b) e (c), se distinguem de redes com camada única, pela presença de camada(s) oculta(s).

Quanto aos tipos de conexões entre neurônios, tem-se:

1. *feedforward* ou acíclica. A saída do neurônio na i-ésima camada não pode ter entradas com neurônios em camadas de índice menor ou igual a i, como mostrado na Figura 3.2 (a), (b) e (c);
2. *feedback* ou cíclica. A saída do neurônio na i-ésima camada tem entradas com neurônios em camadas de índice menor ou igual a i, como mostrado na Figura 3.2 (d).

Finalmente, quanto a sua conectividade, tem-se:

1. rede fracamente (ou parcialmente) conectada, como na Figura 3.2 (c);
2. rede completamente conectada, como mostrado na Figura 3.2 (a),(b) e (d).

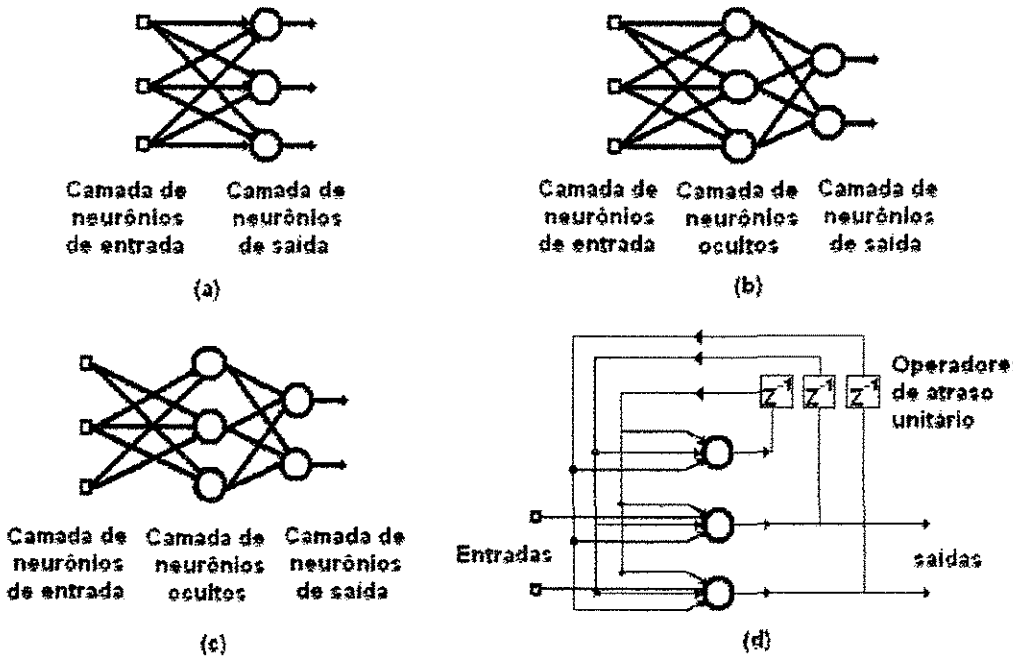


Figura 3.2: Exemplos de arquiteturas de redes neurais artificiais

3.3 Processo de Aprendizagem

É importante para as redes neurais as habilidades de aprender a partir dos seus ambientes e de melhorar o seu desempenho através do treinamento. Uma rede neural aprende acerca do seu ambiente através de um processo interativo de ajustes aplicados a seus pesos sinápticos e níveis de *bias*. Utiliza-se uma definição de aprendizagem que é adaptada de HAYKIN (2001): Aprendizagem é um processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um processo de estimulação pelo ambiente no qual a rede está inserida. O tipo de aprendizagem é determinado pela maneira pela qual a modificação dos parâmetros ocorre.

Esta definição do processo de aprendizagem implica a seguinte seqüência de eventos:

1. a rede neural é estimulada por um ambiente;

2. a rede neural sofre modificações nos seus parâmetros livres (i.e. pesos sinápticos e *bias*) como resultado desta estimulação;
3. a rede neural responde de uma maneira nova ao ambiente, devido às modificações ocorridas na sua estrutura interna.

Como se pode esperar, não há um algoritmo de aprendizagem único para o projeto de redes neurais. Em vez disso, tem-se um conjunto de ferramentas representado por uma variedade de algoritmos de aprendizagem, cada qual oferecendo vantagens específicas. Basicamente, os algoritmos diferem entre si pela forma como são formulados os ajustes dos pesos sinápticos do neurônio. Outro fator a ser considerado é a maneira pela qual uma rede neural se relaciona com seu ambiente, que pode ser supervisionado ou não-supervisionado, que será visto na seção a seguir.

3.4 Modelos de Relacionamento da Rede Neural com o seu Ambiente

O objetivo do processo de aprendizado é encontrar o valor do ajuste do vetor de pesos **w**, podendo ser classificado basicamente em três paradigmas distintos: aprendizado supervisionado, aprendizado não-supervisionado e aprendizado por reforço.

1. O aprendizado supervisionado ou com professor ocorre quando o vetor de padrão possui as características e o rótulo da classe a que o objeto pertence. Freqüentemente, as saídas são fornecidas por um supervisor (especialista) humano.

   Em termos conceituais, pode-se considerar o professor como tendo conhecimento sobre o ambiente, sendo este conhecimento representado por um conjunto de entrada-saída. Entretanto, o ambiente é desconhecido pela rede neural de interesse. Suponha agora que o professor e a rede neural sejam expostos a um vetor de treinamento retirado do ambiente. Em virtude de seu conhecimento prévio, o professor é capaz de fornecer à rede neural uma resposta desejada para aquele vetor de treinamento.

   Os parâmetros da rede são ajustados sob a influência combinada do vetor de treinamento e do sinal do erro. O sinal de erro é a diferença entre a resposta desejada e a resposta real da rede. Esta forma de aprendizagem é usada na

aprendizagem por correção de erro. Tarefas típicas que utilizam aprendizado supervisionado são aproximações de funções e reconhecimento de padrões;

2. No aprendizado não-supervisionado, também referido como auto-organizado, como o nome implica, não há um professor para supervisionar o processo de aprendizagem. Isto significa que não há exemplos rotulados da função a serem aprendidos pela rede. Após os parâmetros livres da rede se ajustarem às regularidades estatísticas dos dados de entrada, ela desenvolve a habilidade de formar representações internas que codificam as características da entrada e desse modo criar automaticamente novas classes. Um exemplo seria a aprendizagem competitiva;
3. O aprendizado por reforço é um modelo intermediário entre aprendizado supervisionado e não-supervisionado. O conjunto de dados não contém a resposta desejada. Em vez disso, a rede retorna um sinal de reforço ou penalidade conforme a melhora ou não do desempenho da rede, sendo que o objetivo é maximizar o reforço e conseqüentemente, a melhora do desempenho.

Na seção a seguir será descrito de forma sucinta o modelo de aprendizagem do *perceptron*, modelo que contribui em muito na área de redes neurais artificiais e que é a base para a Seção 3.6 sobre o *perceptron* de múltiplas camadas.

3.5 O *Perceptron*

Utilizando o modelo de neurônio de McCulloch e Pitts, Frank Rosenblatt escreveu, em 1958, o primeiro conceito de aprendizado das redes neurais artificiais, e demonstrou o Teorema de Convergência do *Perceptron*, em que o algoritmo de aprendizado do *perceptron* sempre converge caso o problema em questão seja linearmente separável.

O aprendizado ou adaptação tem como finalidade calcular o valor a ser aplicado nos pesos w, que tende a encontrar o melhor w_0 que resolve o problema em questão.

Considera-se um neurônio arbitrário da camada de saída de um *perceptron* com vetores de entrada x' e de pesos w', com ativação igual ao produto interno de w e x, $\sum w_i x_i$, isto é, o ângulo g entre w e x, como abaixo:

$$\sum w_i x_i = \begin{cases} = 0, & w \perp x \\ > 0, & g < 90° \\ \leq 0, & g > 90° \end{cases}$$

A condição de ativação do neurônio é quando $\sum w_i x_i = \theta$, onde θ é o valor limiar do neurônio, fazendo $\sum w'x' - \theta = 0$.

Utiliza-se $\{\mathbf{x}, d\}$ como par de treinamento, sendo $\mathbf{x}$ o vetor de entrada, d a resposta desejada de $\mathbf{x}$ e y a resposta produzida pela rede. O erro do neurônio será $e = d - y$ do vetor x. Os valores de y e d no caso do *perceptron* podem ser $y \in \{0, 1\}$ e $d \in \{0, 1\}$, portanto tem-se duas situações para que $\in \neq 0$:

1. a primeira é quando $d = 1$ e $y = 0$; neste caso $e = 1$ e $\sum w'x' < 0$, o que implica que o ângulo entre w' e x' é g (> 90°). De acordo com a Figura 3.3 (a), uma boa opção para a alteração do vetor peso é somar o vetor ηx. Assim $\Delta w = \eta x$ e $w(t + 1) = w(t) + \eta ex$, onde η é a taxa de aprendizado do algoritmo, isto é, o quanto que o vetor peso irá ser modificado;
2. a segunda é quando $d = 0$ e $y = 1$; neste caso $e = -1$ e $\sum w'x' \geq 0$, o que implica que o ângulo entre w' e x' é g (< 90°). De acordo com a Figura 3.3 (b), uma boa opção para a alteração do vetor peso é subtrair o vetor ηx. Assim $\Delta w = -\eta x$ e $w(t + 1) = w(t) - \eta x$, como $e = -1$ pode-se ter $\Delta w = \eta ex$ e $w(t + 1) = w(t) + \eta ex$.

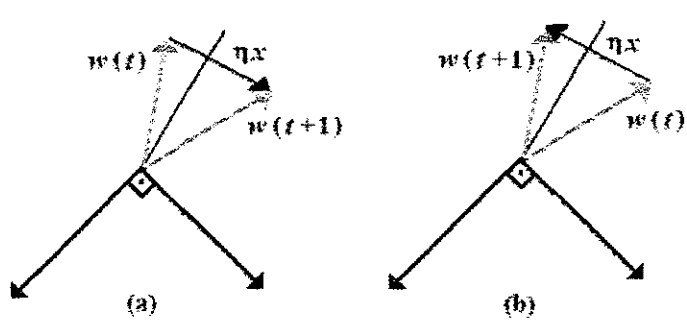


Figura 3.3: Unidade de saída

Apesar de ter causado grande euforia na comunidade científica da época, o *perceptron* não teve vida muito longa, já que as duras críticas de Minsky e Papert a sua capacidade computacional causaram grande impacto sobre as pesquisas em redes neurais artificiais, levando a grande desinteresse pela área durante os anos 70 e início dos anos 80. Esta visão pessimista da capacidade do *perceptron* e das redes neurais de uma maneira geral mudou com as descrições da rede de Hopfield em 1982 e do algoritmo *backpropagation* em 1986. Foi em conseqüência destes trabalhos que a área de redes neurais artificiais ganhou novo impulso, ocorrendo, a partir do final dos anos 80, uma forte expansão no número de trabalhos de aplicação e teóricos envolvendo redes neurais e técnicas correlatas (BRAGA *et al.* 2000).

3.6 O *Perceptron* de Múltiplas Camadas (MLP)

RUMELHART *et al.* (1986) apresentaram a descrição do Algoritmo Retropropagação de erro ou *backpropagation* para arquitetura do *perceptron* de múltiplas camadas, mostrando que a visão de Minsky e Papert sobre o *perceptron* era bastante pessimista.

Os *perceptrons* de múltiplas camadas com retropropagação de erro têm sido aplicados com sucesso para resolver diversos problemas difíceis, através do seu treinamento de forma supervisionada com um algoritmo *backpropagation*. Este algoritmo é baseado na regra de aprendizagem por correção de erro (HAYKIN, 2001).

A estrutura do neurônio artificial, mostrado na Figura 3.1, forma a base para projetos de redes neurais artificiais, em que se pode observar três elementos básicos:

1. a entrada, representada pelo produto interno de x_m e w_{km}. Assim, um sinal xj na entrada da sinapse j conectado ao neurônio k é multiplicado pelo peso sináptico w_{kj}. O primeiro índice se refere ao neurônio em questão e o segundo se refere ao terminal de entrada a qual o peso se refere;
2. um somador, que soma o produto interno dos sinais de entrada com seus respectivos pesos do neurônio;

3. uma função de ativação, ou de restrição, que restringe a amplitude da saída y_k do neurônio.

O modelo neural da Figura 3.1 possui também uma *bias* aplicado externamente, representado por b_k (ou w_0). O *bias* b_k tem o efeito de aumentar ou diminuir a entrada líquida da função de ativação, dependendo se ele é positivo ou negativo, respectivamente.

A Figura 3.4 mostra a arquitetura de uma rede *perceptron* de múltiplas camadas com uma camada de entrada, uma oculta e uma de saída, totalmente conectadas. Isso significa que um neurônio em qualquer camada da rede está conectado a todos os neurônios da camada anterior.

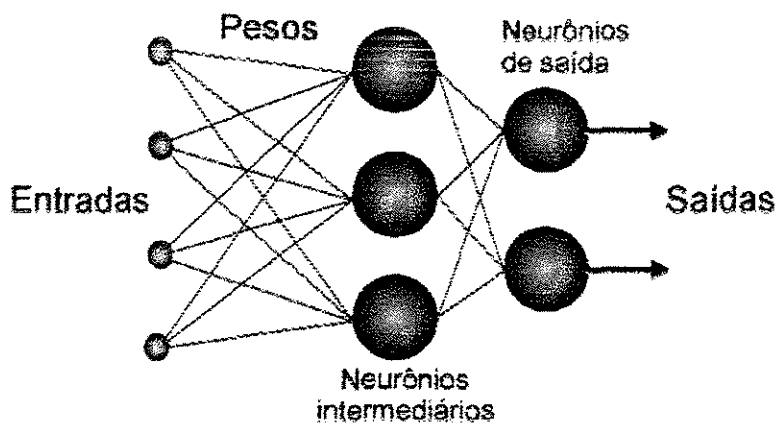


Figura 3.4: Arquitetura *perceptron* de múltiplas camadas com uma camada oculta

3.6.1 Função de Ativação

A função de ativação, representada por $f(\bullet)$ na Figura 3.1, define a saída (em geral não linear) de um neurônio, após o processamento da informação recebida pelo neurônio. Aqui são descritas apenas as três funções de ativação mais utilizadas na literatura.

3.6.1.1 *Função Linear*

Este tipo de função de ativação é muito utilizado nas unidades que compõe a camada de saída das arquiteturas MLP (SILVA, 1998).

A expressão para esta função de ativação é:

$$f(x) = p.x\,,\; f'(x) = p.x$$

3.6.1.2 *Função Sigmóide*

A origem deste tipo de função está vinculada à preocupação em limitar o intervalo de variação da função (0, 1), pela inclusão de um efeito de saturação. Sua derivada também é uma função contínua. As expressões são as seguintes:

$$f(x) = \frac{1}{1+e^{-px}}\,,\; f'(x) = p.f(x).(1-f(x))$$

3.6.1.3 *Função Tangente Hiperbólica*

Pelo fato da função sigmóide apresentar valores de ativação no intervalo (0, 1), em muitos casos ela é substituída pela função tangente hiperbólica, que preserva a forma sigmoidal da função sigmóide, mas assume valores positivos e negativos (-1, 1). A função tangente hiperbólica e sua derivada são dadas pelas expressões:

$$f(x) = \frac{e^{px} - e^{-px}}{e^{px} + e^{-px}} = \tanh(p.x),\; f'(x) = p.\left(1 - f(x)^2\right)$$

3.6.2 O Algoritmo de Retropropagação (*backpropagation*)

O *backpropagation* padrão é um algoritmo de gradiente descendente, no qual os pesos da rede são movidos ao longo do negativo do gradiente da função de desempenho. O termo *backpropagation* refere-se a maneira como o gradiente é computado para redes de múltiplas camadas não lineares. Existem diversas variações do algoritmo básico que são baseados em outras técnicas de otimização, tais como o gradiente conjugado e os métodos de Newton.

Durante o treinamento com o algoritmo *backpropagation*, a rede opera em uma seqüência de dois passos. Primeiro, um conjunto de variáveis é apresentado à camada de entrada da rede. A atividade flui através da rede, camada por camada, até que a resposta seja produzida pela camada de saída. No segundo passo, a saída fornecida pela rede é comparada com a saída desejada para esse conjunto particular. Se esta não estiver correta, o erro é calculado. O erro é propagado a partir da camada de saída para a camada de entrada, e os pesos das conexões das unidades das camadas internas vão sendo modificados conforme o erro retro-propagado (OLIVEIRA, 2000).

Para facilitar a derivação do algoritmo *backpropagation* (retro-propagação), será apresentado um resumo da notação utilizada.

Notação:

i, j índices referentes a diferentes neurônios da rede

n n-ésimo vetor de entrada (iteração)

N número de amostras (padrões de treinamento)

M número de camadas

$y_j(n)$ sinal de saída da unidade j na iteração n

$e_j(n)$ sinal de erro da unidade de saída j na iteração n

$w_{i,j}(n)$ peso sináptico conectando a saída da unidade i à entrada da unidade j na iteração n

$u_j(n)$ ativação da unidade j na iteração n; sinal a ser aplicado à não-linearidade

$f_j(.)$ função de ativação associada à unidade j

$\mathbf{X}$ matriz de dados de entrada (amostras de treinamento)

$\mathbf{S}$ matriz de dados de saída (saídas desejadas)

$x_i(n)$ i-ésimo elemento do vetor de entradas

$s_j(n)$ j-ésimo elemento do vetor de saídas

α taxa de aprendizagem

Todas as letras minúsculas em **negrito** (**a**, **b**, **c**) representam vetores, as letras maiúsculas em **negrito** (**A**, **B**, **C**) representam matrizes e as letras em *itálico* (*a*, *b*, *c*) representam escalares.

As matrizes $\mathbf{W}^m$ (para $m = 0, 1, \ldots, M - 1$; onde M é o número de camadas da rede) possuem dimensão $S^{m+1} \times S^m$, onde S^0 = número de entradas da rede; e os vetores $\mathbf{b}^m$ possuem dimensão $S^{m+1} \times 1$.

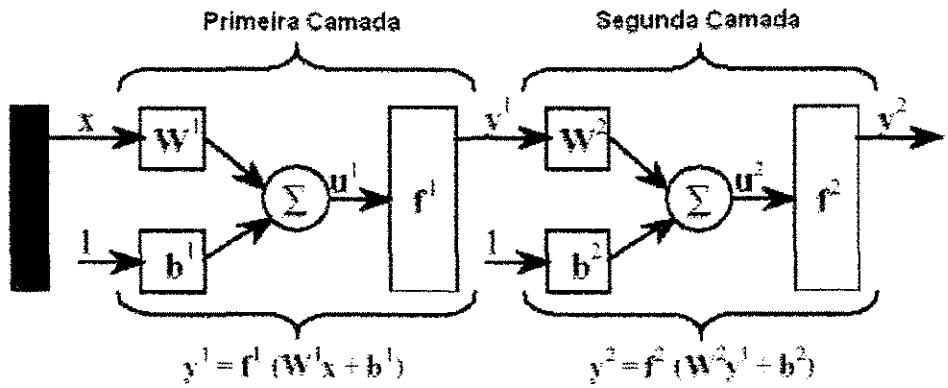


Figura 3.5: Rede neural com 2 camadas. Notação abreviada

Para simplificar o desenvolvimento do algoritmo *backpropagation*, utilizaremos a notação abreviada para uma arquitetura genérica com três camada (DEMULTH & BEALE, 2001; HAGAN *et al.*, 1997; NARENDRA e PARTHASARATHY, 1990) e tomaremos como base a Figura 3.5, na qual temos uma rede neural artificial com uma camada intermediária e uma camada de saída ($M = 2$); as unidades na primeira camada (camada oculta) recebem as entradas externas agrupadas em um vetor na forma:

$$\mathbf{y}^0 = \mathbf{x} \tag{3.1}$$

O vetor de saída da camada oculta da rede é dado por:

$$\mathbf{u}^1 = \mathbf{W}^1\mathbf{x} + \mathbf{b}^1 \tag{3.2}$$

$$\mathbf{y}^1 = \mathbf{f}^1\left(\mathbf{u}^1\right) = \mathbf{f}^1\left(\mathbf{W}^1\mathbf{x} + \mathbf{b}^1\right) \tag{3.3}$$

O vetor de saída da camada de saída da rede é dado por:

$$\mathbf{u}^2 = \mathbf{W}^2\mathbf{y}^1 + \mathbf{b}^2 \tag{3.4}$$

$$\mathbf{y}^2 = \mathbf{f}^2\left(\mathbf{u}^2\right) = \mathbf{f}^2\left(\mathbf{W}^2\mathbf{y}^1 + \mathbf{b}^2\right) \tag{3.5}$$

Logo, a saída da rede é dada em função do vetor de entrada **x**, das matrizes de pesos $\mathbf{W}^1$ e $\mathbf{W}^2$ e dos vetores de limiares $\mathbf{b}^1$ e $\mathbf{b}^2$. A expressão é:

$$\mathbf{y} = \mathbf{y}^2 = \mathbf{f}^2\left(\mathbf{W}^2\mathbf{f}^1\left(\mathbf{W}^1\mathbf{x} + \mathbf{b}^1\right) + \mathbf{b}^2\right) \tag{3.6}$$

Podemos representar as equações acima para uma forma geral onde possuímos um total de *M* camadas na rede. Assim:

$$\mathbf{u}^{m+1} = \mathbf{W}^{m+1}\mathbf{y}^m + \mathbf{b}^{m+1} \tag{3.7}$$

$$\mathbf{y}^{m+1} = \mathbf{f}^{m+1}\left(\mathbf{u}^{m+1}\right) = \mathbf{f}^{m+1}\left(\mathbf{W}^{m+1}\mathbf{y}^m + \mathbf{b}^{m+1}\right) \tag{3.8}$$

onde $m = 0, 1, \ldots, M\text{-}1$.

O algoritmo *backpropagation* para as redes de múltiplas camadas é uma generalização do método dos quadrados mínimos (LS – do inglês *Least Squares*) e utiliza como medida de desempenho o erro quadrático médio (**MSE** – do inglês *mean squared error*). Inicialmente, é apresentado um conjunto de exemplos:

$$\{(\mathbf{x}_1, \mathbf{s}_1), (\mathbf{x}_2, \mathbf{s}_2), \ldots, (\mathbf{x}_N, \mathbf{s}_N)\} \tag{3.9}$$

onde $\mathbf{x}_n$ é a *n*-ésima entrada para a rede e $\mathbf{s}_n$ a saída desejada correspondente ($n = 1, \ldots, N$). Das Equações (2.3) e (2.4) vemos que, se q é o vetor de parâmetros da rede (pesos e limiares), então o vetor de saídas da rede pode ser dado na forma:

$$\mathbf{y}^M = \mathbf{y}(\theta, \mathbf{x})$$

Após cada entrada ser aplicada à rede, a saída produzida pela rede é comparada com a saída desejada, **s**. O algoritmo deve ajustar os parâmetros da rede (pesos e limiares), com o objetivo de minimizar o erro quadrático médio na iteração n. Logo,

$$\mathbf{J}(n) = \mathbf{e}(n)^T \mathbf{e}(n) = (\mathbf{s}(n) - \mathbf{y}(n))^T (\mathbf{s}(n) - \mathbf{y}(n)) \tag{3.10}$$

onde $\mathbf{e}(n)$ é o erro $(\mathbf{s}(n) - \mathbf{y}(n))$ na iteração n.

A lei de ajuste conhecida como *steepest descent* para minimizar o erro quadrático é dada por:

$$w_{i,j}^m(n+1) = w_{i,j}^m(n) - \alpha \frac{\partial J(n)}{\partial w_{i,j}^m} \tag{3.11}$$

$$b_i^m(n+1) = b_i^m(n) - \alpha \frac{\partial J(n)}{\partial b_i^m} \tag{3.12}$$

onde α é a taxa de aprendizagem.

Como o erro é função indireta dos pesos nas camadas intermediárias, a regra da cadeia deverá ser usada para o cálculo das derivadas. O conceito da regra da cadeia será utilizado na determinação das derivadas das Equações (3.11) e (3.12):

$$\frac{\partial J}{\partial w_{i,j}^m} = \frac{\partial J}{\partial u_i^m} \frac{\partial u_i^m}{\partial w_{i,j}^m} \tag{3.13}$$

$$\frac{\partial J}{\partial b_i^m} = \frac{\partial J}{\partial u_i^m} \frac{\partial u_i^m}{\partial b_i^m} \tag{3.14}$$

Porém, $u_i^m = \sum_{j=1}^{S^{m-1}} w_{i,j}^m y_j^{m-1} + b_i^m$, logo $\frac{\partial u_i^m}{\partial w_{i,j}^m} = y_j^{m-1}$, $\frac{\partial u_i^m}{\partial b_i^m} = 1$.

Definindo agora a sensibilidade de J a mudanças no i-ésimo elemento da ativação da rede na camada m como:

$$\delta_i^m = \frac{\partial J}{\partial u_i^m} \tag{3.15}$$

as Equações (3.13) e (3.14) podem ser simplificadas por

$$\frac{\partial J}{\partial w_{i,j}^m} = \delta_i^m y_j^{m-1} \tag{3.16}$$

$$\frac{\partial J}{\partial b_i^m} = \delta_i^m \tag{3.17}$$

Agora é possível aproximar as Equações (3.11) e (3.12) por

$$w_{i,j}^m(n+1) = w_{i,j}^m(n) - \alpha \delta_i^m y_j^{m-1} \tag{3.18}$$

$$b_i^m(n+1) = b_i^m(n) - \alpha \delta_i^m \tag{3.19}$$

Em notação matricial, as equações acima tornam-se:

$$\mathbf{W}^m(n+1) = \mathbf{W}^m(n) - \boldsymbol{\alpha}\boldsymbol{\delta}^m\left(\mathbf{y}^{m-1}\right) \tag{3.20}$$

$$\mathbf{b}^m(n+1) = \mathbf{b}^m(n) - \boldsymbol{\alpha}\boldsymbol{\delta}^m \tag{3.21}$$

onde

$$\boldsymbol{\delta}^m = \frac{\partial J}{\partial \mathbf{u}^m} = \begin{bmatrix} \dfrac{\partial J}{\partial u_1^m} \\ \dfrac{\partial J}{\partial u_2^m} \\ \mathrm{M} \\ \dfrac{\partial J}{\partial u_{S^m}^m} \end{bmatrix} \tag{3.22}$$

Ainda é necessário calcular as sensibilidades $\boldsymbol{\delta}^m$, que requer outra aplicação da regra da cadeia. É este processo que dá origem ao termo retro-propagação (*backpropagation*), pois descreve a relação de recorrência na qual a sensibilidade na camada m é calculada a partir da sensibilidade na camada $m + 1$.

Para derivar a relação de recorrência das sensibilidades, utilizaremos a seguinte matriz jacobiana:

$$\frac{\partial \mathbf{u}^{m+1}}{\partial \mathbf{u}^m} = \begin{bmatrix} \dfrac{\partial u_1^{m+1}}{\partial u_1^m} & \dfrac{\partial u_1^{m+1}}{\partial u_2^m} & \mathrm{K} & \dfrac{\partial u_1^{m+1}}{\partial u_{S^m}^m} \\ \dfrac{\partial u_2^{m+1}}{\partial u_1^m} & \dfrac{\partial u_2^{m+1}}{\partial u_2^m} & \mathrm{K} & \dfrac{\partial u_2^{m+1}}{\partial u_{S^m}^m} \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ \dfrac{\partial u_{S^{m+1}}^{m+1}}{\partial u_1^m} & \dfrac{\partial u_{S^{m+1}}^{m+1}}{\partial u_2^m} & \mathrm{K} & \dfrac{\partial u_{S^{m+1}}^{m+1}}{\partial u_{S^m}^m} \end{bmatrix} \tag{3.23}$$

Em seguida desejamos encontrar uma expressão para esta matriz. Considere o elemento i, j da matriz:

$$\frac{\partial u_i^{m+1}}{\partial u_j^m} = w_{i,j}^{m+1} \frac{\partial y_j^m}{\partial u_j^m} = w_{i,j}^{m+1} \frac{\partial f^m\left(u_j^m\right)}{\partial u_j^m} w_{i,j}^{m+1} \dot{f}^m\left(u_j^m\right) \tag{3.24}$$

onde

$$\dot{f}^{m}\left(u_j^m\right)=\frac{\partial f^m\left(u_j^m\right)}{\partial u_j^m} \tag{3.25}$$

Entretanto a matriz jacobiana pode ser escrita como:

$$\frac{\partial \mathbf{u}^{m+1}}{\partial \mathbf{u}^m}=\mathbf{W}^{m+1}\,\dot{\mathbf{F}}^{m}\left(\mathbf{u}^m\right) \tag{3.26}$$

onde

$$\dot{\mathbf{F}}^{m}\left(\mathbf{u}^m\right)=\begin{bmatrix} \dot{f}^{m}\left(u_1^m\right) & 0 & \Lambda & 0 \\ 0 & \dot{f}^{m}\left(u_2^m\right) & \Lambda & 0 \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ 0 & 0 & \Lambda & \dot{f}^{m}\left(u_{S^m}^m\right) \end{bmatrix} \tag{3.27}$$

Agora podemos escrever a relação de recorrência para a sensibilidade utilizando a regra da cadeia em forma matricial:

$$\boldsymbol{\delta}^m=\frac{\partial J}{\partial \mathbf{u}^m}=\left(\frac{\partial \mathbf{u}^{m+1}}{\partial \mathbf{u}^m}\right)^T\frac{\partial J}{\partial \mathbf{u}^{m+1}}=\dot{\mathbf{F}}^{m}\left(\mathbf{u}^m\right)\left(\mathbf{W}^{m+1}\right)^T\frac{\partial J}{\partial \mathbf{u}^{m+1}}$$

$$\boldsymbol{\delta}^m=\dot{\mathbf{F}}^{m}\left(\mathbf{u}^m\right)\left(\mathbf{W}^{m+1}\right)^T\boldsymbol{\delta}^{m+1} \tag{3.28}$$

Observe que as sensibilidades são propagadas da última para a primeira camada através da rede:

$$\delta^M\rightarrow\delta^{M-1}\rightarrow\Lambda\rightarrow\delta^2\rightarrow\delta^1 \tag{3.29}$$

Ainda existe um último passo a ser executado para que o algoritmo de retro-propagação fique completo. Precisamos do ponto de partida, δ^M, para a relação de recorrência da Equação (3.28). Este ponto é obtido na última camada:

$$\delta_i^M = \frac{\partial J}{\partial u_i^M} = \frac{\partial(\mathbf{s}-\mathbf{y})^T(\mathbf{s}-\mathbf{y})}{\partial u_i^M} = -2(s_i - y_i)\frac{\partial y_i}{\partial u_i^M} \tag{3.30}$$

Como

$$\frac{\partial y_i}{\partial u_i^M} = \frac{\partial y_i^M}{\partial u_i^M} = \dot{f}^M\left(u_j^M\right) \tag{3.31}$$

Podemos escrever

$$\delta_i^M = -2(s_i - y_i)\dot{f}^M\left(u_j^M\right) \tag{3.32}$$

A expressão acima pode ser colocada em forma matricial, resultando

$$\delta^M = -2\dot{\mathbf{F}}^M\left(\mathbf{u}^M\right)(\mathbf{s}-\mathbf{y}) \tag{3.33}$$

3.6.3 Algoritmos de otimização para treinamento supervisionado

Nesta parte são descritos três métodos de otimização não-linear irrestrita para treinamento de redes multicamadas. O treinamento de redes neurais com várias camadas pode ser entendido como um caso especial de aproximação de funções, onde não é levado em consideração nenhum modelo explícito dos dados (SHEPHERD, 1997). Serão apresentados os seguintes algoritmos:

- Levenberg-Marquardt (LM);
- Broyden-Fletcher-Goldfarb-Shanno (BFGS);
- Gradiente conjugado escalonado (SCG).

A escolha dos métodos acima foi feita com base na capacidade que os mesmos possuem de conseguir convergências mais rápidas que os demais algoritmos nas mais variadas aplicações, como reconhecimento de padrões e em problemas de aproximação de funções. O desenvolvimento dos métodos a seguir foram extraídos de DEMULTH e BEALE, 2002; HAGAN *et al.*, 1997; EDGAR *et al.* 2001; CHAPRA e CANALE, 2002 e principalmente SILVA, 1998.

3.6.3.1 Método de Levenberg-Marquardt (LM)

Este método é bastante eficiente quando estamos tratando de redes que não possuem mais do que algumas centenas de conexões a serem ajustadas (HAGAN, 1994). Isto deve-se, principalmente, ao fato de que estes algoritmos necessitam armazenar uma matriz quadrada cuja dimensão é da ordem do número de conexões da rede (SILVA, 1998).

Se considerarmos como funcional de erro a soma dos erros quadráticos, e ainda levarmos em conta que o problema pode ter múltiplas saídas, obtemos a seguinte expressão para o funcional de erro:

$$J(\theta)=\sum_{i=1}^{N}\sum_{j=1}^{m}\left(g_{ij}(\mathbf{x})-\hat{g}_{ij}(\mathbf{x},\theta)\right)^2=\sum_{k=1}^{q}r_l^2 \tag{3.34}$$

onde $J(\theta)$ é o funcional de erro, $\hat{g}_{ij}(\mathbf{x},\theta)$ é o modelo que procura aproximar a função $g_{ij}(\mathbf{x})$, N é o número de amostras, l o número de unidades intermediárias, r o erro residual, m o número de saídas, e q o produto $N \times m$.

Seja **J** o Jacobiano (matriz das derivadas primeiras) do funcional J dado pela Equação (3.14). Esta matriz pode ser escrita da seguinte forma:

$$\mathbf{J} \equiv \begin{bmatrix} \nabla r_1^T \\ \nabla r_2^T \\ \mathrm{M} \\ \nabla r_q^T \end{bmatrix} \tag{3.35}$$

onde r é denominado erro residual.

Diferenciando a Equação (3.34) obtemos:

$$\nabla J = 2\mathbf{J}^T\mathbf{r} = 2\sum_{k=1}^{q} r_k \nabla \mathbf{r}_k \tag{3.36}$$

$$\nabla^2 J = 2\left(\mathbf{J}^T\mathbf{J} + \sum_{k=1}^{q} r_k \nabla^2 \mathbf{r}_k \right) \tag{3.37}$$

A matriz de derivadas segundas do funcional de erro é chamada de matriz hessiana. Quando os erros residuais são suficientemente pequenos, a matriz hessiana pode ser aproximada pelo primeiro termo da Equação (3.37), resultando em:

$$\nabla^2 J \approx 2\mathbf{J}^T\mathbf{J} \tag{3.38}$$

Esta aproximação geralmente é válida em um mínimo de J para a maioria dos propósitos, e é a base para o método de Gauss-Newton (HAGAN, 1994). A lei de atualização torna-se então:

$$\Delta\boldsymbol{\theta} = \left[\mathbf{J}^T\mathbf{J}\right]^{-1}\mathbf{J}^T\mathbf{r} \tag{3.39}$$

A modificação de Levenberg-Marquardt para o método de Gauss-Newton é:

$$\Delta\boldsymbol{\theta} = \left[\mathbf{J}^T\mathbf{J} + \mu\mathbf{I}\right]^{-1}\mathbf{J}^T\mathbf{r} \tag{3.40}$$

O efeito da matriz adicional $\mu\mathbf{I}$ é adicionar μ a cada autovalor de $\mathbf{J}^T\mathbf{J}$. Uma vez que a matriz $\mathbf{J}^T\mathbf{J}$ é semidefinida positiva e, portanto o autovalor mínimo possível é zero, qualquer valor positivo, pequeno, mas numericamente significativo, de μ será suficiente para restaurar a matriz aumentada e produzir uma direção descendente de busca.

Os valores de μ podem ser escolhidos de várias maneiras; a mais simples é escolhê-lo zero ao menos que a matriz hessiana encontrada na iteração *i* seja singular. Quando isso ocorrer, um valor pequeno como $\boldsymbol{\mu} = 10^{-4}\sum_1^N \left(\mathbf{J}^T\mathbf{J}\right)_{ii}$ pode ser usado. Outras formas de determinação do parâmetro μ são sugeridas por HAGAN (1994), MCKEON *et al.* (1997) e FINSCHI (1996).

É importante observar que, quanto maior for o valor de μ, menor é a influência da informação de segunda ordem e mais este algoritmo se aproxima de um método de primeira ordem.

3.6.3.2 – Método de Broyden-Fletcher-Goldfarb-Shanno (BFGS)

Este método é classificado como método quase-Newton. A idéia por trás dos métodos quase-Newton é fazer uma aproximação iterativa da inversa da matriz hessiana, de forma que:

$$\lim_{i\to\infty} \mathbf{H}_i = \nabla^2 \mathbf{J}(\boldsymbol{\theta})^{-1} \tag{3.41}$$

São considerados os métodos teoricamente mais sofisticados na solução de problemas de otimização não-linear irrestrita e representam o ápice do desenvolvimento de algoritmos através de uma análise detalhada de problemas quadráticos.

A cada passo a inversa da hessiana é aproximada pela soma de duas matrizes simétricas de posto 1, procedimento que é geralmente chamado de correção de posto 2 (*rank 2 correction procedure*).

3.6.3.2.1 Construção da Inversa

A idéia é construir uma aproximação da inversa da matriz hessiana, utilizando informações de primeira ordem obtidas durante o processo iterativo de aprendizagem. A aproximação atual $\mathbf{H}_i$ é utilizada a cada iteração para definir a próxima direção descendente do método. Idealmente, as aproximações convergem para a inversa da matriz hessiana.

Suponha que o funcional de erro $J(\theta)$ tenha derivada parcial contínua até segunda ordem. Tomando dois pontos θ_i e θ_{i+1}, defina $\mathbf{g}_i = -\nabla J(\theta_i)^T$ e $\mathbf{g}_{i+1} = -\nabla J(\theta_{i+1})^T$. Se a hessiana, $\nabla^2\mathbf{J}(\theta)$, é constante, então temos:

$$\mathbf{q}_i \equiv \mathbf{g}_{i+1} - \mathbf{g}_i = \nabla^2 J(\theta)\mathbf{p}_i \tag{3.42}$$

$$\mathbf{p}_i \equiv \alpha \mathbf{d}_i \tag{3.43}$$

Vemos então que a avaliação do gradiente em dois pontos fornece informações sobre a matriz hessiana $\nabla^2\mathbf{J}(\theta)$. Tomando-se P direções linearmente independentes $\{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_{P-1}\}$, é possível determinar unicamente $\nabla^2\mathbf{J}(\theta)$, caso se conheça $\mathbf{q}_i$, $i = 0, 1, \ldots, P-1$. Para tanto, basta aplicar iterativamente a Equação (3.44) a seguir, com $\mathbf{H}_0 = \mathbf{I}_P$ (matriz identidade de dimensão P).

$$\mathbf{H}_{i+1} = \mathbf{H}_i + \frac{\mathbf{p}_i\mathbf{p}_i^T}{\mathbf{p}_i^T\mathbf{q}_i}\left[1 + \frac{\mathbf{q}_i^T\mathbf{H}_i\mathbf{q}_i}{\mathbf{p}_i^T\mathbf{q}_i}\right] - \frac{\mathbf{H}_i\mathbf{q}_i\mathbf{p}_i^T + \mathbf{p}_i\mathbf{q}_i^T\mathbf{H}_i}{\mathbf{p}_i^T\mathbf{q}_i} \tag{3.44}$$

para $i = 0, 1, \ldots, P\text{-}1$.

Após P iterações sucessivas, se $J(\theta)$ for uma função quadrática, então $\mathbf{H}_P = \nabla^2\mathbf{J}(\boldsymbol{\theta})^{-1}$. Como geralmente não estamos tratando de problemas quadráticos, a cada P iterações é recomendável que se faça uma reinicialização do algoritmo, ou seja, a direção de minimização é oposta àquela dada pelo vetor gradiente e $\mathbf{H} = \mathbf{I}_P$.

3.6.3.3 – Método do Gradiente conjugado escalonado (SCG)

Existe um consenso geral da comunidade de análise numérica que a classe de métodos de otimização chamados métodos do gradiente conjugado tratam de problemas de grande escala de maneira efetiva (VAN DER SMAGT, 1994).

Os métodos do gradiente conjugado possuem sua estratégia baseada no modelo geral de otimização apresentado no algoritmo padrão e do gradiente, mas escolhem a direção de busca $\mathbf{d}_i$, o passo α_i e o coeficiente de momento β_i (Equação (3.45)) mais eficientemente utilizando informações de segunda ordem. É projetado para exigir menos cálculos que o método de Newton e apresentar taxas de convergência maiores que as do método do gradiente.

$$\boldsymbol{\theta}_{i+1} \equiv \boldsymbol{\theta}_i + \boldsymbol{\alpha}_i \mathbf{d}_i + \boldsymbol{\beta}_i \Delta\boldsymbol{\theta}_{i-1} \tag{3.45}$$

Antes de apresentar o método do gradiente conjugado é necessário apresentar um resultado intermediário denominado método das direções conjugadas.

3.6.3.3.1 Métodos das Direções Conjugadas

A lei de adaptação dos processos em estudo assume a forma da Equação (3.46):

$$\boldsymbol{\theta}_{i+1} \equiv \boldsymbol{\theta}_i + \boldsymbol{\alpha}_i \mathbf{d}_i, \qquad i \geq 0 \tag{3.46}$$

onde $\theta_i \in \Re^P$ é o vetor de parâmetros, $\alpha_i \in \Re^+$ é um escalar que define o passo de ajuste e $\mathbf{d}_i \in \Re^P$ é a direção de ajuste, todos definidos na iteração i. Havendo convergência, a solução ótima $\theta^* \in \Re^P$ pode ser expressa na forma:

$$\boldsymbol{\theta} \equiv \boldsymbol{\alpha}_0 \mathbf{d}_0 + \boldsymbol{\alpha}_1 \mathbf{d}_1 + \ldots = \sum_i \boldsymbol{\alpha}_i \mathbf{d}_i \tag{3.47}$$

Assumindo por hipótese que o conjunto $\{\mathbf{d}_0, \mathbf{d}_1, \ldots, \mathbf{d}_{P-1}\}$ forma uma base de $\Re^P$ e $\alpha = (\alpha_0 \ \ldots \ \alpha_{P-1})^T$ é a representação de θ^* nesta base, então é possível obter θ^* em P iterações da Equação (3.45) na forma:

$$\boldsymbol{\theta} \equiv \boldsymbol{\alpha}_0\mathbf{d}_0 + \boldsymbol{\alpha}_1\mathbf{d}_1 + \ldots\boldsymbol{\alpha}_{P-1}\mathbf{d}_{P-1} = \sum_{i=0}^{P-1} \boldsymbol{\alpha}_i\mathbf{d}_i \tag{3.48}$$

Dada uma matriz **A** simétrica de dimensão $P \times P$, as direções $\mathbf{d}_i \in \Re^P$, $i = 0, \ldots, P\text{-}1$, são ditas serem **A**-conjugadas ou **A**-ortogonais se: $\mathbf{d}_j^T\mathbf{A}\mathbf{d}_i = 0$, para $i \neq j$ e $i,j = 0, \ldots, P\text{-}1$.

Se a matriz **A** for também definida positiva, o conjunto de P direções **A**-conjugadas forma uma base do $\Re^P$. Dessa forma, os coeficientes $\boldsymbol{\alpha}_j^*$, $j = 1,\ldots, P\text{-}1$ podem ser determinados pelo procedimento descrito abaixo.

Dada uma matriz **A** simétrica, definida positiva e de dimensão $P \times P$, multiplicando-se a Equação (3.48) à esquerda por $\mathbf{d}_j^T\mathbf{A}\mathbf{d}_i$, com $0 \leq j \leq P\text{-}1$, resulta:

$$\mathbf{d}_j^T\mathbf{A}\boldsymbol{\theta}^* = \sum_{i=0}^{P-1} \boldsymbol{\alpha}_i^*\mathbf{d}_j^T\mathbf{A}\mathbf{d}_i, \qquad j = 1,\ldots, P\text{-}1 \tag{3.49}$$

Escolhendo as direções $\mathbf{d}_i \in \Re^P$, como sendo **A**-conjugadas, é possível aplicar os resultados apresentados acima para obter:

$$\alpha_j^* = \frac{\mathbf{d}_j^T\mathbf{A}\boldsymbol{\theta}^*}{\mathbf{d}_j^T\mathbf{A}\mathbf{d}_j}, \qquad j = 1,\ldots, P\text{-}1 \tag{3.50}$$

É necessário eliminarmos θ^* da expressão (3.50), e para que isso seja feito são necessárias duas hipóteses adicionais:

- Suponha que o problema é quadrático, ou seja: $J(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta}^T\mathbf{Q}\boldsymbol{\theta} - \mathbf{b}^T\boldsymbol{\theta}$

Então no ponto de mínimo θ^*, é válida a expressão:

$$\nabla J(\boldsymbol{\theta}^*) = 0 \Rightarrow \mathbf{Q}\boldsymbol{\theta}^* - \mathbf{b} = 0 \Rightarrow \mathbf{Q}\boldsymbol{\theta}^* = \mathbf{b} \tag{3.51}$$

• Suponha que $\mathbf{A} = \mathbf{Q}$.

Sendo assim, a Equação (3.50) resulta em:

$$\alpha_j^* = \frac{\mathbf{d}_j^T \mathbf{b}}{\mathbf{d}_j^T \mathbf{A}\mathbf{d}_j}, \qquad j = 1,..., P-1 \tag{3.52}$$

e o ponto de mínimo θ* é dado por:

$$\boldsymbol{\theta}^* = \sum_{j=0}^{P-1} \frac{\mathbf{d}_j^T \mathbf{b}}{\mathbf{d}_j^T \mathbf{A}\mathbf{d}_j} \mathbf{d}_j \tag{3.53}$$

Assumindo solução iterativa com θ* expresso na forma da Equação (3.53), os coeficientes $\alpha_j^* = 1,..., P-1$, são dados por:

$$\boldsymbol{\theta} = \boldsymbol{\theta}_0 + \boldsymbol{\alpha}_0^* \mathbf{d}_1 + \boldsymbol{\alpha}_1^* \mathbf{d}_1 + ... \boldsymbol{\alpha}_{P-1}^* \mathbf{d}_{P-1} \tag{3.54}$$

$$\alpha_j^* = \frac{\mathbf{d}_j^T \mathbf{Q}(\theta^* - \theta_0)}{\mathbf{d}_j^T \mathbf{Q}\mathbf{d}_j}, \qquad j = 1,..., P-1 \tag{3.55}$$

Na j-ésima iteração, e levando-se em conta a Equação (3.54), obtém-se:

$$\alpha_j^* = \frac{\mathbf{d}_j^T \nabla J(\theta_j)}{\mathbf{d}_j^T \mathbf{Q}\mathbf{d}_j}, \qquad j = 1,..., P-1 \tag{3.56}$$

e a lei de ajuste do método das direções conjugadas é dada por:

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - \frac{\mathbf{d}_i^T \nabla J(\boldsymbol{\theta})}{\mathbf{d}_i^T \mathbf{Q} \mathbf{d}_i} \mathbf{d}_i \tag{3.57}$$

3.6.3.3.2 O Método do Gradiente Conjugado

Antes de aplicarmos a lei de ajuste dada pela Equação (3.57), é necessário obter as direções **Q**-conjugadas $\mathbf{d}_i \in \Re^P$, $i = 0, \ldots, P\text{-}1$. Uma maneira de determinar estas direções é tomá-las na forma (BAZARAA *et. al.*, 1993):

$$\begin{cases} \mathbf{d}_0 = -\nabla J(\boldsymbol{\theta}_0) \\ \mathbf{d}_{i+1} = -\nabla J(\boldsymbol{\theta}_{i+1}) + \boldsymbol{\beta}_i \mathbf{d}_i \quad i \geq 0 \end{cases} \tag{3.58}$$

$$\text{com} \quad \boldsymbol{\beta}_i = \frac{\nabla J(\boldsymbol{\theta}_{i+1})^T \mathbf{Q} \mathbf{d}_i}{\mathbf{d}_i^T \mathbf{Q} \mathbf{d}_i}$$

3.6.3.3.3 Gradiente Conjugado Escalonado

MOLLER (1993) introduziu uma nova variação no algoritmo de gradiente conjugado (Gradiente Conjugado Escalonado – SCG), que evita a busca unidimensional a cada iteração utilizando uma abordagem de Levenberg-Marquardt cujo objetivo é fazer um escalonamento do passo de ajuste α.

Quando não estamos tratando com problemas quadráticos, a matriz **Q** deve ser aproximada pela matriz Hessiana calculada no ponto θ_i, e a Equação (3.56) torna-se:

$$\alpha_j^* = -\frac{\mathbf{d}_j^T \nabla J(\boldsymbol{\theta}_j)}{\mathbf{d}_j^T \nabla^2 J(\boldsymbol{\theta}_j) \mathbf{d}_j} \tag{3.59}$$

A idéia utilizada por Moller é estimar o termo denominado $\mathbf{s}_j = \nabla^2 J(\boldsymbol{\theta}_j)\mathbf{d}_j$ do método do gradiente conjugado por uma aproximação da forma:

$$\mathbf{s}_j = \nabla^2 J(\boldsymbol{\theta}_j)\mathbf{d}_j \approx \frac{\nabla J(\boldsymbol{\theta}_j + \boldsymbol{\sigma}_j \mathbf{d}_j) - \nabla J(\boldsymbol{\theta}_j)}{\boldsymbol{\sigma}_j}, \quad 0 < \boldsymbol{\sigma}_j << 1 \tag{3.60}$$

A aproximação tende, no limite, ao valor de $\nabla^2 J(\boldsymbol{\theta}_j)\mathbf{d}_j$.Combinando esta estratégia com a abordagem do gradiente conjugado e Levenberg-Marquardt, obtém-se um algoritmo diretamente aplicável ao treinamento de redes MLP. Isso é feito da seguinte maneira:

$$\mathbf{s}_j = \frac{\nabla J(\boldsymbol{\theta}_j + \boldsymbol{\sigma}_j \mathbf{d}_j) - \nabla J(\boldsymbol{\theta}_j)}{\boldsymbol{\sigma}_j} + \boldsymbol{\lambda}_j \mathbf{d}_j \tag{3.61}$$

Seja δ_j o denominador da equação (3.59); então, utilizando a expressão (3.60), resulta:

$$\boldsymbol{\delta}_j = \mathbf{d}_j^T \mathbf{s}_j \tag{3.62}$$

O ajuste do parâmetro λ_j a cada iteração e a análise do sinal de δ_j permitem determinar se a hessiana é definida positiva ou não.

A aproximação quadrática $J_{quad}(\boldsymbol{\theta})$, utilizada pelo algoritmo, nem sempre representa uma boa aproximação para $J(\theta)$, uma vez que λ_j escalona a matriz hessiana de maneira artificial. Um mecanismo para aumentar e diminuir λ_j é necessário para fornecer uma boa aproximação, mesmo quando a matriz for definida positiva. Define-se:

$$\boldsymbol{\Delta}_j = \frac{J(\boldsymbol{\theta}_j) - J(\boldsymbol{\theta}_j + \boldsymbol{\alpha}_j \mathbf{d}_j)}{J(\boldsymbol{\theta}_j) - J_{quad}(\boldsymbol{\alpha}_j \mathbf{d}_j)}$$

$$\Delta_j = \frac{2\delta_j \left[J(\theta_j) - J(\theta_j + \alpha_j \mathbf{d}_j) \right]}{\mu_j^2} \tag{3.63}$$

onde $\mu_j = -\mathbf{d}_j^T \nabla J(\theta_j)$.

O termo Δ_j representa uma medida da qualidade da aproximação $J_{quad}(\theta)$ em relação a $J(\theta_j + \alpha_j \mathbf{d}_j)$ no sentido de que quanto mais próximo de 1 estiver Δ_j, melhor é a aproximação.

3.7 Definindo a rede MLP para o estudo da modelagem do FCC

A seguir são descritos os procedimentos tomados para o treinamento da rede, tais como: modo de treinamento, topologia da rede, divisão dos dados e avaliação da eficiência do treinamento. Como já descrito anteriormente, foram utilizadas três diferentes metodologias para treinamento da rede MLP; em todas as metodologias foi utilizada a seguinte configuração com as funções de ativação: função tangente hiperbólica na camada oculta e função linear na camada de saída. O objetivo é obter uma rede neural com o melhor desempenho ou o menor erro quadrático médio de previsão das amostras de validação. No Capítulo 5 encontram-se os resultados obtidos para as diferentes metodologias abordadas.

3.7.1 Modo de Treinamento

O programa usado para treinar e testar as redes neurais foi o *Neural Networks Toolbox* para uso com *Matlab*. Neste *Toolbox* os algoritmos foram desenvolvidos para treinamento em lote. Segundo DEMUTH e BEALE (2001), existem duas diferentes formas de treinamento para o algoritmo *backpropagation*: o treinamento na forma seqüencial e o treinamento na forma de lote. Na forma seqüencial, a atualização dos pesos é realizada após a apresentação de cada exemplo (ou amostra) de treinamento. Já na forma em lote, todos os exemplos (ou amostras) são aplicados a rede antes da atualização dos pesos (uma

apresentação completa de todo o conjunto de treinamento é denominada de época). Para saber mais sobre as vantagens e desvantagens dos modos de treinamento consulte HAYKIN (2001).

3.7.2 Número de Camadas e de neurônios (topologia da rede)

A literatura indica que muitos dos problemas estudados com o uso de redes neurais artificiais têm sido resolvido utilizando apenas uma, e algumas vezes, duas camadas ocultas (KONDERLA e MOKANEK, 2000; SWINGLER, 1996). Neste trabalho, utilizaram-se redes contendo uma única camada oculta. Segundo OLIVEIRA (2000), o uso de uma única camada interna tem se mostrado suficiente na modelagem de processos químicos, visto que quando há necessidade de modelos mais complexos o ajuste do número de neurônios na camada oculta geralmente é suficiente.

O número de neurônios na camada de entrada é, em geral, igual ao número de variáveis de entrada do processo. Entretanto, este número pode ser reduzido através do uso de técnicas estatística de redução de variáveis como a análise dos componentes principais (PCA) e mínimos quadrados parciais (PLS), as quais serão descritas no Capítulo 4.

Para o número de neurônios na camada oculta não há ainda uma regra que indique o número necessário para se obter resultados satisfatórios no treinamento da rede. TURNER *et al.* (1996) apresentam algumas observações gerais para determinação da topologia da rede consistindo em:

1. A rede deve ter a estrutura mais simples possível, para evitar sobre-parametrização;
2. Pode ser demonstrado que qualquer função contínua não linear pode ser modelada utilizando uma camada oculta;
3. O número de neurônios na camada oculta deve ser inicialmente igual ao número de entradas. Do ponto de vista prático, este procedimento funciona de maneira satisfatória e tende a manter um número relativamente pequeno de pesos necessários para a rede. Se a rede falhar para modelar as relações de entrada e saída, o número de neurônios na camada oculta pode ser aumentado.

Da mesma forma que na camada de entrada, o número de neurônios na camada de saída é igual ao número de variáveis de saídas (variáveis a serem preditas) do processo.

Segundo OLIVEIRA-ESQUERRE (2003), é recomendável que cada modelo apresente uma única resposta (um neurônio) na camada de saída, o que diminui o número de parâmetros a serem ajustados e conseqüentemente a carga computacional exigida. Uma exceção a esta regra é para situações onde se deseja predizer diversas respostas correlacionadas, como as concentrações de diferentes constituintes de uma mistura em um sistema fechado. No presente estudo, foram utilizados dois modelos de redes neurais: 1) um modelo que forneça todas as variáveis de saída a partir das entradas fornecida e; 2) um modelo de RNA para cada saída da rede.

3.7.3 Amostras: divisão para treinamento, validação e teste

Em geral, o número de amostra (conjunto de dados) disponível é geralmente imposto ou limitado em problemas práticos. A literatura mostra que para modelagem de FCC via redes neurais não há trabalhos onde se encontra um considerável conjunto de dados (MICHALOPOULOS *et al.*, 2001; BOLLAS *et al.*, 2003; MCGREAVY *et al.*, 1994). Segundo OLIVEIRA-ESQUERRE (2003), é possível obter excelentes resultados para a modelagem de sistemas utilizando um número limitado de dados durante o treinamento. Entretanto, se for tentado validar o modelo para um conjunto independente de dados, geralmente, uma significativa degradação dos resultados será observada devido ao sobre-ajuste (ou overfitting) dos parâmetros e conseqüentemente perda da habilidade de generalização.

Dentro desse contexto, um importante passo no desenvolvimento de um modelo está na divisão do conjunto de dados disponíveis em dois ou três subconjunto:

1. Treinamento – utilizado para estimar os parâmetros do modelo;
2. Validação – utilizada para verificar a habilidade de generalização do modelo frente a amostras independentes do conjunto de treinamento;
3. Teste – utilizado para validar o modelo usando novas amostras.

Dependendo da quantidade de dados disponível, pode-se ter apenas o conjunto de dados de treinamento e validação. No presente estudo, tem-se uma quantidade de dados reduzida (91 amostras) e, assim sendo, será dividido nesse dois conjuntos: 75 % para treinamento e 25% para validação (BORGES, 2001).

De acordo com DESPAGNE e MASSART (1998), o ideal para um conjunto de dados considerado grande é dividir este conjunto em 40% das amostras para treinamento, 20 % para validação e 40% para teste A performance da rede não deve ser julgada pelo ajuste dos dados de treino, pois estes podem ser ajustados perfeitamente. Os resultados podem ser apresentados tanto pelo conjunto de validação como pelo conjunto de teste. Outros autores (HAYKIN, 2001; PRECHELT, 1997) sugerem que o conjunto seja dividido da seguinte forma: 50% dos dados para treinamento, 25% para validação e 25% para teste.

No processo de aprendizagem e validação deve-se observar com atenção a escolha dos conjuntos de dados, pois a rede deve ser treinada sobre o mais amplo domínio possível, de forma que o conjunto de validação esteja contido no conjunto de aprendizagem. Uma das limitações das redes reside na dificuldade de extrapolar dados para os quais não foi treinada.

3.7.4 Generalização

Quando a rede é treinada para atingir um erro mínimo esta, na maioria dos casos, é incapaz de predizer bem amostras que não foram usadas no conjunto de treinamento. A este fato é dado o nome de sobre-ajuste (*overfitting*) pois a rede se especializou nos dados de treinamento e perdeu sua capacidade de generalizar para novas situações. A seguir são apresentados dois métodos para melhorar a generalização dos dados: a parada antecipada e a regularização.

3.7.4.1 Parada Antecipada

Quando é feito o treinamento de uma rede neural, geralmente deseja-se obter uma rede com a melhor capacidade de generalização possível, ou seja, a maior capacidade de responder corretamente a dados que não foram utilizados no processo de treinamento. As arquiteturas convencionais, totalmente conectadas, como o MLP, estão sujeitas a sofrerem sobre-treinamento (*overtraining*): quando a rede parece estar representando o problema cada vez melhor, ou seja, o erro do conjunto de treinamento continua diminuindo, em

algum ponto deste processo a capacidade de responder a um novo conjunto de dados piora. Para combater o sobre-treinamento pode-se utilizar os procedimentos de parada antecipada que são largamente utilizados por serem de fácil entendimento e implementação (PRECHELT, 1998).

Na parada antecipada, o conjunto de treinamento é usado para computar o gradiente e atualizar os pesos da rede. O erro do conjunto de validação é monitorado durante o processo de treinamento. No entanto, quando a rede inicia a sobre-ajustar os dados, o erro no grupo de validação irá aumentar. Quando o erro de validação aumenta para um número específico de iterações, o treinamento é parado e os pesos no erro mínimo de validação são retornados. Esta regra é conhecida como parada antecipada e está ilustrada na Figura 3.6. A parada antecipara foi utilizada no presente trabalho. A raiz quadrada do erro quadrático médio (*RMSE - Root Mean Squared Error*) do conjunto de validação (dados da SIX) e teste (dados da RLAM) foi usada para avaliar a performance dos modelos empíricos usados. *RMSE* é definida no Capítulo 4 (Equação 4.15).

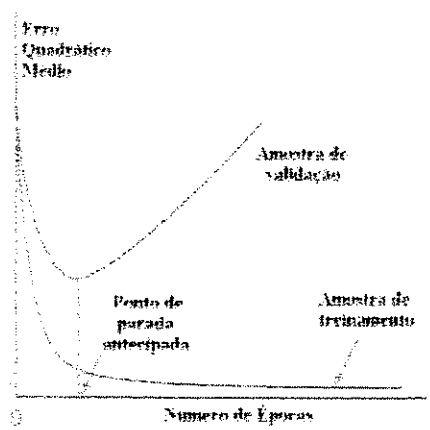


Figura 3.6: Ilustração da regra de parada antecipada.

3.7.4.2 Regularização Bayesiana

Um modelo desejado de rede neural deve produzir pequeno erro não somente nos dados de treinamento mas também nos dados que não pertencem ao conjunto de treinamento (conjunto de validação e/ou teste). Para produzir uma rede com a melhor

capacidade de generalização, MACKAY (1992) propôs um método para restringir os valores dos parâmetros da rede através da regularização. A técnica de regularização força a rede a responder suavemente e então o sobreajuste é pouco provável (FORESEE e HAGAN, 1997).

Na técnica de regularização, a função de custo F é definida como:

$$F_{reg} = \alpha F + \beta F_w \tag{3.64}$$

onde:

F – função típica para treinar redes neurais do tipo MLP que é a soma dos erros quadráticos dado por:

$$F = \frac{1}{N}\sum_{i=1}^{N}(s_i - y_i)^2 \tag{3.65}$$

F_w – é a soma dos quadrados dos parâmetros (pesos e *bias*) que é dado por:

$$F_w = \frac{1}{n}\sum_{j=1}^{n} w_j^2 \tag{3.66}$$

α e β – são os parâmetros da função objetivo.

O problema da regularização é a dificuldade em determinar um valor adequado para a taxa. Se o valor for muito grande poderá levar a um sobreajuste e se for muito pequeno a rede poderá não ajustar adequadamente os dados de treinamento (FISCHER, 2002). É desejável determinar esses parâmetros de uma forma automatizada; uma destas abordagens é um processo que usa a estrutura Bayesiana.

Na estrutura Bayesiana, os pesos são considerados aleatórios e variáveis. Depois os dados são tomados e a função densidade para os pesos podem ser antecipados de acordo com a regra de Bayes:

$$P(\mathbf{w} \mid D, \alpha, \beta, M) = \frac{P(D \mid \mathbf{w}, \beta, M) P(\mathbf{w} \mid \alpha, M)}{P(D \mid \alpha, \beta, M)} \tag{3.67}$$

Onde D representa o conjunto de dados, M é o modelo de rede neural usado e $\mathbf{w}$ é o vetor de pesos da rede. $P(\mathbf{w} \mid \alpha, M)$ é a densidade anterior, o qual representa o conhecimento dos pesos antes de qualquer dado ser coletado. $P(D \mid \mathbf{w}, \beta, M)$ é a função de probabilidade dos dados, dados os pesos $\mathbf{w}$. $P(D \mid \alpha, \beta, M)$ é o fator de normalização, que garante que a probabilidade total seja igual a 1.

Se for assumido que o ruído no conjunto de treinamento é Gaussiano e a distribuição anterior para os pesos é Gaussiana, a densidade de probabilidade pode ser escrita como:

$$\begin{aligned} P(D \mid \mathbf{w}, \beta, M) &= \frac{1}{Z_D(\beta)} \exp(-\beta E_D) \\ P(\mathbf{w} \mid D, \alpha, \beta, M) &= \frac{1}{Z_D(\alpha)} \exp(-\alpha E_w) \end{aligned} \tag{3.68}$$

onde $Z_D(\beta) = \left(\pi/\beta\right)^{n/2}$ e $Z_w(\alpha) = \left(\pi/\alpha\right)^{N/2}$. Se substituirmos estas probabilidades na Equação 3.67, obtemos:

$$\begin{aligned} P(\mathbf{w} \mid D, \alpha, \beta, M) &= \frac{\frac{1}{Z_D(\beta)} \frac{1}{Z_W(\alpha)} \exp(-(\beta E_D + \alpha E_w))}{\text{fator de normalização}} \\ &= \frac{1}{Z_F(\alpha, \beta)} \exp(-F(\mathbf{w})) \end{aligned} \tag{3.69}$$

Nesta estrutura Bayesiana, os pesos ótimos deverão maximizar a probabilidade posterior $P(\mathbf{w} \mid D, \alpha, \beta, M)$. Maximizando a probabilidade posterior é equivalente a minimizar a função objetiva regularizada $F = \alpha E_D + \beta E_w$.

Para otimizar os parâmetros da função objetivo α e β, agora vamos considerar a aplicação da regra de Bayes. Agora tem-se:

$$P(\alpha,\beta \mid D,M)=\frac{P(D \mid \alpha,\beta,M)P(\alpha,\beta \mid M)}{P(D \mid M)} \tag{3.70}$$

Se assumirmos como uniforme a densidade anterior $P(\alpha,\beta|D, M)$ para os parâmetros α e β, então a maximização da posterior é realizada pela maximização da função de probabilidade $P(D|\alpha,\beta,M)$. Note que esta função de probabilidade é o fator de normalização da Equação 3.67. Desde que todas as probabilidades tenham a forma Gaussiana, pode-se conhecer a forma da densidade posterior da Equação 3.67. Isto é mostrado na Equação 3.69. Agora pode-se resolver a Equação 3.67 para o fator de normalização.

$$\begin{aligned} P(D \mid \alpha,\beta,M) &= \frac{P(D \mid \mathbf{w},\beta,M)P(\mathbf{w} \mid \alpha,M)}{P(\mathbf{w} \mid D,\alpha,\beta,M)} = \\ &= \frac{\left[\frac{1}{Z_D(\beta)}\exp(-\beta E_D)\right]\left[\frac{1}{Z_w(\alpha)}\exp(-\alpha E_w)\right]}{\frac{1}{Z_F(\alpha,\beta)}\exp(-F(\mathbf{w}))} = \\ &= \frac{Z_F(\alpha,\beta)}{Z_D(\beta)Z_w(\alpha)}\cdot\frac{\exp(-\beta E_D-\alpha E_w)}{\exp(-F(\mathbf{w}))} = \frac{Z_F(\alpha,\beta)}{Z_D(\beta)Z_w(\alpha)} \end{aligned} \tag{3.71}$$

Note que se conhece constantes $Z_D(\beta)$ e $Z_w(\alpha)$ da Equação 3.68, somente uma parte não é conhecida $ZF(\alpha,\beta)$. Mas pode-se estimá-la por expansão com série de Taylor. Desde que a função objetivo tenha a forma quadrática em uma pequena área ao redor do ponto mínimo, podemos expandir $F(\mathbf{w})$ em volta do ponto mínimo da densidade posterior $\mathbf{w}^{MP}$, onde o gradiente é zero. Resolvendo para a constante de normalização obtemos:

$$Z_F = (2\pi)^{N/2}\left(\det\left(\left(\mathbf{H}^{MP}\right)^{-1}\right)\right)^{1/2}\exp\left(-F\left(\mathbf{w}^{MP}\right)\right) \tag{3.72}$$

onde $H=\beta\nabla^2 E_D+\alpha\nabla^2 E_w$ é a matriz Hessiana da função objetivo. Colocando este resultado na Equação 3.71, pode-se resolver para os valores ótimos de α e β no ponto de mínimo. Fazemos isto pegando a derivada em relação a cada logaritmo de Equação 3.71 e igualamos a zero. Obtemos:

$$\alpha^{MP} = \frac{\gamma}{2E_w\left(\mathbf{w}^{MP}\right)} \quad \text{e} \quad \beta^{MP} = \frac{n-\gamma}{2E_D\left(\mathbf{w}^{MP}\right)} \tag{3.73}$$

onde $\gamma = N - 2\alpha^{MP} tr\left(\mathbf{H}^{MP}\right)$ é chamado de número efetivo de parâmetros e N é o número total de parâmetros. O parâmetro γ é a medida de quantos parâmetros da rede são efetivamente usados na redução da função erro. Este parâmetro pode variar de 0 até N.

FORESEE e HAGAN (1997) propuseram aplicar a aproximação Gauss-Newton a matriz Hessiana, a qual pode ser convenientemente implementado se o algoritmo de otimização Levenberg-Marquardt é usado para a localização do ponto mínimo. Este minimiza a computação adicional requerida para a regularização.

3.8 Considerações finais sobre o capítulo

O objetivo deste capítulo foi apresentar um estudo básico sobre redes neurais artificiais para uma melhor compreensão dos modelos desenvolvidos para o processo de craqueamento catalítico.

Foi feita a apresentação do algoritmo de retro-propagação do gradiente do erro (*backpropagation*) para o treinamento supervisionado de redes multicamadas. Em seguida, foram descritos três métodos de otimização não linear para treinamento de redes do tipo MLP: método de Levenberg-Marquardt (LM), método de Broyden-Fletcher-Goldfarb-Shanno (BFGS) e método de gradiente conjugado escalonado (SCG

Este capítulo terminou mostrando a metodologia adotada para avaliar as redes neurais usadas. Foram modificados os métodos de otimização para treinamento das redes e dentro de cada método também foi modificado o número de neurônios na camada oculta. A finalidade foi obter o melhor método de treinamento com um definido esquema de funções de transferência (neurônios ocultos e de saída) com um número ótimo de neurônios na camada oculta. Também foram consideradas as duas técnicas para generalização das redes neurais usadas (parada antecipada e regularização Bayesiana) visando obter um modelo com maior poder de predição. Os resultados estão descritos no Capítulo 5 e foram associados ao erro do conjunto de teste.

CAPÍTULO 4

QUIMIOMETRIA

4.1 Introdução

No moderno ambiente industrial, os dados de processo fornecem a base para monitoramento, avaliação e controle da qualidade dos produtos formados. A coleta e armazenamento de grandes quantidades de dados têm-se tornados operações rotineiras como conseqüência do avanço tecnológico de computadores. Uma importante etapa no entendimento do comportamento de um processo é a extração de características importantes contidas dentro dos dados do processo (MARTIN e MORRIS, 1998). Devido as muitas variáveis envolvidas na entrada e na saída de muitos processos (em particular o processo de craqueamento catalítico), técnicas da quimiometria têm-se mostrado mais adequadas para tratar dados com essas características pois permite a redução da dimensionalidade do problema, tornando possível resumir a informação contida num grande número de variáveis altamente correlacionadas por um número menor de componentes principais (ou variáveis latentes).

O uso da quimiometria tem tido muitas aplicações na indústria química tais como detecção de falhas no processo (MARTIN e MORRIS, 1998), análise exploratória de dados (SANTEN *et al.*, 1997), monitoramento de processos (KOURTI e MACGREGOR, 1995; ZHANG *et al.*, 1997; KULKARNI *et al.*, 2004) e modelagem de processos (OLIVEIRA-ESQUERRE, 2003; KULKARNI *et al.*, 2004).

O presente capítulo apresenta alguns métodos da quimiometria que foram usados no trabalho. O objetivo é obter modelos através da redução da dimensionalidade das variáveis do processo de craqueamento catalítico usando métodos clássicos de regressão (análise dos componentes principais e mínimos quadrados parciais) e métodos que combinem a quimiometria com redes neurais artificiais. A nomenclatura usada aqui é a mesma usada no

Capítulo 3, ou seja, **X** é a matriz das variáveis independentes (variáveis de entrada do processo) e **Y** e a matriz das variáveis dependentes (variáveis de saída do processo).

Antes da apresentação dos métodos da quimiometria, será feita uma discussão sobre as formas de pré-processamento dos dados.

4.2 Pré-processamentos

Antes de iniciar qualquer tipo de análise multivariada é necessária a realização de uma manipulação matemática prévia do conjunto de dados para adequação ou, às vezes, até mesmo a remoção de possíveis fontes de variação (BEEBE *et al.*, 1998). Em diversos problemas temos variáveis com diferentes dimensões e amplitudes e é necessário um tratamento prévio para expressar cada observação em dimensões e amplitudes equivalentes, sem perda de informações. Existem várias formas de realizar este pré-processamento, os mais comuns são centrar os dados na média, escalamento pela variância e o autoescalamento. Outros tipos de pré-processamentos podem ser encontrados em BEEBE *et al.* (1998).

4.2.1 Dados centrados na média

Neste tipo de pré-processamento, a média de cada variável é subtraída de seus respectivos elementos, como indicado na Equação 4.1. Desta forma, a origem dos eixos no qual os dados se encontram é deslocada de modo a colocar os dados numa forma mais conveniente à análise e visualização.

$$x_{ij(cm)} = x_{ij} - \bar{x}_j \tag{4.1}$$

Onde:

$x_{ij(cm)}$ – valor centrado na média para a variável j na amostra i;

x_{ij} – valor da variável j na amostra i;

$\bar{x}_j$ – média dos valores das amostras na coluna j, calculado pela Equação (4.2):

$$\bar{x}_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij} \tag{4.2}$$

Na Equação 4.2, m é o número de amostras.

4.2.2 Escalamento pela variância

Neste tipo de pré-processamento, cada valor da variável j é dividido pelo seu desvio padrão, como na Equação 4.3. É utilizado quando as variáveis possuem dimensões muito discrepantes entre si. Desta forma, o peso das variáveis em diferentes escalas é considerado equivalente, minimizando o risco de perda de informações relevantes.

$$x_{ij(ev)} = \frac{x_{ij}}{S_j} \tag{4.3}$$

onde:

$x_{ij(ev)}$ – valor escalado pela variância para a variável j na amostra i;

x_{ij} – valor da variável j na amostra i;

S_j – desvio padrão dos valores da variável j, calculado a partir da variância s_j^2 que é dada por:

$$s_j^2 = \frac{1}{m-1}\sum_{i=1}^{m}\left(x_{ij} - \bar{x}_j\right)^2 \tag{4.4}$$

4.2.3 Autoescalamento

O autoescalamento aplica ambas as técnicas descritas anteriormente de uma só vez, de modo que a transformação realizada sobre o conjunto original dos dados permite que cada variável apresente média zero e variância igual a um. Desta forma será dada a mesma importância para todas as variáveis, independente da sua dimensão (Equação 4.5).

$$x_{ij(as)} = \frac{x_{ij} - \bar{x}_j}{S_j} \tag{4.5}$$

onde:

$x_{ij(as)}$ – valor centrado na média para a variável *j* na amostra *i*;

No presente trabalho, utilizou-se a última técnica para pré-processamento dos dados, pois os mesmos apresentam medidas das variáveis em unidades e amplitudes diferentes.

4.3 Análise dos Componentes Principais (PCA)

4.3.1 Introdução

A análise dos componentes principais (PCA – *Principal Component Analysis*) é uma ferramenta para compressão de dados e extração de informações. A técnica de PCA encontra combinações de variáveis, ou fatores, que descrevem a maior tendência nos dados (WISE *et al.*, 2003).

De acordo com FERREIRA *et al.* (1999), a PCA consiste numa manipulação da matriz de dados com o objetivo de representar variações presentes em muitas variáveis, através de um número menor de fatores. Constrói-se um novo sistema de eixos (denominados rotineiramente de fatores, componentes principais, variáveis latentes ou ainda autovetores) para representar as amostras, no qual a natureza multivariada dos dados pode ser visualizada em poucas dimensões.

De uma forma simplificada, a PCA corresponde a decomposição da matriz de dados **X**, de dimensão *m* x *n*, no produto de duas matrizes: a matriz dos escores, **T**, e a transposta da matriz dos "pesos" (*loadings*), $\mathbf{P}^T$, como mostra a Equação 4.6.

$$\mathbf{X} = \mathbf{T}.\mathbf{P}^T \tag{4.6}$$

Os vetores linha da matriz dos escores **T** correspondem às projeções das *m* amostras da matriz original **X**, nos novos eixos formados pelas *n* combinações lineares das variáveis originais, ou seja, cada vetor coluna da matriz dos escores corresponde às coordenadas das amostras em cada componente principal gerada. Desta forma, o número *m* de linhas da matriz original é igual ao número de linhas da matriz dos escores e o número de colunas corresponde ao número de *n* componentes geradas (WOLD *et al.*, 1987).

Na matriz dos pesos **P**, cada vetor coluna corresponde aos pesos que cada variável possui na combinação linear das *n* componentes geradas. Então esta matriz possui *n* linhas (as variáveis originais) e *n* colunas (as componentes geradas) (WOLD *et al.*, 1987).

Há uma variedade de algoritmos usados para calcular os escores e os pesos para uma análise dos componentes principais, sendo que os mais conhecidos são a decomposição de valores singulares (SVD) e o NIPALS (*non iterative partial least squares*).

Cada componente principal gerada descreve uma certa quantidade de informação através da combinação linear das variáveis originais na direção de maior variância dos dados, mas não descreve a variância total. A cada componente modelada uma certa quantidade de informação permanece sem ser explicada, ou seja, uma quantidade de variância continua não descrita. A esta quantidade de variância não descrita chamamos de resíduos e podem ser organizados na forma da matriz de resíduos **E** (RIBEIRO, 2001). A Equação 4.6 pode ser reescrita como:

$$\mathbf{X} = \mathbf{T}_k . \mathbf{P}_k^{\mathrm{T}} + \mathbf{E} \tag{4.7}$$

Este resíduo é formado pela subtração de cada elemento da matriz original **X** dos produtos da matriz dos escores e da matriz transposta dos pesos, como na Equação 4.8. Este resíduo será utilizado na escolha do número ótimo de componentes principais (*k*) a ser utilizado na análise.

$$\mathbf{E} = \mathbf{X} - \mathbf{T}_k . \mathbf{P}_k^{\mathrm{T}} \tag{4.8}$$

A dimensão da matriz de dados original necessária à completa descrição do sistema corresponde ao número de colunas, ou seja, ao número de variáveis (n). Após a PCA, a dimensão do sistema corresponde ao número de colunas da matriz de pesos, ou seja, ao número de componentes principais (k) escolhidas, que é menor que o número de variáveis originais (RIBEIRO, 2001).

A seguir será feita uma descrição mais formal do método PCA e da estatística associada com este método extraído de WISE *et al.* (2003).

4.3.2 O método PCA

A PCA tem como base encontrar os autovalores e autovetores de uma matriz de covariância. A matriz de covariância é uma medida da associação entre as variáveis. Para uma dada matriz **X** com m linhas e n colunas, com cada variável sendo uma coluna e cada amostra uma linha, a matriz de covariância de **X** é definida como:

$$\mathbf{Z} = \mathbf{cov}(\mathbf{X}) = \frac{1}{\mathrm{m}-1}\mathbf{X}^{\mathrm{T}}.\mathbf{X} \tag{4.9}$$

A PCA decompõe a matriz de dados **X** como soma dos produtos dos vetores $\mathbf{t}_i$ e $\mathbf{p}_i$ mais a matriz de resíduos **E**:

$$\mathbf{X} = \mathbf{t}_1.\mathbf{p}_1^{\mathrm{T}} + \mathbf{t}_2.\mathbf{p}_2^{\mathrm{T}} + \mathrm{K} + \mathbf{t}_k.\mathbf{p}_k^{\mathrm{T}} + \mathbf{E} \tag{4.10}$$

onde k é o número de componentes principais escolhido para o modelo. Os vetores $\mathbf{t}_i$ são conhecidos como escores e contém informação sobre como as amostras se relacionam. Os vetores $\mathbf{p}_i$ são conhecidos como os pesos e contém informação sobre como as variáveis se relacionam.

Na decomposição da PCA, os vetores $\mathbf{p}_i$ são os autovetores da matriz de covariância, isto é, para cada $\mathbf{p}_i$

$$\mathbf{Z}.\mathbf{p}_i = \lambda_i.\mathbf{p}_i \tag{4.11}$$

onde λ_i é o autovalor associado com o autovetor $\mathbf{p}_i$. Os $\mathbf{t}_i$ formam grupos ortogonais ($\mathbf{t}_i^T.\mathbf{t}_j = 0$ para $\mathbf{i} \neq \mathbf{j}$), enquanto os $\mathbf{p}_i$ são ortonormais ($\mathbf{p}_i^T.\mathbf{p}_j = 0$ para $\mathbf{i} \neq \mathbf{j}$, $\mathbf{p}_i^T.\mathbf{p}_j = 1$ para $\mathbf{i} = \mathbf{j}$). Observe que, para $\mathbf{X}$ e qualquer par $\mathbf{t}_i$, $\mathbf{p}_i$ temos:

$$\mathbf{X}.\mathbf{p}_i = \mathbf{t}_i \tag{4.12}$$

isto é, o vetor escores $\mathbf{t}_i$ é a combinação linear das variáveis originais definidas por $\mathbf{p}_i$. Uma outra forma de ver isto é que $\mathbf{t}_i$ são projeções de $\mathbf{X}$ sobre $\mathbf{p}_i$. Os pares $\mathbf{t}_i$, $\mathbf{p}_i$ são arranjados em ordem decrescente de acordo com os λ_i associados.

Os λ_i são medidas das quantidades de variância descrita pelo par $\mathbf{t}_i$, $\mathbf{p}_i$. Neste contexto, podemos associar variância com informação. Como os pares $\mathbf{t}_i$, $\mathbf{p}_i$ estão em ordem decrescente λ_i, o primeiro par captura a maior quantidade de informação na decomposição. De fato, pode-se mostrar que o par $\mathbf{t}_i$, $\mathbf{p}_i$ captura a maior quantidade de variação nos dados que é possível capturar com um fator linear e cada par subseqüente captura a maior quantidade possível de variância restante após subtrair $\mathbf{t}_i.\mathbf{p}_i^T$ de $\mathbf{X}$.

Geralmente os dados podem ser descritos adequadamente usando poucos fatores (escores) ao invés das variáveis originais, sem perda significante de informação. A análise dos componentes principais é exemplificada graficamente na Figura 4.1. Nesta figura vemos três variáveis medidas numa coleção de amostras. Quando postos em um gráfico tridimensional observa-se que todas as amostras podem ser confinadas em uma elipse. Pode-se observar que as amostras variam mais ao longo de um eixo da elipse do que ao longo do outro eixo. A primeira componente principal (CP) descreve a direção da maior variação no conjunto de dados, o qual é o maior eixo da elipse. A segunda CP descreve a direção da segunda maior variação que é o menor eixo da elipse. Neste caso, um modelo de PCA (os vetores dos escores e dos pesos e os autovalores associados) com duas componentes principais descrevem adequadamente toda a variação nas dimensões.

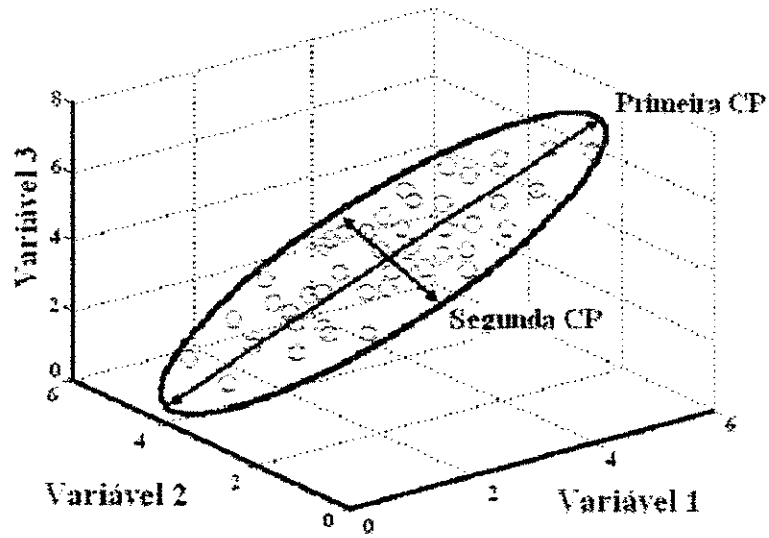


Figura 4.1: Representação gráfica da análise dos componentes principais

Um ponto importante a ser considerado é a determinação do número de componentes principais (k) a ser usado, pois o uso de todas as componentes principais após a decomposição da matriz de dados não é, em geral, justificado. Existem diversos critérios (heurísticos e estatísticos) para decidir quantas componentes principais devem ser usadas na PCA, dentre eles temos: a porcentagem de variância explicada e a validação cruzada.

A porcentagem de variância explicada, um critério heurístico, é aplicada quando se tem suficiente experiência em conjuntos de dados similares (OTTO, 1999). A fração de variância explicada (acumulada), S_e^2, é calculada a partir da razão da soma de d autovalores importantes e a soma de todos p autovalores, segundo a Equação (4.13):

$$S_e^2 = \frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{p} \lambda_i} .(100\%) \tag{4.13}$$

Se todas as componentes principais são usadas no modelo, 100% da variância é explicada. Em geral uma percentagem fixa de variância explicada é especificada, por exemplo, 90% da variância total. Por exemplo, considerando o conjunto de dados indicados na Figura 4.1 temos o seguinte resultado mostrado na Tabela 4.1:

Tabela 4.1: Autovalores e variâncias explicadas para o conjunto fictício da Figura 4.1.

Componente	Autovalor (λ)	Variância Explicada (%)	Variância Acumulada (%)
CP1	3,352	71,03	71,03
CP2	1,182	25,05	96,08
CP3	0,185	3,92	100,00

A CP1 descreve 71,03% da variância total dos dados, o que é insuficiente, pois a variância residual ainda é muito grande; alguma informação poderá estar sendo perdida, por exemplo, a contribuição de variáveis com alto peso na CP2 poderia estar sendo desprezada, o que prejudicaria a descrição dos dados. Torna-se necessário a utilização de mais uma componente. A CP2 descreve 25,05% da variância total dos dados, que somada à CP1 descreverão juntas 96,08%. A CP3 descreve apenas 3,92% da variância total dos dados; a sua inclusão irá descrever todo o conjunto de dados (100% da variância total). Apesar disto, a quantidade de informação descrita pela CP3 é pequena e a sua inclusão poderia estar inserindo resíduo desnecessário a descrição dos dados (RIBEIRO, 2001). Pode-se dizer então, para este caso, que duas componentes principais seriam suficientes para o modelo de PCA.

4.3.3 Estatística T^2 de Hotelling para PCA

Os escores **t** de um modelo PCA que tenham média zero (assumindo que a matriz de dados **X** foi centrada na média ou autoescalada antes de decomposição) com variância igual aos seus autovalores associados. Assim, é assumido que os escores são distribuídos normalmente onde se pode calcular os limites de confiança para os escores aplicando a distribuição *t* de Student (WISE *et al.*, 2003). Para um completo desenvolvimento da estatística T^2 partindo da teoria da distribuição t consultar (BARTHUS, 1999).

O valor de T^2 é estimado pelos escores **t** das medidas multivariadas no espaço definido pelas *k* componentes principais. O T^2 é uma medida da variação de cada amostra dentro do modelo PCA e é definido como:

$$\mathbf{T_i}^2 = \sum_{i=1}^{k} \frac{\mathbf{t_i^2}}{\mathbf{s_{ii}^2}} \tag{4.16}$$

Onde $\mathbf{t_i}$ são os escores e $\mathbf{s_{ii}^2}$ é a medida da variância destes escores. A variância é dada pelos autovalores (λ).

Quando se divide os escores pela variância, ou seja pelos autovalores, estabelece-se que cada um dos componentes principais contribuem igualmente para o cálculo de T^2. O cálculo do limite de confiança para T^2 é dado com base na distribuição F, como segue (BARTHUS, 1999):

$$\mathbf{T}^2 = \frac{(m-1)k}{m-k} \mathbf{.F}_{k,m-k,\alpha} \tag{4.17}$$

Onde m é o número de amostras usadas para desenvolver o modelo de PCA, k é o número de componentes retidas no modelo e α a porcentagem da região do limite de confiança.

De posse do limite de confiança para T^2 e dos valores individuais de cada amostra, pode-se determinar quais delas estão dentro ou fora do modelo de PCA.

4.3.4 Regressão dos Componentes Principais (PCR)

A regressão dos componentes principais (*principal components regression*) utiliza um modelo PCA para decompor a matriz de dados **X** de variáveis independentes e então relaciona os resultados da PCA com o vetor da variável dependente **y**.

A idéia básica por trás da PCR é que nem toda variação no conjunto de dados independentes pode predizer as variáveis dependentes. Em outras palavras, a PCR é utilizada quando um grande número de variáveis independentes que apresentam extensiva colinearidade ou correlação entre elas está disponível para a formação de um modelo de predição. Colinearidade adiciona redundância ao modelo causando instabilidade numérica na estimativa dos coeficientes de regressão (OLIVEIRA-ESQUERRE, 2003).

A regressão das componentes principais é feita através da regressão do vetor **y** sobre os k primeiros escores obtidos pela decomposição da matriz **X**. O vetor de coeficientes de regressão **b** é obtido usando a equação de regressão, o qual relaciona a matriz de escores ($\mathbf{T}_k$) com o vetor **y**.

Matematicamente falando, primeiro fazemos a PCA da matriz de dados **X**, conforme Equação 4.7 já descrita:

$$\mathbf{X} = \mathbf{T}_k . \mathbf{P}_k^{\mathrm{T}} + \mathbf{E} \tag{4.7}$$

Em seguida, é feita a correlação entre a matriz de escores ($\mathbf{T}_k$) (com k componentes principais e que contem as informações das variáveis dependentes) e o vetor **y** da variável dependente como descrito pela Equação 4.16.

$$\mathbf{y} = \mathbf{T}_k . \mathbf{b} + \mathbf{e} \tag{4.16}$$

A solução para determinar o vetor de coeficientes de regressão (**b**) pode ser vista na Equação 4.17.

$$\mathbf{b} = \left(\mathbf{T}_k{}^{\mathrm{T}} . \mathbf{T}_k\right)^{-1} . \mathbf{T}_k . \mathbf{y} \tag{4.17}$$

Neste caso, a inversão $\left(\mathbf{T}_k{}^{\mathrm{T}} . \mathbf{T}_k\right)$ não irá causar problemas devido à ortogonalidade entre as linhas da matriz dos escores.

Apesar da PCR resolver o problema de colinearidade e possibilitar a formação de um modelo de menor dimensão (permitindo redução de ruído); este apresenta o risco de informações preditivas úteis estarem sendo descartadas com as componentes principais e algum ruído permanecer nas componentes usadas na regressão. Isto ocorre pelo fato de que para a formação das componentes não se utilizam informações sobre a relação entre **X** e a variável a ser predita, mesmo havendo uma tendência geral das componentes com maiores variâncias explicarem melhor as variáveis dependentes (OLIVEIRA-ESQUERRE, 2003).

4.4 Mínimos Quadrados Parciais (PLS)

4.4.1 Introdução

O método PLS (*partial least square*) é uma técnica estatística de regressão multivariada a qual realiza uma decomposição simultânea das matrizes **X** (variáveis independentes) e **Y** (variáveis dependentes) em uma soma de produtos de dois vetores (os escores e os pesos).

Sabe-se que é possível representar uma matriz de dados, sem a perda de informação estatística útil, pela sua matriz de escores, com a vantagem de não haver correlação entre as variáveis. Isto é exatamente o que se faz no PLS, ou seja, tanto a matriz de variáveis independentes (**X**) como a das variáveis dependentes (**Y**) são representadas por seus escores, utilizando a redução de variáveis pela análise dos componentes principais (BARTHUS, 1999). O modelo resultante é:

$$\begin{aligned} \mathbf{X} &= \mathbf{T}.\mathbf{P}^{\mathrm{T}} + \mathbf{E} \\ \mathbf{Y} &= \mathbf{U}.\mathbf{Q}^{\mathrm{T}} + \mathbf{F} \end{aligned} \tag{4.18}$$

onde **T** e **U** são as matrizes de escores das matrizes **X** e **Y**, respectivamente; **P** e **Q** são as matrizes de pesos das matrizes **X** e **Y**, respectivamente; e **E** e **F** são os resíduos.

A correlação entre os dois blocos **X** e **Y** é simplesmente uma relação linear obtida pelo coeficiente de regressão linear, tal como descrito abaixo,

$$\mathbf{u}_h = \mathbf{b}_h.\mathbf{t}_h \tag{4.19}$$

para h variáveis latentes, sendo que os valores de $\mathbf{b}_h$ são agrupados na matriz diagonal **B**, que contém os coeficientes de regressão entre a matriz de escores **U** de **Y** e a matriz de escores **T** de **X**. A melhor relação linear possível entre os escores desses dois blocos é obtida através de pequenas rotações das variáveis latentes dos blocos de **X** e **Y** (site do LAQQA – Laboratório de Quimiometria em Química Analítica, 2004).

A matriz **Y** pode ser calculada de $\mathbf{u}_h$,

$$\mathbf{X} = \mathbf{T}.\mathbf{B}.\mathbf{Q}^{\mathrm{T}} + \mathbf{F} \tag{4.20}$$

e a concentração de novas amostras prevista a partir dos novos escores, **T***, substituídos na equação anterior.

$$\mathbf{Y} = \mathbf{T}^*.\mathbf{B}.\mathbf{Q}^{\mathrm{T}} \tag{4.21}$$

Nesse processo, é necessário achar o melhor número de variáveis latentes, o que normalmente é feito usando a validação cruzada (descrito na Seção 4.3.2), na qual o erro mínimo de previsão é determinado.

4.4.2 O método PLS

Matematicamente, o algoritmo PLS muda os escores entre **X** e **Y** quando procede a decomposições das matrizes, resultando em variáveis latentes altamente correlacionáveis. Neste caso, existe um compromisso entre a explicação da variância em **X** e em encontrar a correlação com **Y**.

O PLS é iniciado selecionando uma coluna de **Y**, $\mathbf{y}_i$, como estimativa de partida para $\mathbf{u}_h$ (geralmente a coluna de **Y** com maior variância é escolhida). As componentes principais são estimadas iterativamente, isto é, pelo uso do algoritmo NIPALS. O algoritmo é mostrado a seguir:

Faça h =1 (1° variável latente).

(1) Como descrito acima, escolhemos uma coluna de **Y** e a usamos como vetor de partida para $\mathbf{u}_1$:

$$\mathbf{u}_h = \mathbf{u}_1 = \mathbf{y}_j$$

(2) Compute os pesos de **X** do PLS:

$$\mathbf{w}_1^T = \frac{\mathbf{u}_1^T.\mathbf{X}}{\mathbf{u}_1^T.\mathbf{u}_1}$$

(3) Escalar os pesos para um vetor de comprimento um:

$$\mathbf{w}_1^T = \frac{\mathbf{w}_1^T}{\left\|\mathbf{w}_1^T\right\|} = \frac{\mathbf{w}_1^T}{\left(\mathbf{w}_1^T.\mathbf{w}_1\right)^{1/2}}$$

(4) Estime os escores da matriz **X**:

$$\mathbf{t}_1 = \mathbf{X}.\mathbf{w}_1^T$$

(5) Compute os pesos da matriz **Y**:

$$\mathbf{q}_1^T = \frac{\mathbf{t}_1^T.\mathbf{Y}}{\mathbf{t}_1^T.\mathbf{t}_1}$$

(6) Gerar o vetor dos escores de **Y**:

$$\mathbf{u}_1 = \frac{\mathbf{Y}.\mathbf{q}_1}{\mathbf{q}_1^T.\mathbf{q}_1}$$

(7) Cheque a convergência comparando $\mathbf{t}_1$ na etapa (4) com o valor da iteração anterior. Se eles forem iguais (ou dentro de erro considerado), siga para etapa (8). Caso contrário, retorne a etapa (2) e use $\mathbf{u}_1$ calculado na etapa (6).

(8) Obtendo os pesos de **X**:

$$\mathbf{p}_1^T = \frac{\mathbf{t}_1^T.\mathbf{X}}{\mathbf{t}_1^T.\mathbf{t}_1}$$

(9) Determinando o coeficiente de relação interna (escalar):

$$b_1 = \frac{\mathbf{u}_1^T.\mathbf{t}_1}{\mathbf{t}_1^T.\mathbf{t}_1}$$

(10) Calculando os resíduos:

$$\mathbf{E}_h = \mathbf{E}_{h-1} - \mathbf{t}_h.\mathbf{p}_h^T \qquad (\text{onde } \mathbf{E}_0 = \mathbf{X})$$

$$\mathbf{F}_h = \mathbf{F}_{h-1} - b_h.\mathbf{t}_h.\mathbf{q}_h^T \qquad (\text{onde } \mathbf{F}_0 = \mathbf{Y})$$

(11) Volte a etapa (1) para implementar o procedimento para a próxima variável latente, fazendo:

$$\mathbf{X} = \mathbf{E}_h$$

$$\mathbf{Y} = \mathbf{F}_h$$

$$h = h + 1$$

(12) Após o cálculo para d variáveis latentes, determine os coeficientes de regressão:

$$\mathbf{B} = \mathbf{W}\left(\mathbf{P}^{\mathrm{T}}.\mathbf{W}\right)^{-1}.\mathbf{Q}^{\mathrm{T}}$$

4.4.3 Validação Cruzada

Na obtenção de um modelo PLS (ou PCR) é fundamental determinar o número correto de variáveis latentes (ou componentes principais). Isto pode ser feito utilizando o método da validação cruzada.

Neste método separa-se uma parte (ou apenas uma) das amostras da matriz de dados, **X**, e constrói-se o modelo de PCA com as amostras restantes. Estima-se os erros de previsão para as amostras que foram separadas através da soma dos quadrados dos erros de previsão, *PRESS*, dado pela Equação 4.14;

$$PRESS = \sum_i \left(\hat{y}_i - y_i\right)^2 \tag{4.14}$$

ou a raiz quadrada do erro quadrático médio (*RMSE*) de previsão (*root mean square error prediction*):

$$RMSE = \sqrt{\frac{\sum_i \left(\hat{y}_i - y_i\right)^2}{n}} \tag{4.15}$$

onde

$\hat{y}_i$ é o valor previsto para a amostra i utilizando o modelo;

y_i é o valor medido para a amostra i;

n - número de amostras do conjunto

A seguir, estas amostras são incluídas na matriz de dados e outras são retiradas e o processo é repetido até que todas as amostras do conjunto de dados sejam utilizadas. Este processo é repetido para modelos com uma, duas e assim por diante, componentes principais. O número de componentes principais significantes (k) é obtido pelo menor valor de *PRESS* ou *RMSE*.

4.5 PCR e PLS com relações internas não-lineares

Como apresentado anteriormente, PCR e PLS são técnicas lineares. PCR e PLS podem ser usados com dados não-lineares de duas maneiras. Primeiro, pode ser possível transformar os dados para um formato linear (ou aproximadamente linear). Tais transformações são geralmente conseguidas por meio de considerações teóricas. A segunda forma, é transformar os métodos PCR e PLS em métodos não-lineares através da mudança da relação interna entre os escores de **X** e **Y**. Existem trabalhos que propõem a incorporação da técnica de redes neurais artificiais como relação não-linear em modelos de PCR e PLS (MARTIN e MORRIS, 1998; QIN e MCAVOY,1992; HOLCOMB e MORARI, 1992; MALTHOUSE et al., 1996; BAFFI et al., 1999; DONG e MCAVOY, 1995).

4.6 Considerações finais sobre o capítulo

Este capítulo apresentou os métodos da estatística multivariada que serão empregados para uma análise exploratória dos dados visando buscar possíveis amostras com comportamento anômalo e em seguida aplicar os modelos de regressão que foram descritos.

Os modelos PCR e PLS descritos são técnicas de regressão lineares e que, possivelmente, não irão fornecer bons resultados para o processo de craqueamento catalítico visto que o mesmo se apresenta como um processo altamente não-linear. Por isso, serão aplicados também os modelos PCR e PLS que utilizam redes neurais artificiais nas relações internas.

O capítulo a seguir descreve os resultados de tais modelos e os compara com os resultados obtidos pela modelagem que utiliza apenas redes neurais artificiais.

CAPÍTULO 5

RESULTADOS E DISCUSSÕES

Neste capítulo são apresentados os resultados obtidos pelos modelos empíricos estudados nos capítulos anteriores. Nas Seções 5.1 e 5.2 são mostrados os resultados referentes à planta piloto de FCC da SIX para o modelo global e o modelo individual, respectivamente. Os bons resultados motivaram a aplicação dos modelos em uma unidade industrial de FCC da Unidade de Negócios Refinaria Landulpho Alves (RLAM) da Petrobras. Esses resultados estão nas Seções 5.3 e 5.4 (modelos global e individual, respectivamente).

5.1 Modelagem Global – Dados da SIX

5.1.1 Análise Exploratória dos Dados

Antes de iniciar as modelagens com redes neurais deve-se verificar se o conjunto de dados possui amostras (pontos experimentais) com comportamento anômalo (*outliers*), que podem representar valores muito baixo ou muito alto de alguma variável ou mesmo valores faltantes. Para isto, foi usada a PCA para a matriz **X** das variáveis independentes.

Como descrito no Capítulo 4, os dados foram pré-processados com o autoescalamento devido às variareis serem diferentes. Foram retiradas 7 amostras que possuíam lacunas em algumas variáveis, reduzindo o conjunto de dados para 84 amostras.

A análise dos componentes principais foi feita com as 84 amostras da matriz **X** e os resultados obtidos estão descritos na Tabela 5.1.

Na Tabela 5.1 observa-se que 5 componentes principais seriam necessários para a descrição do sistema devido ao valor da porcentagem de variância descrita (acumulada) que

foi de 98,43%,. A Figura 5.1 mostra o gráfico das componentes principais em função da variância acumulada. Nesta figura observa-se que há contribuições significativas na construção do modelo de PCA até a quinta componente principal. Já a sexta e sétima componentes principais dão pequenas contribuições na construção do modelo de PCA e estão associadas a ruídos nas medidas experimentais.

Tabela 5.1: Porcentagem de variância descrita pelas componentes principais para a matriz de dados **X** (84 amostras e 7 variáveis).

Componente	Autovalor (λ)	Variância Explicada (%)	Variância Acumulada (%)
CP1	2,7224	38,8909	38,8909
CP2	1,6968	24,2398	63,1307
CP3	1,0737	15,3392	78,4699
CP4	0,8247	11,7808	90,2507
CP5	0,5727	8,1807	98,4315
CP6	0,0958	1,3681	99,7996
CP7	0,0140	0,2004	100,0000

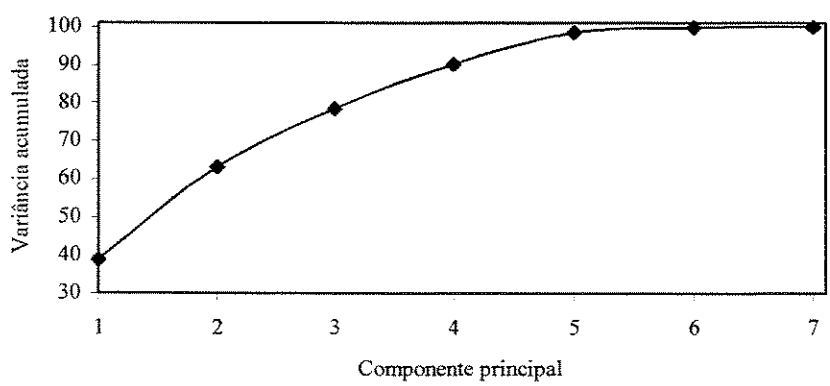


Figura 5.1: Gráfico das componentes principais em função da variância acumulada para os dados da SIX com 84 amostras.

Como o número necessário de componentes principais para descrever o conjunto de dados é cinco, foram colocados em gráficos os escores das componentes principais para revelar possíveis amostras cujo comportamento seja muito diferente do conjunto restante e os pesos das componentes principais para mostrar quais variáveis originais têm maior importância na combinação linear de cada componente principal. Em todos os gráficos a quinta componente principal não foi colocada por apresentar a menor porcentagem de variância descrita.

A Figura 5.2 mostra o gráfico dos pesos para as primeiras quatro componentes principais. Observa-se que todas as variáveis apresentam efeitos consideráveis nos resultados pois apresentam valores altos para os pesos. Como exemplo, a variável temperatura de reação (TRX) possui um valor baixo do peso na primeira componente principal (38,89% de variância explicada), no entanto, apresenta valores altos nas demais componentes (que explicam 51,36% do sistema restante). Resultados semelhantes podem ser observados para as demais variáveis e podem ser visualizados na Tabela 5.2.

Tabela 5.2: Pesos de cada variável nas cinco componentes principais usadas.

variável	PC1	PC2	PC3	PC4	PC5
VCA	-0,4978	-0,3598	0,02714	-0,0166	0,3076
TCA	0,4126	-0,5297	0,2102	-0,0673	0,0621
H	0,4334	-0,4919	0,2482	-0,0611	0,0555
TRX	-0,0566	0,2119	0,7310	0,6384	0,0953
TCER	-0,3575	-0,3619	0,0995	0,1026	-0,8463
VvL	-0,4893	-0,3476	0,0269	-0,0048	0,4127
VvD	-0,1537	0,2264	0,5903	-0,7572	-0,0537

Ainda na Figura 5.2, pode-se observar que em todos os gráficos as variáveis vazão de carga (VCA) e vazão do vapor de dispersão (VvL) estão intimamente relacionadas e que pode-se utilizar apenas uma delas.

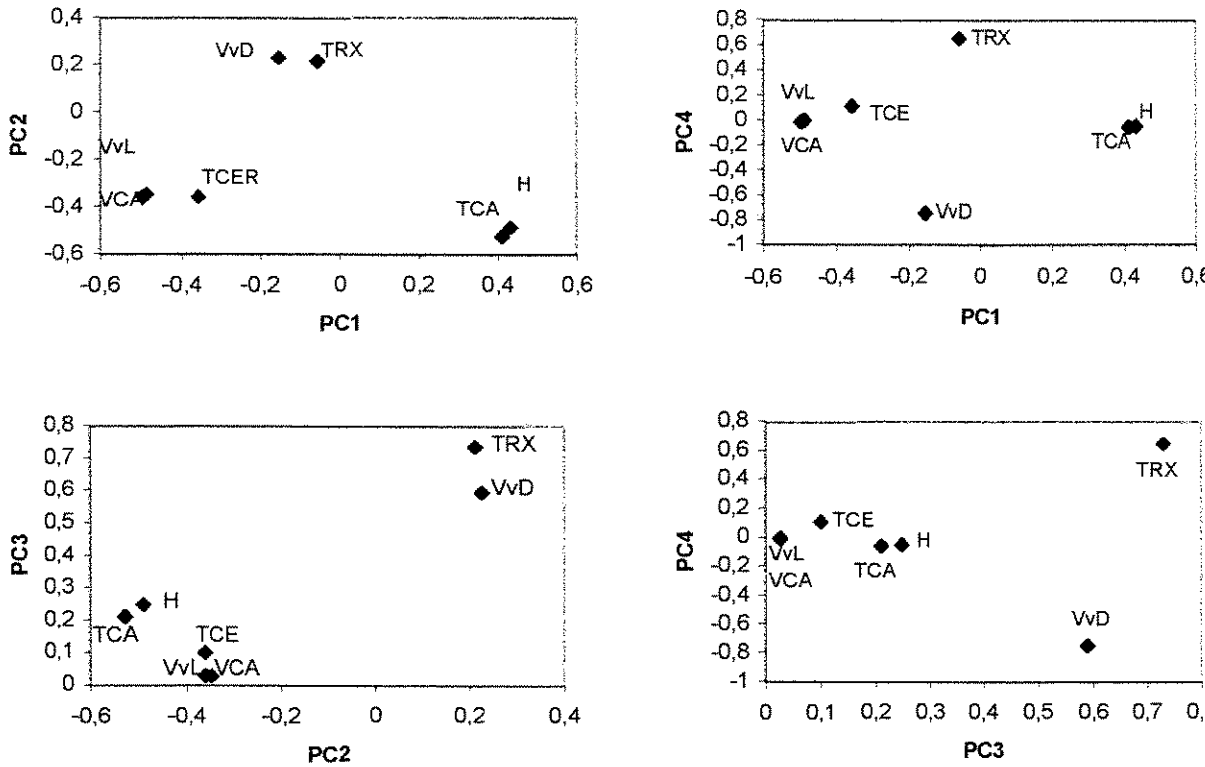


Figura 5.2: Gráfico dos pesos em PC1, PC2, PC3 e PC4.

Como o conjunto de dados não é muito grande, estas variáveis foram analisadas e verificou-se que as variações que ocorrem em uma das variáveis, VCA por exemplo, também eram acompanhadas por variações em outra variável (no caso de VvL). A PCA, através dos gráficos dos pesos, mostrou que estas variáveis são correlacionáveis e portanto uma delas pode ser dispensada. Foi escolhida a variável vazão de carga (VCA) para as próximas etapas do estudo.

A Figura 5.3 mostra o gráfico dos escores para as quatro componentes principais no modelo. Observa-se a formação de grupos de amostras, os quais estão diretamente relacionados a posição de injeção da carga (ou elevação do *riser*), H. Estes grupos estão bem nítidos nos gráficos PC1 *versus* PC2 e PC3 *versus* PC4. Pode-se observar também que as amostras 30 e 31 estão muito distantes dos agrupamentos formados, o que pode ser um indicativo de possíveis *outliers*.

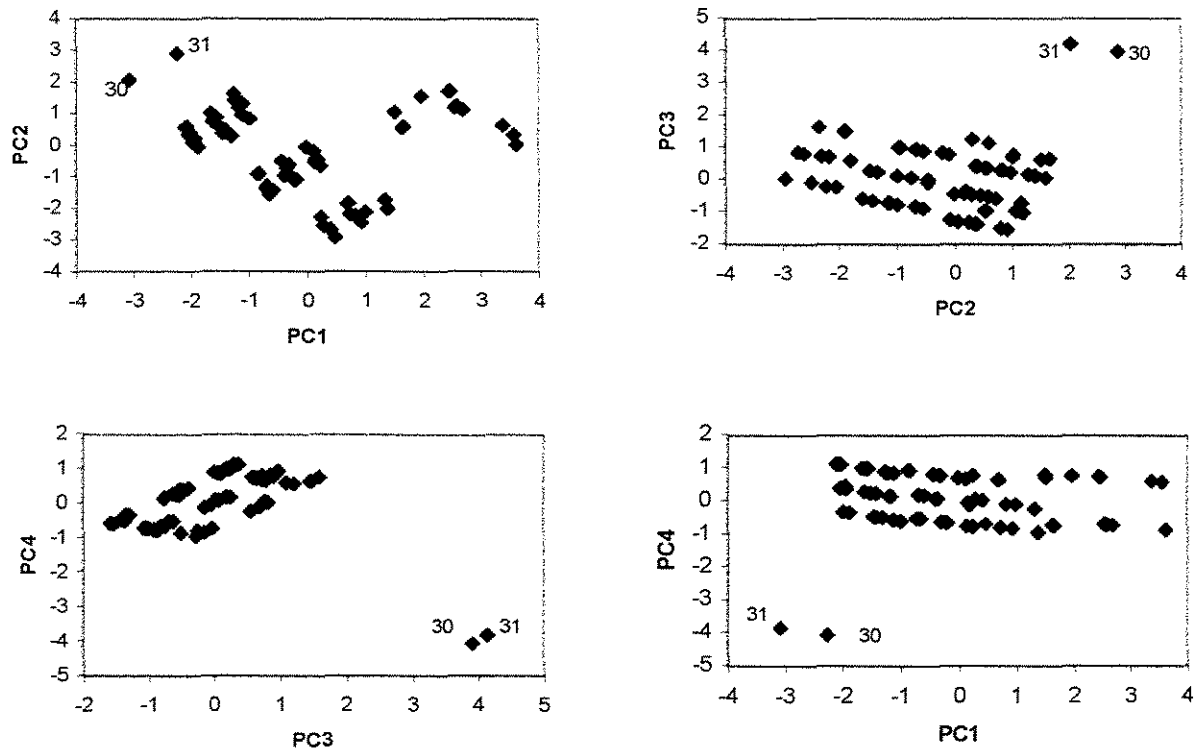


Figura 5.3: Gráfico dos escores em PC1, PC2, PC3 e PC4.

Para confirmar se tais amostras são *outliers*, foi usada a estatística T^2 de Hotelling. A Figura 5.4 mostra que as amostras estão fora do modelo de PCA com 5 componentes principais. No entanto, tais amostras foram analisadas para saber o porquê delas estarem tão distantes das demais. As amostras 30 e 31 apresentaram medidas muito diferentes para a variável vazão do vapor de dispersão (VvD), pois as demais 82 amostras tiveram um valor igual para esta variável, e as amostras 30 e 31 foram modificadas a níveis muito acima das restantes.

Um problema surge com o descarte das amostras 30 e 31: a remoção da variável VvD. Com a retirada das amostras 30 e 31 a variável VvD torna-se uma constante e não poderá ser usada no desenvolvimento dos modelos empíricos. Assim, foi desenvolvido um novo modelo de PCA com a retirada das variáveis VvL (por ser altamente correlacionável com a variável VCA) e VvD (por causa da retirada das amostras 30 e 31).

Desta forma, o novo modelo de PCA irá usar 5 variáveis independentes: VCA (vazão de carga – variável 1), TCA (temperatura da carga – variável 2), H (posição de injeção da carga – variável 3), TRX (temperatura do reator – variável 4) e TCER (temperatura da fase densa – variável 5).

Novamente os dados foram pré-processados com o autoescalamento devido à discrepância entre os valores das variáveis e o conjunto de dados possui agora 82 amostras. A análise dos componentes principais foi feita com as 82 amostras da matriz **X** e os resultados obtidos estão descritos na Tabela 5.3.

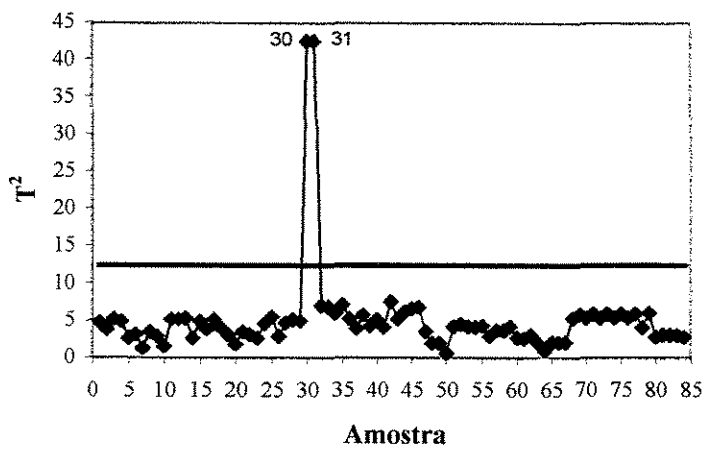


Figura 5.4: Gráfico de T^2 em função das 84 amostras.

Tabela 5.3: Porcentagem de variância descrita pelas componentes principais para a matriz de dados **X** modificada (82 amostras e 5 variáveis).

Componente	**Autovalor (λ)**	**Variância Explicada (%)**	**Variância Acumulada (%)**
CP1	2,1811	43,6214	43,6214
CP2	1,3919	27,8387	71,4600
CP3	0,9878	19,7562	91,2162
CP4	0,4245	8,4896	99,7058
CP5	0,0147	0,2942	100,0000

Na Tabela 5.3 observa-se que três componentes principais já descrevem bem o sistema, pois descrevem 91,22% da variância acumulada. Se considerarmos a quarta componente principal tem-se 99,71% da variância, porém possivelmente podemos estar adicionando ruídos ao conjunto de dados, pois a inclusão desta componente fará com que o modelo descreve quase todo o conjunto original dos dados incluindo os ruídos existentes. A Figura 5.5 mostra a variância acumulada em função das componentes principais.

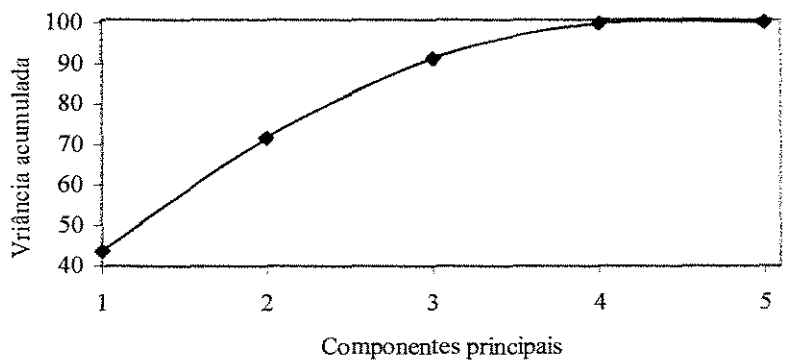


Figura 5.5: Gráfico das componentes principais em função da variância acumulada para os dados da SIX com 82 amostras.

A Figura 5.6 mostra o gráfico dos pesos em função das variáveis consideradas. Observa-se que todas as variáveis apresentam valores de pesos altos, o que sugere que todas elas foram importantes na construção do modelo de PCA.

A Figura 5.7 mostra os gráfico dos escores para as 82 amostras consideradas. Nesta figura observa-se que não há amostras tão distantes do conjunto como ocorreu com as amostras 30 e 31 que foram descartadas nesta etapa. Isto fica evidenciado pela Figura 5.8 que mostra o gráfico de T^2 de Hotelling versus o número de amostras, em que pode se verificar que todas as amostras estão abaixo do limite de confiança (95%).

Os resultados anteriores indicaram que o uso de menos variáveis é necessário para obter modelos mais confiáveis. No entanto, é de interesse para a Petrobras a construção de um modelo que utilize todas as variáveis consideradas nos experimentos. Assim sendo, foram desenvolvidos modelos empíricos que utilizem todas as 7 variáveis independentes (conjunto 1) e modelos empíricos que 5 variáveis independentes (conjunto 2), de acordo com os resultados obtidos pela análise exploratória.

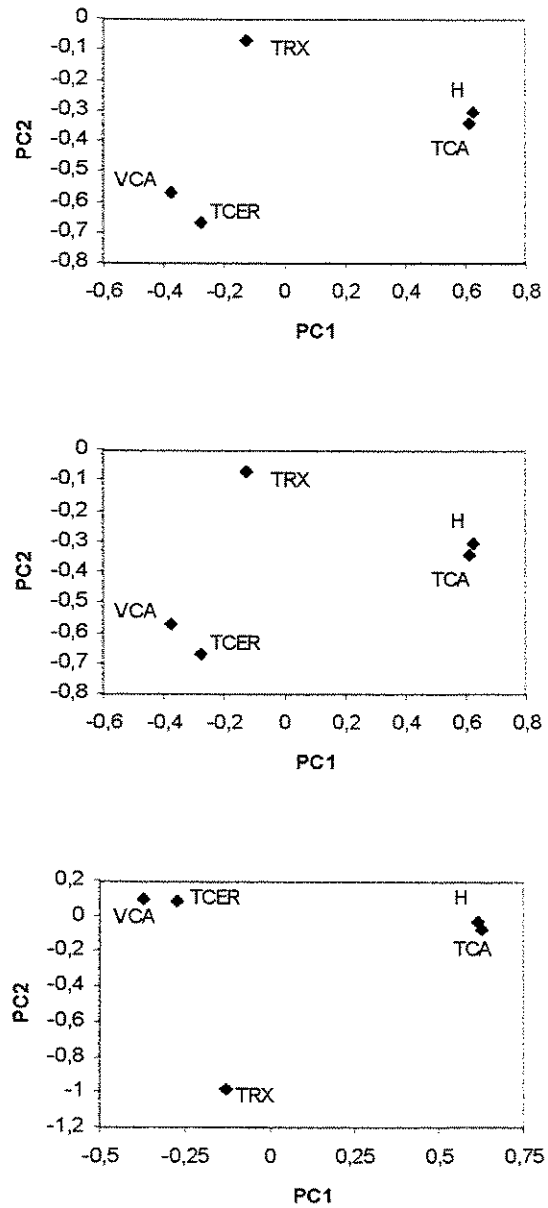


Figura 5.6: Gráficos dos pesos em PC1, PC2 e PC3.

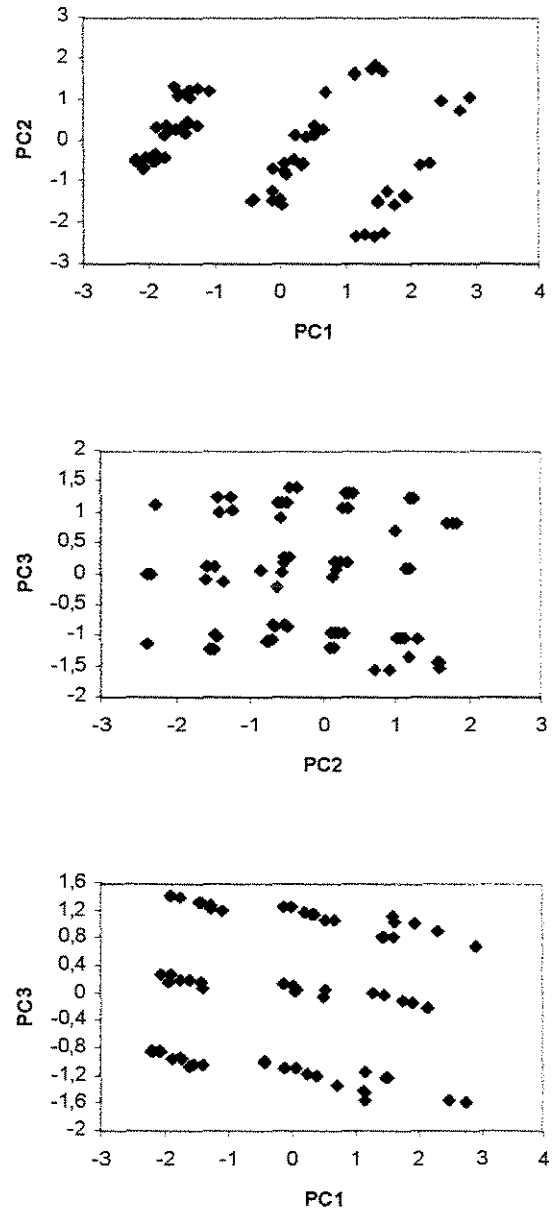


Figura 5.7: Gráficos dos escores em PC1, PC2 e PC3.

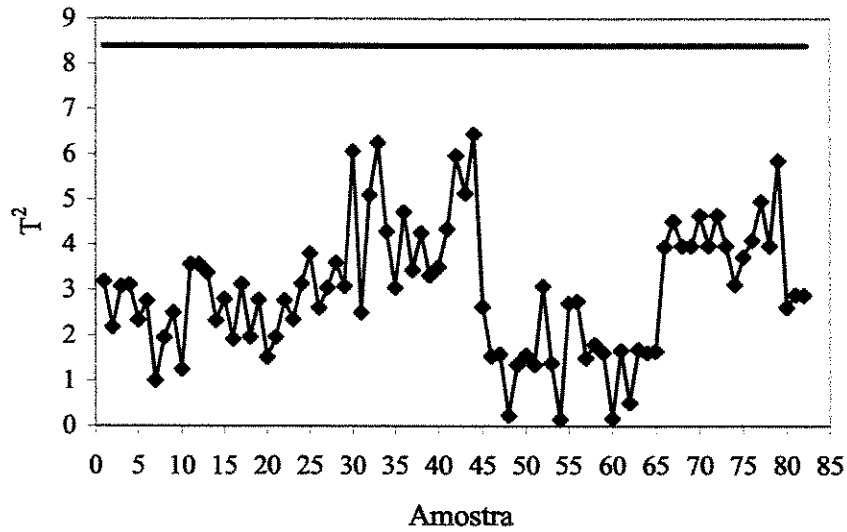


Figura 5.8: Gráfico de T^2 em função das 82 amostras.

5.1.2 Modelos de Regressão Linear (PCR e PLS)

A seguir, são descritos os resultados obtidos pelos modelos de regressão usando PLS e PCR para os dois conjuntos considerados aqui. Em cada conjunto os dados são separados em dois grupos: treinamento e validação. Como definido nos Capítulo 3 e 4, o critério de avaliação utilizado será o erro quadrático médio para o conjunto de validação (*RMSE*). As amostras usadas na etapa de validação são as mesmas para os dois conjuntos e para os modelos de regressão.

5.1.2.1 PCR

5.1.2.1.1 Conjunto 1

Utilizando a validação cruzada para determinar o número de componentes principais para ser usado no modelo de PCR, observou-se que 5 componentes principais são suficientes para descrever o conjunto 1 (7 variáveis e 84 amostras). A Figura 5.9 mostra o gráfico de *RMSE versus* as variáveis (que estão na forma de números) para a etapa de treinamento (obtenção dos parâmetros de ajuste) para cada componente principal estudada, e a Figura 5.10 mostra o gráfico de *RMSE versus* as variáveis para a etapa de validação para cada componente principal estudada. Na Figura 5.10 observa-se que a partir da quinta componente principal não há muita variação nos valores de *RMSE* para todas as variáveis. O uso de 5 componentes principais para este conjunto de dados descreve 98,41% da variância da matriz **X**.

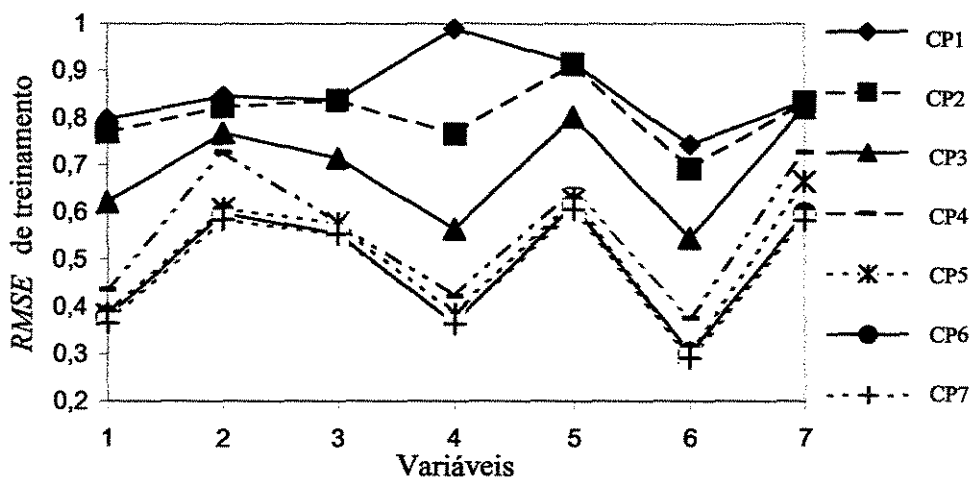


Figura 5.9: Gráfico de *RMSE* (treinamento) em função das variáveis para diferentes números de componentes principais – PCR para o conjunto 1.

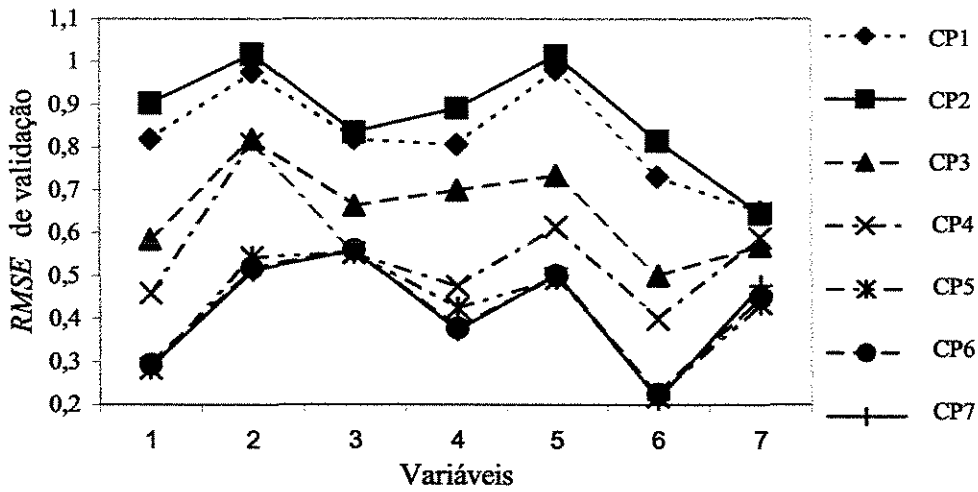


Figura 5.10: Gráfico de *RMSE* (validação) em função das variáveis para diferentes números de componentes principais – PCR para o conjunto 1.

5.1.2.1.2 Conjunto 2

Seguindo o mesmo procedimento do conjunto 1, foi determinado o número de componentes principais para ser usado no modelo de PCR para o conjunto 2; observou-se que 4 componentes principais são suficientes para descrever o conjunto 2 (7 variáveis e 84 amostras). A Figura 5.11 mostra o gráfico de *RMSE versus* as variáveis para etapa de treinamento e a Figura 5.12 mostra o gráfico de *RMSE versus* as variáveis para etapa de validação. Na Figura 5.12 observa-se que na quarta componente principal não há muita variação nos valores de *RMSE* em comparação com a quinta componente principal. O uso de 4 componentes principais para este conjunto de dados descreve 99,75% da variância da matriz **X**.

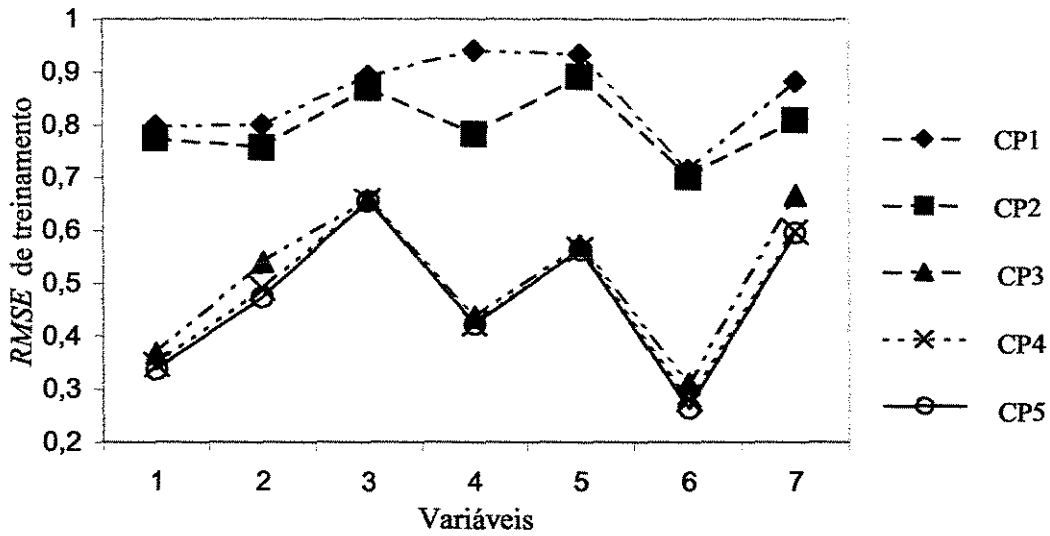


Figura 5.11: Gráfico de *RMSE* (treinamento) em função das variáveis para diferentes números de componentes principais – PCR para o conjunto 2.

5.1.2.1.3 Discussão sobre a PCR

A Tabela 5.4 mostra os resultados obtidos pelos dois conjuntos para a predição das amostras usadas na etapa de validação. Como se pode observar, o conjunto 2 (com 5 variáveis) apresentou os menores erros de predição (*RMSE*) o que, por sua vez, apresentou coeficientes de correlação próximo da unidade para a maioria das variáveis estudadas, que pode ser visto na Tabela 5.5.

Apesar do conjunto 2 apresentar menores erros de predição em relação ao conjunto 1, observa-se que para algumas variáveis a PCR apresenta coeficientes de correlação abaixo de 0,9 (principalmente para as variáveis de interesse: RGAS e RGLP) o que pode ser um indicativo de que, mesmo com a aplicação da técnica de redução de variáveis, o comportamento altamente não-linear do processo permanece de forma ainda acentuada.

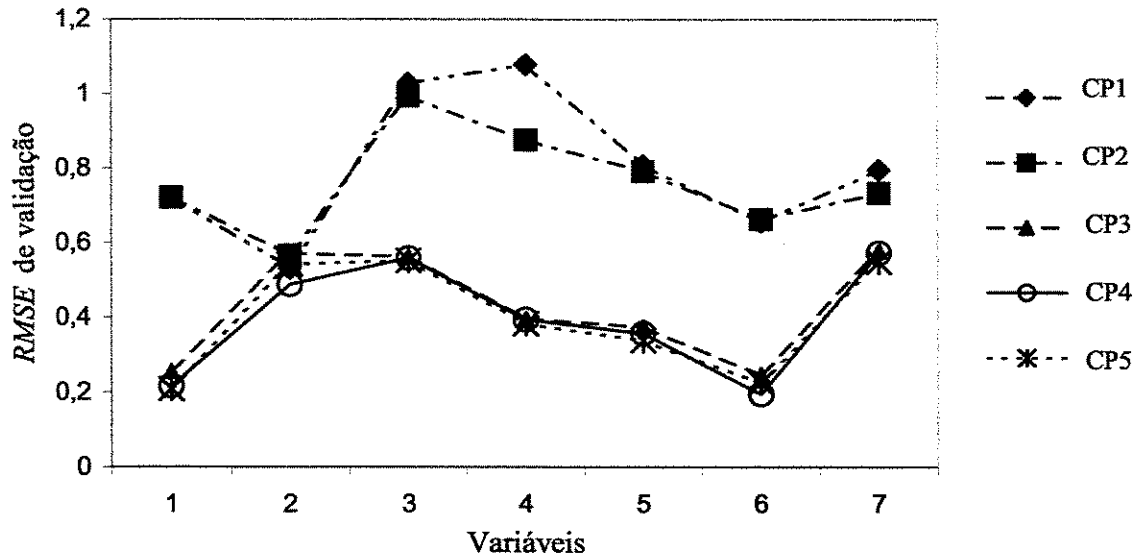


Figura 5.12: Gráfico de *RMSE* (validação) em função das variáveis para diferentes números de componentes principais – PCR para o conjunto 2.

Tabela 5.4: *RMSE* na etapa de validação para cada variável estudada em função dos modelos de PCR estudados.

	RMSE			
	PCR1		PCR2	
Variável	Treinamento	Validação	Treinamento	Validação
CONV	0,3843	0,2842	0,3467	0,2166
RGAS	0,6070	0,5431	0,4892	0,4870
RGLP	0,5766	0,5539	0,6547	0,5579
RGC	0,3839	0,4247	0,4233	0,3937
RLCO	0,6270	0,4933	0,5638	0,3574
ROD	0,3033	0,2192	0,2805	0,1945
RCOQ	0,6650	0,4340	0,5976	0,5717

Tabela 5.5: Análise de regressão entre as respostas dos modelos de PCR e as saídas desejadas para cada variável de saída.

	PCR1			PCR2		
Variável	*R*	*a*	*b*	*R*	*a*	*b*
CONV	0,9696	0,0281	0,9027	0,9786	0,0556	1,0089
RGAS	0,9027	0,0313	0,6620	0,8369	0,0255	0,9719
RGLP	0,8574	0,0396	0,8324	0,8986	0,1049	0,6461
RGC	0,8781	0,0885	0,8741	0,9436	0,0643	0,8109
RLCO	0,9080	-0,0120	0,7386	0,9214	-0,0477	0,8903
ROD	0,9807	-0,0277	0,9458	0,9841	-0,0574	1,0259
RCOQ	0,9109	-0,0976	0,7508	0,8263	0,1124	0,6347

5.1.2.2 PLS

5.1.2.2.1 Conjunto 1

A Tabela 5.6 mostra a variância capturada por cada variável latente (VL) no modelo de PLS para o conjunto 1. Observa-se que a partir da quarta variável latente a variância acumulada para a matriz **Y** não aumenta muito com a adição de outras variáveis latentes. Porém descreve 85,17% da variância da matriz **X**, ou seja, é possível que informações importantes estejam sendo perdidas por descrever uma quantidade relativamente baixa de informação. Assim sendo, foi realizada a validação cruzada para se determinar o número ótimo de variáveis latentes.

Através da validação cruzada, tem-se que 5 variáveis latentes descrevem bem os dados pois forneceram os menores valores de *RMSE* para a maioria das variáveis. A Figura 5.13 mostra o gráfico de *RMSE versus* as variáveis (que estão na forma de números) para a etapa de treinamento (obtenção dos parâmetros de ajuste) para cada variável latente estudada, e a Figura 5.14 mostra o gráfico de *RMSE versus* as variáveis para a etapa de validação, para cada variável latente estudada. Nestas figuras observa-se que o modelo com 5 variáveis latentes apresenta os menores valores de *RMSE* para a maioria das variáveis. O modelo de PLS com 5 variáveis latentes descreve 97,79% da variância em **X** e 74,50% da variância em **Y**. Pode-se observar que o modelo com 5 variáveis latentes teve um valor baixo para variância acumulada em Y, isto poderia ser contornado adicionando mais variáveis latentes ao modelo (mais que as sete mostradas na Tabela 5.6). No entanto, a adição de mais componentes irá descrever quase toda a variação em Y mas irá fornecer um número maior de variáveis latentes em relação as variáveis originais. Optou-se neste trabalho por utilizar sempre o número de variáveis latentes menor que o número de variáveis originais.

Tabela 5.6: Porcentagem de variância capturada pelo modelo PLS – Conjunto1.

	Matriz **X**		Matriz **Y**	
Variável Latente	Variância Explicada (%)	Variância Acumulada (%)	Variância Explicada (%)	Variância Acumulada (%)
VL1	36,08	36,08	47,37	47,37
VL2	21,25	57,33	15,24	62,61
VL3	18,09	75,42	6,36	68,97
VL4	9,75	85,17	3,06	72,03
VL5	12,62	97,79	0,80	72,83
VL6	1,97	99,76	1,67	74,50
VL7	0,24	100,00	0,77	75,27

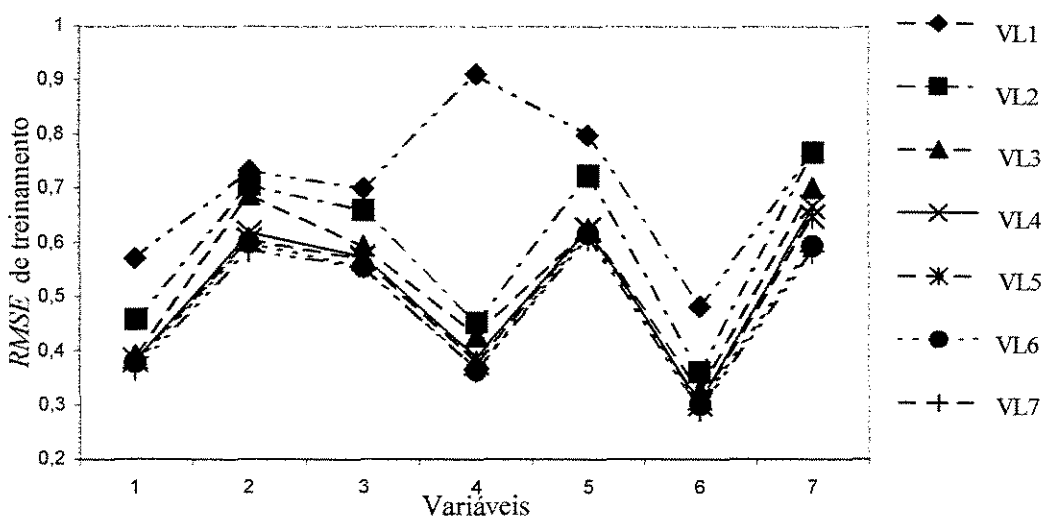


Figura 5.13: Gráfico de *RMSE* (treinamento) em função das variáveis para diferentes números de componentes principais – PLS para o conjunto 1.

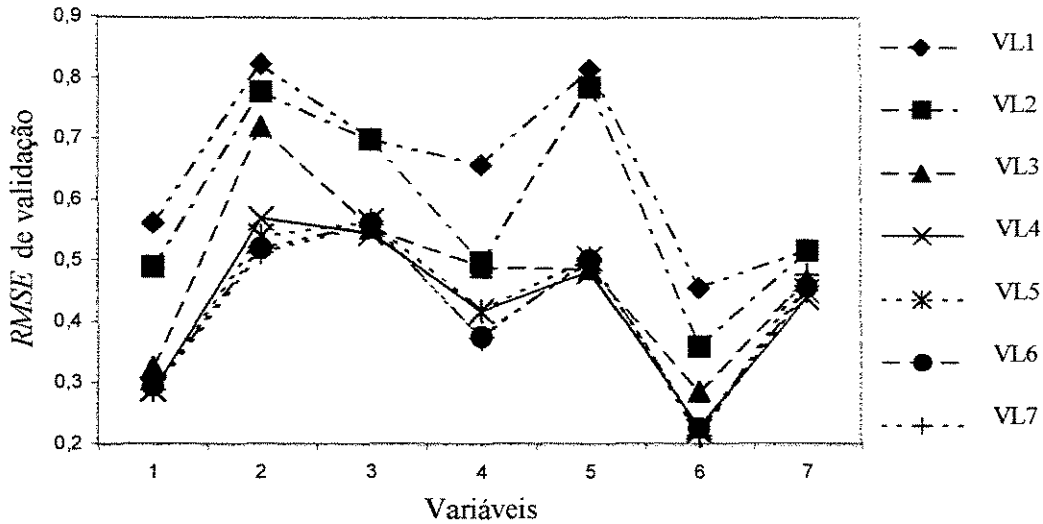


Figura 5.14: Gráfico de *RMSE* (validação) em função das variáveis para diferentes números de componentes principais – PLS para o conjunto 1.

5.1.2.2.2 Conjunto 2

A Tabela 5.7 mostra a variância capturada por cada variável latente (VL) no modelo de PLS para o conjunto 2. Através da validação cruzada, foi determinado o número ótimo de variáveis latentes. Aqui, o número de variáveis latentes foi quatro e descreveu 99,76% da variância em **X** e 75,00% da variância em **Y**. As Figuras 5.15 e 5.16 mostram os gráficos de *RMSE* em função das variáveis do processo para cada variável latente considerada para as etapas de treinamento e validação, respectivamente. Nestas figuras observa-se que o modelo com 4 variáveis latentes fornece o mesmo resultado que o modelo com 5 variáveis latentes.

Tabela 5.7: Porcentagem de variância capturada pelo modelo PLS – Conjunto 2.

	Matriz **X**		Matriz **Y**	
Variável Latente	Variância Explicada (%)	Variância Acumulada (%)	Variância Explicada (%)	Variância Acumulada (%)
VL1	39,01	39,01	54,26	54,26
VL2	23,45	62,46	11,70	65,96
VL3	27,18	89,65	7,59	73,55
VL4	10,11	99,76	1,45	75,00
VL5	0,24	100,00	0,49	75,49

5.1.2.2.3 Discussão sobre a PLS

A Tabela 5.8 mostra os resultados obtidos pelos dois conjuntos para a predição das amostras usadas na etapa de validação. Como pode-se observar, o conjunto 2 (com 5 variáveis) apresentou os menores erros de predição (*RMSE*) no modelo de PLS com 4 variáveis latentes em relação ao modelo de PLS com 5 variáveis latentes do conjunto 1 (7 variáveis). Os coeficientes de correlação próximos da unidade para a maioria das variáveis estudadas confirmam que o modelo de PLS com 4 variáveis latentes (conjunto 2) representa melhor o conjunto de dados em relação ao outro modelo de PLS com 5 variáveis latentes.

Os resultados obtidos pelo PLS são semelhantes aos obtidos pelo PCR, ou seja, o conjunto 2 apresentou menores erros de predição em relação ao conjunto 1 e, para algumas variáveis, o PLS apresenta coeficientes de correlação abaixo de 0,9, principalmente para as variáveis de interesse: RGAS e RGLP, indicando que essas variáveis (que são de maior interesse) ainda têm um caráter não-linear mesmo reduzindo o número de variáveis no estudo e aplicando as técnicas da estatística multivariada.

Os resultados obtidos pelo PLS são quase iguais aos que foram obtidos pelo PCR, com erros de predição nos conjuntos de treinamento e de validação um pouco menores.

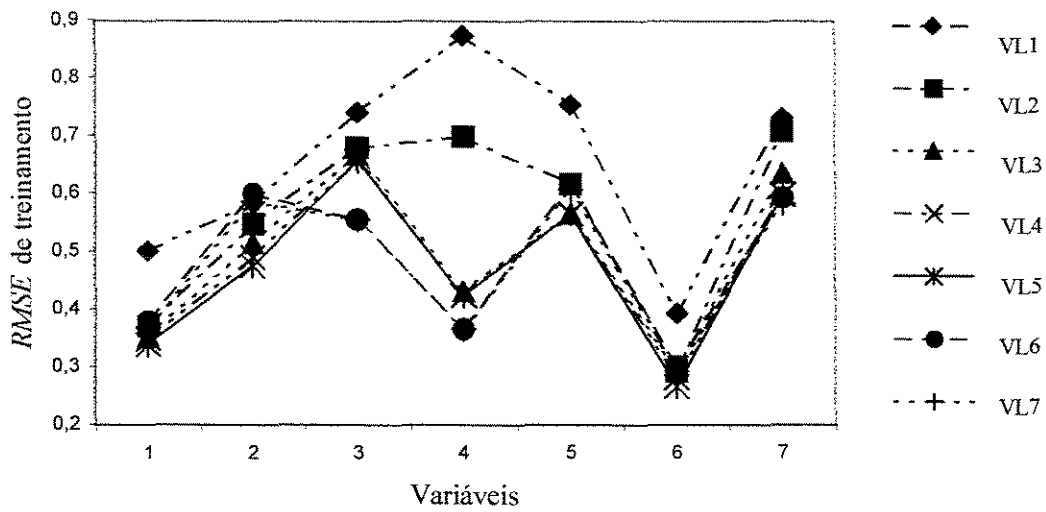


Figura 5.15: Gráfico de *RMSE* (treinamento) em função das variáveis para diferentes números de componentes principais – PLS para o conjunto 2.

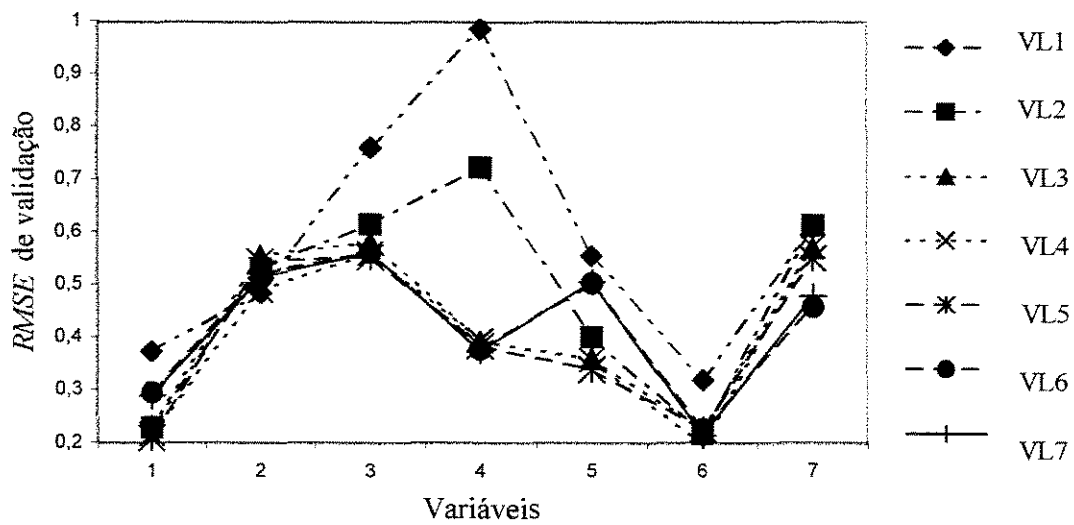


Figura 5.16: Gráfico de *RMSE* (validação) em função das variáveis para diferentes números de componentes principais – PLS para o conjunto 2.

Tabela 5.8: *RMSE* na etapa de validação para cada variável estudada em função dos modelos de PLS estudados.

	RMSE			
	PLS1*		**PLS2***	
Variável	**Treinamento**	**Validação**	**Treinamento**	**Validação**
CONV	0,3812	0,2871	0,3464	0,2157
RGAS	0,6166	0,5677	0,4886	0,4874
RGLP	0,5730	0,5436	0,6547	0,5579
RGC	0,3870	0,4162	0,4233	0,3936
RLCO	0,6236	0,4811	0,5637	0,3564
ROD	0,3021	0,2260	0,2799	0,1942
RCOQ	0,6655	0,4388	0,5974	0,5702

* PLS1 e PLS2 são os modelos PLS para os conjuntos 1 e 2, respectivamente.

Tabela 5.9: Análise de regressão entre as respostas dos modelos de PLS e as saídas desejadas para cada variável de saída.

	PLS1*			**PLS2***		
Variável	***R***	***a***	***B***	***R***	***a***	***b***
CONV	0,9685	0,0280	0,9028	0,9788	0,0555	1,0098
RGAS	0,9033	0,0854	0,6686	0,8369	0,0254	0,9728
RGLP	0,8536	0,0378	0,8383	0,8986	0,1050	0,6461
RGC	0,8835	0,0888	0,8855	0,9436	0,0643	0,8110
RLCO	0,90319	-0,0116	0,7374	0,9219	-0,0475	0,8913
ROD	0,9804	-0,0278	0,9460	0,9842	-0,0574	1,0266
RCOQ	0,9015	-0,1051	0,7655	0,8274	0,1123	0,6362

* PLS1 e PLS2 são os modelos PLS para os conjuntos 1 e 2, respectivamente.

5.1.3 Redes Neurais com Parada Antecipada

Como descrito no Capítulo 3, foram testadas três diferentes metodologias para o treinamento das redes MLP. Inicialmente foi utilizada a técnica da parada antecipada para um determinado número de neurônios na camada oculta. O número de neurônios na camada oculta variou de 1 a 20. Para cada número de neurônio considerado, foram feitos 30 treinamentos por causa dos valores iniciais dos pesos e bias da rede terem sido fornecidos de forma randômica. Os parâmetros da rede que forneceram o menor erro de predição no conjunto de validação são guardados para posterior comparação com os erros gerados por outras configurações com diferentes números de neurônios ocultos. Também foi utilizada a técnica de regularização Bayesiana para tentar melhorar a generalização das redes neurais e fazer comparações com os resultados obtidos pela técnica de validação cruzada com parada antecipada.

5.1.3.1 Conjunto 1

Os resultados obtidos para as diferentes metodologias de treinamento das redes MLP para o conjunto 1 (84 amostras e 7 variáveis) estão mostradas nas Figuras 5.17 – 5.19. Nesta figuras pode-se observar que a raiz quadrada dos erros quadráticos médios (*RMSE*) do conjunto de treinamento diminui à medida que o número de neurônios na camada oculta aumenta. Já o *RMSE* para o conjunto de validação diminui até atingir um valor mínimo e depois aumenta à medida que são acrescentados neurônios na camada oculta.

A razão para este comportamento está relacionada ao número de parâmetros da rede (pesos e biases) que cresce à medida que adicionamos neurônios na camada oculta, mas como o número de amostras não altera a rede torna-se especialista no conjunto de treinamento apresentado perdendo sua capacidade de generalização (HAYKIN, 2001).

A Tabela 5.10 mostra os resultados dos métodos LM, BFGS e SCG para os valores de *RMSE* global para treinamento e validação do conjunto 1. Observa-se que o método BFGS apresentou os menores valores de *RMSE* para treinamento e validação no conjunto como um todo (todas as amostras e todas as variáveis).

Tabela 5.10: *RMSE* globais para os métodos LM, BFGS e SCG – Conjunto 1.

		RMSE	
Método	Número de neurônios	Treinamento	Validação
LM	5	0,4551	0,4263
BFGS	8	0,4528	0,4070
SCG	12	0,4790	0,4182

No entanto, pela Tabela 5.11, observa-se que o método SCG apresentou uma maior capacidade de previsão para cinco variáveis de saída (CONV, RGAS, RGLP, RGC e ROD), pois, apresentou os menores valores de *RMSE* para essas variáveis. Já o método BFGS teve os menores *RMSE* para as variáveis RLCO e RCOQ. A justificativa para essas diferenças reside na variável RCOQ pois o método SCG teve um valor de *RMSE* muito alto (0,5403) enquanto o método BFGS teve um valor menor (0,4297). Esta diferença é considerável e tem influência no valor de *RMSE* global.

Assim sendo, o critério do *RMSE* global foi usado para definir a topologia da rede (número de neurônios ocultos) para um determinado método. Para comparação entre os métodos foi considerado o *RMSE* das variáveis de saída. O método que apresentar os menores *RMSE* na etapa de validação para a maioria das variáveis será o método escolhido.

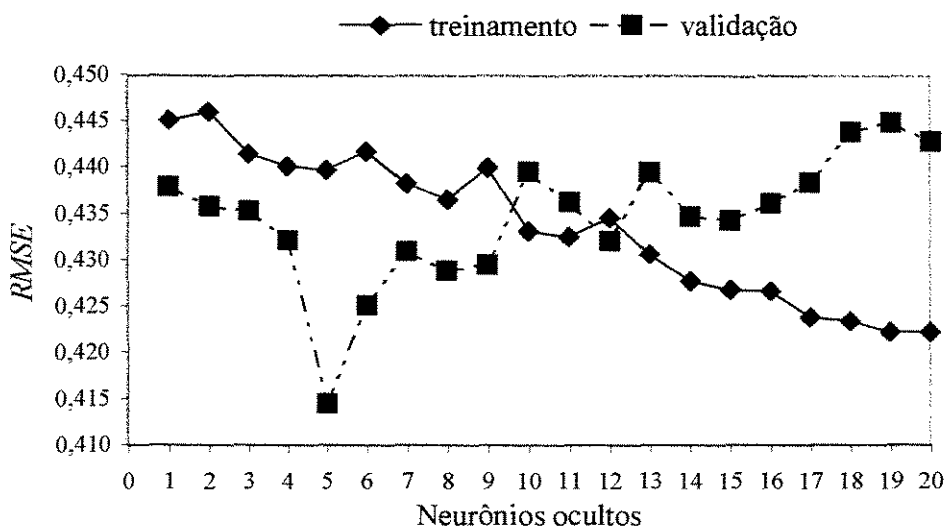


Figura 5.17: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método LM – conjunto 1.

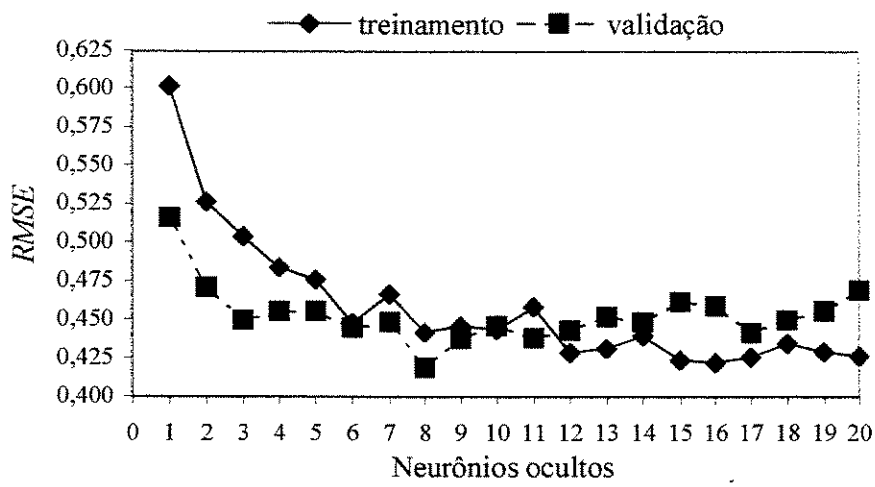


Figura 5.18: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método BFGS – conjunto 1.

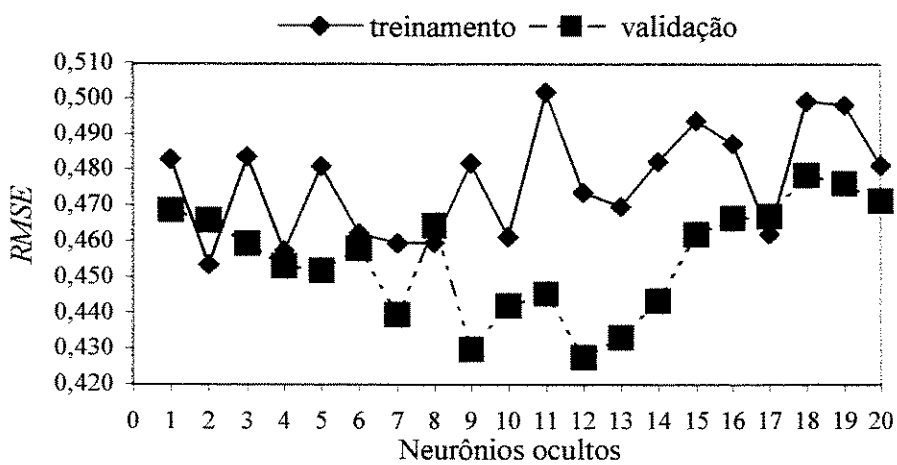


Figura 5.19: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método SCG – conjunto 1.

Tabela 5.11: Melhores resultados para o *RMSE* de validação para as variáveis dependentes nos diferentes métodos considerados – conjunto 1.

	RMSE					
	LM		BFGS		SCG	
Variável	Treinamento	Validação	Treinamento	Validação	Treinamento	Validação
CONV	0,3327	0,3006	0,3363	0,2599	0,3516	0,2531
RGAS	0,4791	0,5167	0,4839	0,4984	0,5013	0,4605
RGLP	0,5209	0,4995	0,4831	0,4994	0,5313	0,4919
RGC	0,3219	0,4005	0,3799	0,4178	0,3784	0,3635
RLCO	0,5773	0,4873	0,5638	0,4440	0,5882	0,4977
ROD	0,2469	0,2319	0,2495	0,1944	0,2679	0,1857
RCOQ	0,5834	0,4624	0,5748	0,4297	0,6229	0,5403

Uma outra forma para se verificar que o método SCG forneceu os melhores resultados de previsão é realizar uma análise de regressão entre as saídas da rede e as saídas desejadas. A Tabela 5.12 mostra os valores *R* (coeficiente de correlação), de *a* (coeficiente linear) e *b* (coeficiente angular) obtidos para cada variável de saída considerando os diferentes métodos estudados. Pode-se observar que o método SCG apresentou, para o conjunto 1, valores de *R* mais próximos da unidade para a maioria das variáveis.

Tabela 5.12: Análise de regressão entre as respostas das redes e as saídas desejadas para cada variável de saída – conjunto 1.

	LM			BFGS			SCG		
variável	*R*	*a*	*b*	*R*	*a*	*b*	*R*	*a*	*b*
CONV	0,9649	-0,0466	0,8715	0,9730	-0,0087	0,8979	0,9768	-0,0464	0,8835
RGAS	0,9126	-0,1118	0,6610	0,9563	-0,1261	0,6953	0,9566	-0,1813	0,6850
RGLP	0,8708	0,0347	0,7904	0,8870	0,1034	0,8941	0,8875	0,1033	0,8983
RGC	0,8919	-0,0051	0,8651	0,9113	0,1734	0,9569	0,9153	0,0843	0,8484
RLCO	0,9054	-0,0128	0,7116	0,9211	-0,0144	0,7568	0,8959	-0,0399	0,7424
ROD	0,9799	0,0775	0,9232	0,9846	0,0018	0,9252	0,9815	0,0928	0,9099
RCOQ	0,8845	-0,0074	0,7759	0,9034	-0,0490	0,8417	0,8568	-0,1310	0,8232

5.1.3.2 Conjunto 2

Foram obtidos os valores de *RMSE* global para as etapas de treinamento e validação dos métodos LM, BFGS e SCG seguindo o mesmo procedimento da seção 5.3.1. As Figuras 5.20 – 5.22 mostram resultados semelhantes aos obtidos pelo conjunto 1, ou seja, à medida que aumentamos o número de neurônios na camada oculta, o erro de treinamento diminui e o erro de validação diminui até atingir um mínimo e depois aumenta continuamente.

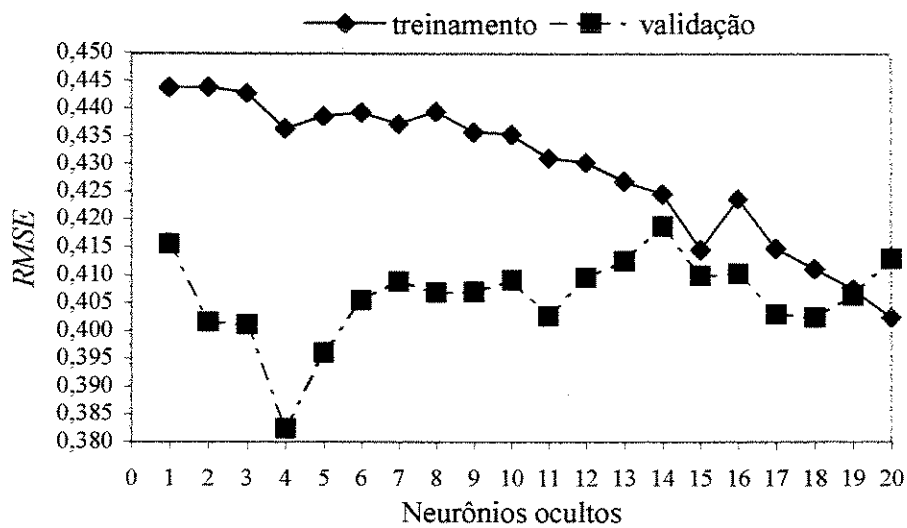


Figura 5.20: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método LM – conjunto 2.

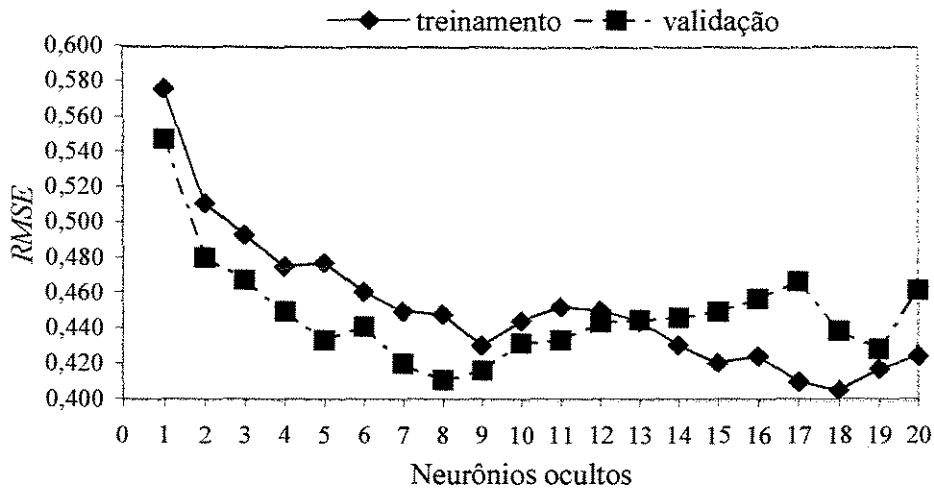


Figura 5.21: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método BFGS – conjunto 2.

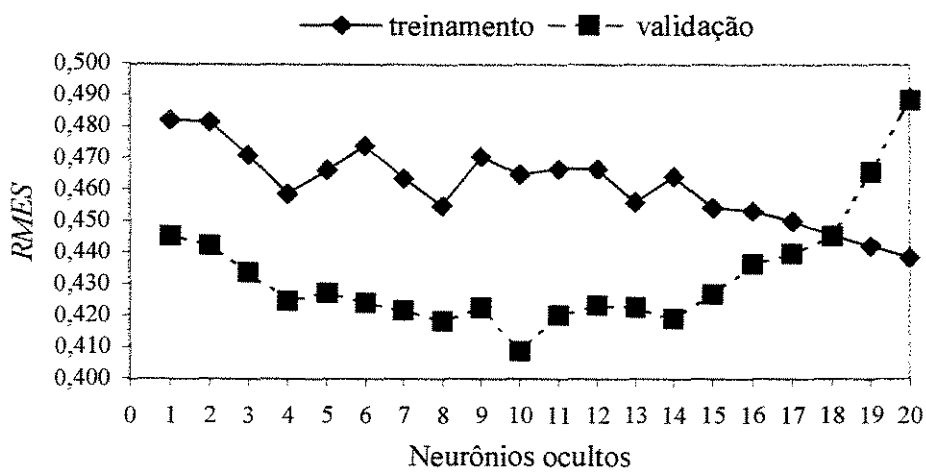


Figura 5.22: *RMSE* para treinamento e validação em função do numero de neurônios ocultos para o método SCG – conjunto 2.

A Tabela 5.13 mostra os resultados dos métodos LM, BFGS e SCG para os valores de *RMSE* (global) para treinamento e validação do conjunto 2 (82 amostras e 5 variáveis). Observa-se que o método LM apresentou os menores valores de *RMSE* nas etapas de treinamento e validação. Este método também apresentou a menor topologia em relação aos demais.

Tabela 5.13: *RMSE* globais para os métodos LM, BFGS e SCG – Conjunto 2.

		RMSE	
Método	Número de neurônios	Treinamento	Validação
LM	4	0,4412	0,3823
BFGS	8	0,4561	0,4058
SCG	10	0,4655	0,4078

A Tabela 5.14 mostra os valores de *RMSE* para cada variáveis de saída. Nesta tabela também observa-se que o método LM apresentou os menores *RMSE* para quatro variáveis de saída (CONV, RGAS, RLCO, RCOQ), enquanto o método BFGS apresentou *RMSE* menor para duas variáveis (RGC e ROD) e o método SCG teve o menor *RMSE* para a variável RGLP.

Tabela 5.14: Melhores resultados para o *RMSE* de validação para as variáveis dependentes nos diferentes métodos considerados – Conjunto 2.

	RMSE					
	LM		BFGS		SCG	
Variável	Treinamento	Validação	Treinamento	Validação	Treinamento	Validação
CONV	0,3144	0,2029	0,3148	0,2152	0,3167	0,2395
RGAS	0,4443	0,4141	0,4850	0,4814	0,4399	0,4307
RGLP	0,5202	0,4558	0,5819	0,5244	0,5520	0,4111
RGC	0,3880	0,4298	0,3641	0,3705	0,3999	0,4476
RLCO	0,5356	0,3513	0,5447	0,3754	0,5655	0,4102
ROD	0,2267	0,1928	0,2486	0,1603	0,2566	0,2027
RCOQ	0,5542	0,4960	0,5405	0,5437	0,6105	0,5853

Também foi feita a análise de regressão para a etapa de validação dos métodos estudados. A Tabela 5.15 mostra os resultados da análise de regressão para o conjunto 2. Como era de se esperar, o método LM obteve os melhores ajustes para a maioria das variáveis estudadas pois os coeficientes estão próximos dos valores ideais.

Tabela 5.15: Análise de regressão entre as respostas das redes e as saídas desejadas para cada variável de saída – conjunto 2.

	LM			BFGS			SCG		
variável	*R*	*a*	*b*	*R*	*a*	*b*	*R*	*a*	*b*
CONV	0,9774	0,0099	0,9996	0,9740	-0,0162	1,0067	0,9715	0,0118	0,9950
RGAS	0,8804	-0,0035	0,9469	0,8431	-0,0447	0,9200	0,8753	-0,0271	0,9604
RGLP	0,9156	0,0149	0,7708	0,8868	0,0373	0,7097	0,9391	0,0625	0,7627
RGC	0,9291	0,0461	0,7673	0,9137	-0,0155	0,8315	0,9286	0,0351	0,7075
RLCO	0,9421	-0,0145	1,0370	0,9193	-0,0452	0,9225	0,9041	0,0059	0,9168
ROD	0,9810	-0,0016	0,9798	0,9888	0,0464	1,0215	0,9796	-0,0187	0,9919
RCOQ	0,8730	0,1072	0,7449	0,8564	0,1888	0,7041	0,8125	0,0595	0,6238

5.1.3.3 Discussão sobre redes neurais artificiais

Os resultados mostraram que a metodologia utilizada para modelar o FCC usando redes neurais com a técnica da parada antecipada para obter um modelo que forneça todas as variáveis de saída apresentou algumas dificuldades quanto à obtenção do melhor método que deve ser utilizado.

Nos dois conjuntos estudados observou-se que não houve um método que fosse capaz de dar bons resultados para todas as variáveis de saída. Para o conjunto 1, o método BFGS forneceu os menores valores de *RMSE* global para treinamento e validação. No entanto, o método SCG forneceu os menores *RMSE* para 5 variáveis de saída, evidenciando que este método foi o que melhor modelou o processo, mesmo fornecendo predições não tão boas para as variáveis RLCO e RCOQ. Esta rede teve 12 neurônios na camada oculta.

Para o conjunto 2, o método LM forneceu os melhores resultados tanto o global como para a maioria das variáveis de saída, porque forneceu os menores valores de *RMSE* na etapa de treinamento e validação. Esta rede teve 5 neurônios na camada oculta.

Como o conjunto de validação foi o mesmo para os dois conjuntos estudados pode-se fazer uma comparação entre os resultados obtidos. Observa-se (ver Tabelas 5.11 e 5.14) que o conjunto 2 com o método LM teve os menores *RMSE* na etapa de validação para 5

variáveis de saída, enquanto o conjunto 1 teve *RMSE* menor para as variáveis RGC e ROD. Esta diferença pode ser justificada pela remoção das duas amostras (30 e 31) com comportamento anômalo e das variáveis VvD e VvL. Em outras palavras, a retirada das amostras e das variáveis ajudou na performance do modelo.

As Figuras 5.23 – 5.29 mostram os desvios percentuais absolutos obtidos pelo valor absoluto da diferença entre o valor desejado e o valor calculado pelo modelo de redes neurais dividido pelo valor desejado. O modelo considerado é o de rede neural usando o método LM com validação cruzada para o conjunto 2, pois este, no geral, obteve o melhor resultado (menores *RMSE* na etapa de validação). Observa-se que, para a conversão, 98,78% das amostras estão com desvio (absoluto) inferior a 10%. Já para os rendimentos os resultados foram os seguintes: 97,56% para gasolina, 85,37% para GLP, 84,15% para gás combustível, 96,34% para LCO, 93,90% para óleo decantado e 95,12% para coque.

A seguir são mostradas outras duas modelagens feitas para os conjuntos 1 e 2 com a finalidade de obter modelos com maior poder de predição. A primeira é o uso de redes neurais usando o método LM com a técnica de regularização Bayesiana. A segunda é o uso do PLS com redes neurais artificiais nas suas relações internas (ou PLS não linear).

5.1.4 Redes Neurais com Regularização

Os resultados a seguir estão resumidos pois as discussões pertinentes sobre a topologia da rede e erros de treinamento e validação já foram feitas na seção 5.1.3.

Para o conjunto 1 a rede neural com o menor *RMSE* na predição do conjunto de validação possui nove neurônios na camada oculta. Já para o conjunto 2 a rede neural teve dez neurônios na camada oculta. A Tabela 5.13 mostra os valores de *RMSE* para cada variável de saída no treinamento e validação da rede. Pode-se observar que o conjunto 2 novamente apresentou os menores *RMSE* nas etapas de treinamento e validação.

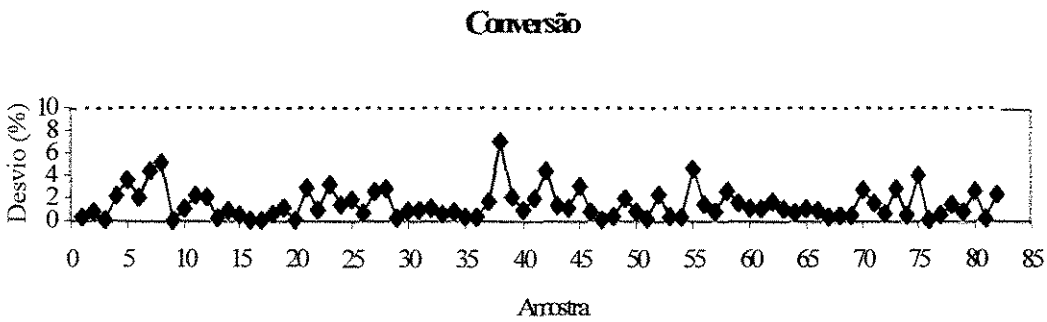


Figura 5.23: Desvios para a variável de saída conversão em função das amostras para o método LM com parada antecipada.

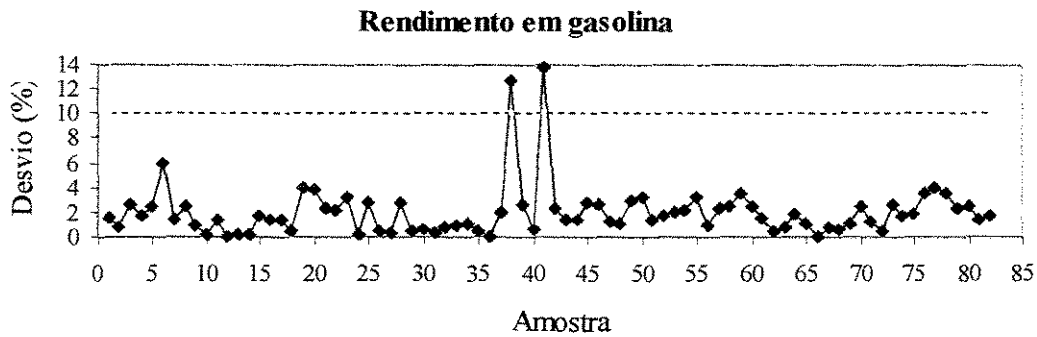


Figura 5.24: Desvios para a variável de saída rendimento em gasolina em função das amostras para o método LM com parada antecipada.

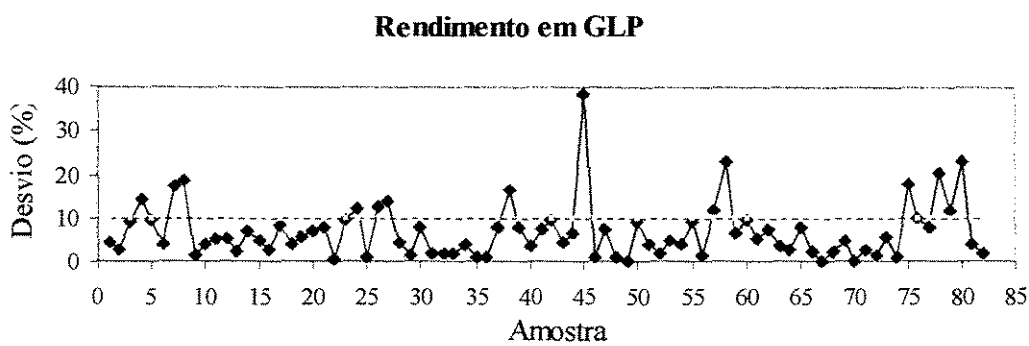


Figura 5.25: Desvios para a variável de saída rendimento em GLP em função das amostras para o método LM com parada antecipada.

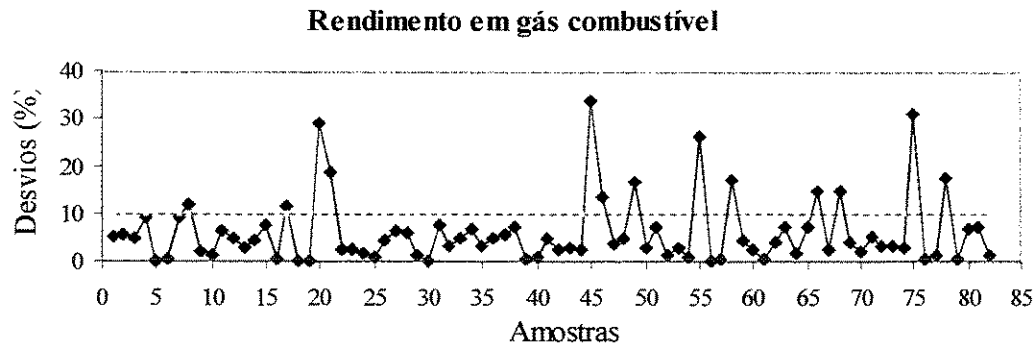


Figura 5.26: Desvios para a variável de saída rendimento em gás combustível em função das amostras para o método LM com parada antecipada.

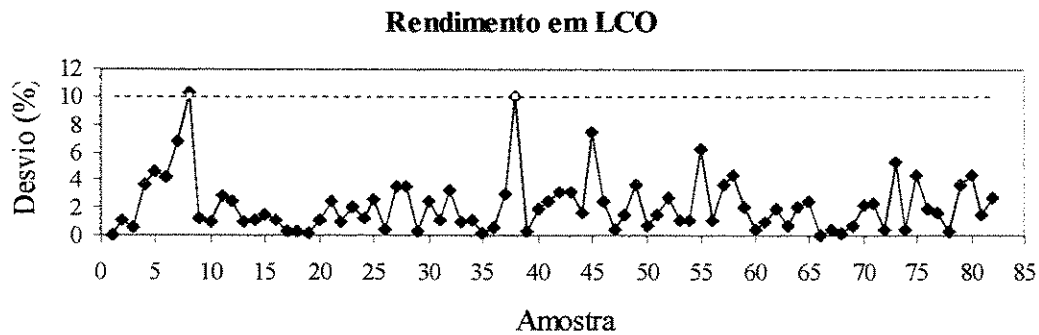


Figura 5.27: Desvios para a variável de saída rendimento em LCO em função das amostras para o método LM com parada antecipada.

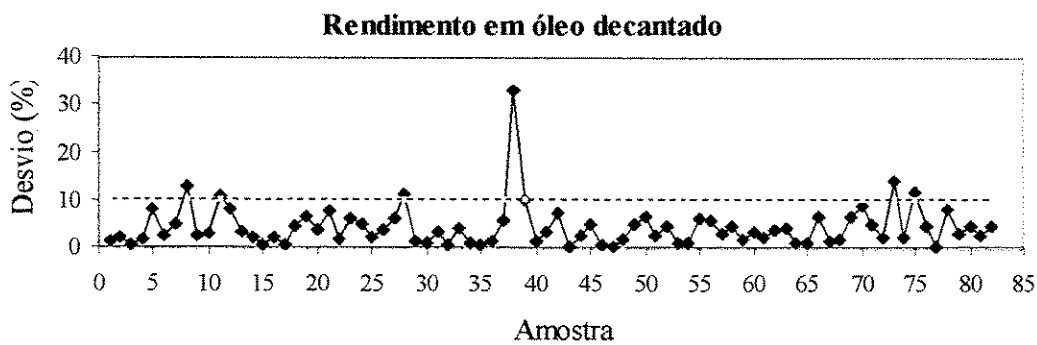


Figura 5.28: Desvios para a variável de saída rendimento em óleo decantado em função das amostras para o método LM com parada antecipada.

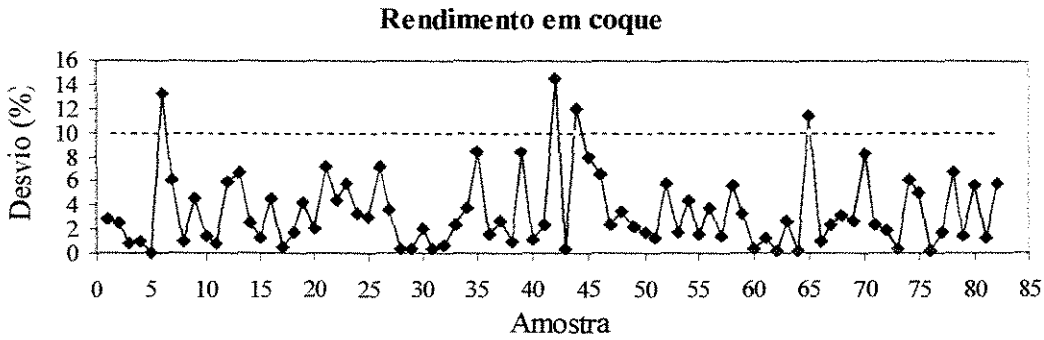


Figura 5.29: Desvios para a variável de saída rendimento em coque em função das amostras para o método LM com parada antecipada.

Tabela 5.13: *RMSE* dos conjuntos 1 e 2 nas etapas de treinamento e validação para o método LM com regularização.

	RMSE			
	Conjunto 1		Conjunto 2	
Variável	Treinamento	Validação	Treinamento	Validação
CONV	0,1147	0,1031	0,0985	0,0659
RGAS	0,2315	0,3093	0,177474	0,1936
RGLP	0,2534	0,2237	0,295732	0,3502
RGC	0,1086	0,1151	0,13153	0,1885
RLCO	0,3539	0,2648	0,269985	0,1683
ROD	0,0615	0,0599	0,051138	0,0379
RCOQ	0,3176	0,2121	0,296009	0,3005

Fazendo uma comparação com o melhor resultado obtido usando a técnica de parada antecipada (método LM), observa-se que o método LM com regularização apresentou *RMSE* menores para todas as variáveis de saída tanto na etapa de treinamento quanto na de validação. Para quantificar essa melhoria, as Figuras 5.30 – 5.36 mostram os desvios percentuais obtidos pelo modelo aqui considerado. Observa-se que, para a conversão, 98,78% das amostras estão com desvio (absoluto) inferior a 10%. Já para os rendimentos os resultados foram os seguintes: 100,00% para gasolina, 85,37% para GLP,

84,15% para gás combustível, 96,34% para LCO, 93,90% para óleo decantado e 95,12% para coque. Resumindo, o modelo LM com regularização forneceu os mesmos resultados em relação ao mesmo modelo mas com a técnica de parada antecipada, exceto para o rendimento em gasolina onde os resultados foram superiores (todas as amostras tiveram desvios absolutos abaixo de 10%).

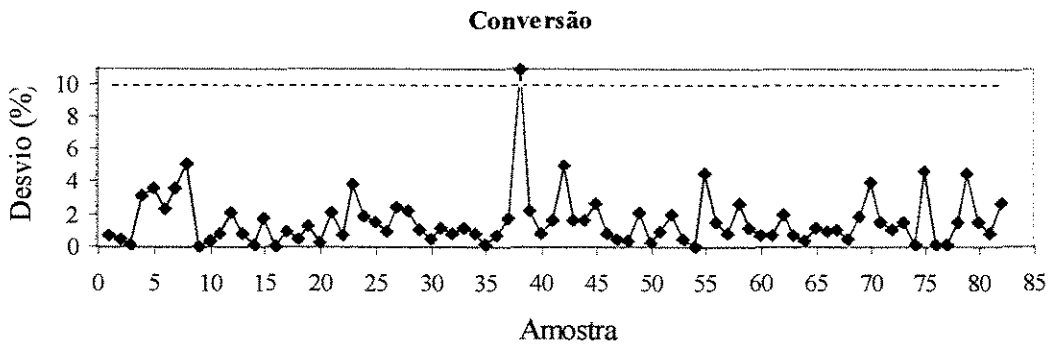


Figura 5.30: Desvios para a variável de saída conversão em função das amostras para o método LM com regularização Bayesiana.

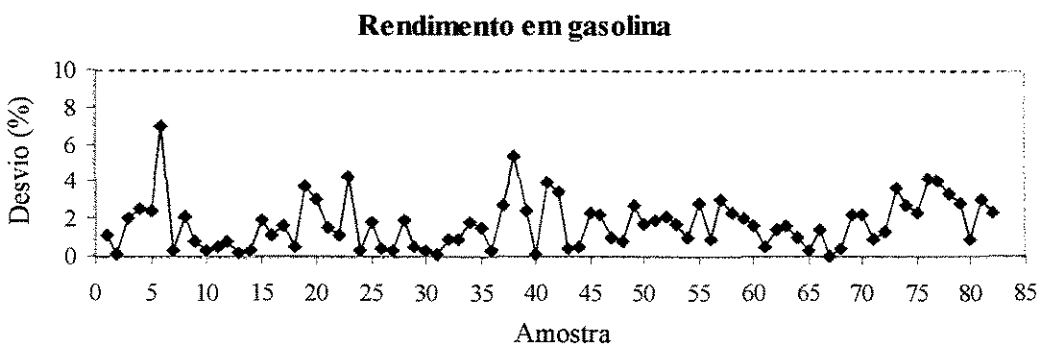


Figura 5.31: Desvios para a variável de saída rendimento em gasolina em função das amostras para o método LM com regularização Bayesiana.

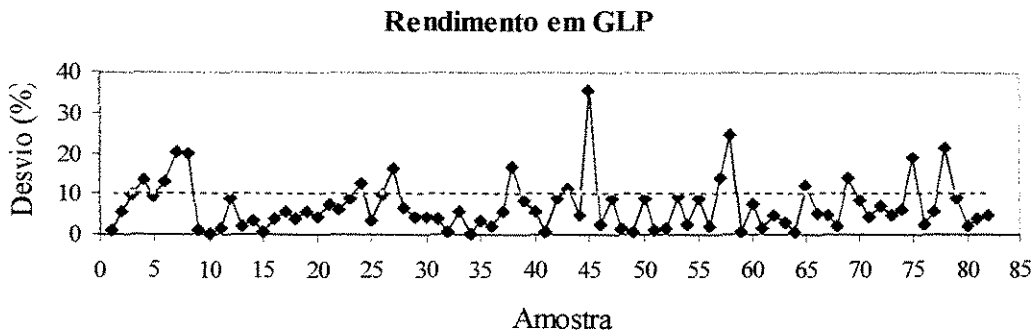


Figura 5.32: Desvios para a variável de saída rendimento em GLP em função das amostras para o método LM com regularização Bayesiana.

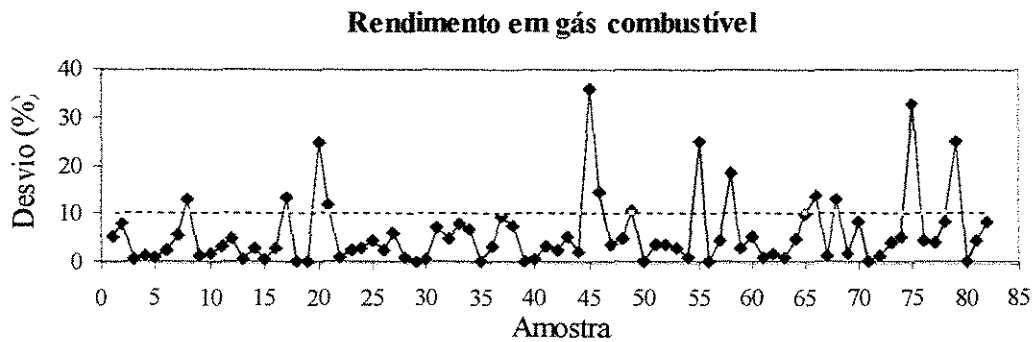


Figura 5.33: Desvios para a variável de saída rendimento em gás combustível em função das amostras para o método LM com regularização Bayesiana.

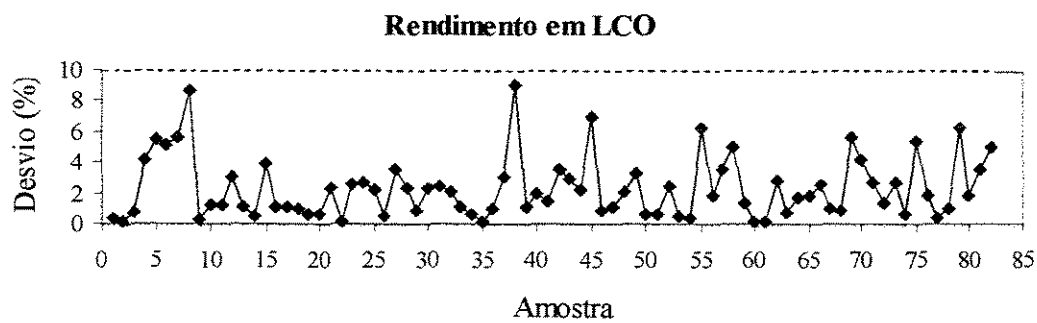


Figura 5.34: Desvios para a variável de saída rendimento em LCO em função das amostras para o método LM com regularização Bayesiana.

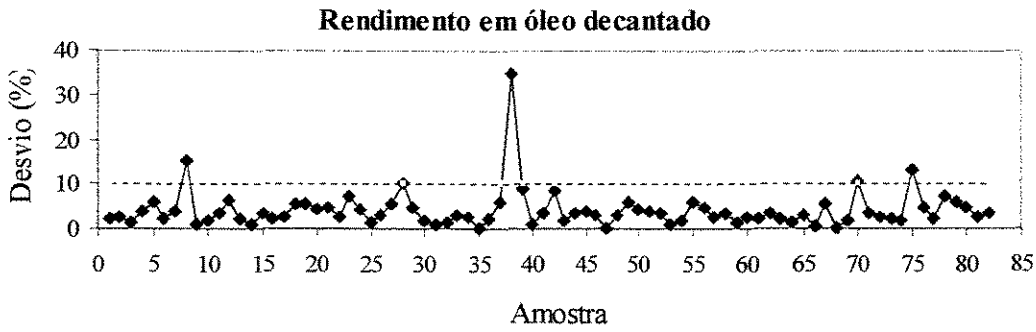


Figura 5.35: Desvios para a variável de saída rendimento em óleo decantado em função das amostras para o método LM com regularização Bayesiana.

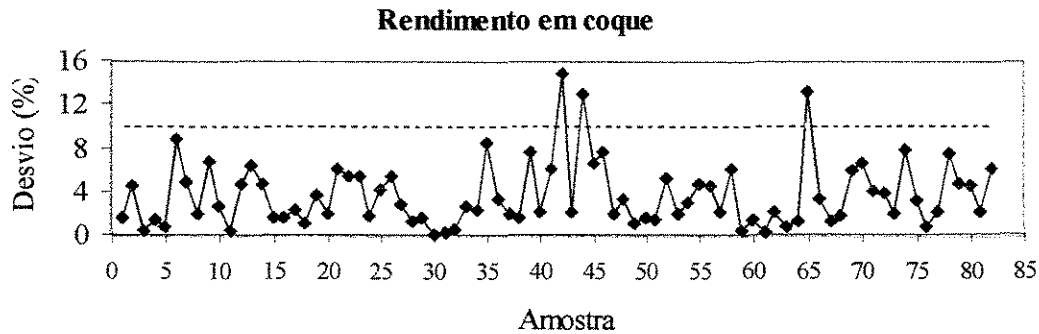


Figura 5.36: Desvios para a variável de saída rendimento em coque em função das amostras para o método LM com regularização Bayesiana.

5.1.5 PLS com Redes Neurais Artificiais

Assim como na Seção 5.1.5, aqui também foram apresentados os resultados de forma resumida visto que toda a discussão pertinente sobre a técnica de PLS podem ser encontrados na Seção 5.1.2.2.

O modelo de PLS com redes neurais para o conjunto 1 utilizou 5 variáveis latentes (VL), pois, através da validação cruzada, foi determinado este número de variáveis latentes. Já para o conjunto 2 foi encontrado que quatro variáveis latentes fornecia o melhor modelo. A Tabela 5.14 mostra os valores de *RMSE* para cada variável de saída no treinamento e

validação do modelo aqui considerado. Observa-se que este tipo de modelo não trouxe melhoria na modelagem dos dados da SIX pois os valores de *RMSE* estão acima dos que foram obtidos pelo método LM com regularização, o qual forneceu os melhores resultados para a modelagem global dos dados da SIX.

Tabela 5.14: *RMSE* dos conjuntos 1 e 2 nas etapas de treinamento e validação para o método PLS com redes neurais artificiais.

	RMSE			
	Conjunto 1		Conjunto 2	
Variável	Treinamento	Validação	Treinamento	Validação
CONV	0,3904	0,3298	0,3613	0,2042
RGAS	0,5951	0,6339	0,4999	0,4546
RGLP	0,5800	0,4891	0,6565	0,5763
RGC	0,3526	0,3712	0,4169	0,3995
RLCO	0,6348	0,5343	0,5856	0,3597
ROD	0,3002	0,243	0,2787	0,1783
RCOQ	0,6663	0,504	0,6051	0,577

5.2 Modelagem Individual – Dados da SIX

Os resultados obtidos na Seção 5.1 mostraram que o modelo global apresentou resultados satisfatórios para algumas variáveis de saída mas para outras o modelo não teve um bom poder de predição. Por isso, a proposta desta Seção é modelar cada variável de saída separadamente, ou seja, criar um modelo empírico para cada variável de saída.

O conjunto 2 é o grupo de dados que foi considerado nesta Seção, pois o conjunto 1 apresentou os maiores valores de *RMSE* para todos os modelos considerados visto que este conjunto possuía *outliers* e variáveis correlacionadas.

A seguir são apresentados os resultados (em termos de *RMSE*) para os seguintes modelos: Redes neurais com o método LM e regularização; PLS com redes neurais e PCR com redes neurais (chamado aqui de PCR-RNA).

5.2.1 Redes Neurais com Regularização

Seguindo o mesmo procedimento da Seção 5.1.4, os resultados obtidos para modelagem de cada variável de saída são apresentados na Tabela 5.15. Nesta tabela são encontrados os valores de *RMSE* para as etapas de treinamento e validação, bem como a topologia da rede obtida. Esses resultados serão comparados aos obtidos nas Seções a seguir.

Tabela 5.15: *RMSE* para o conjunto 2 nas etapas de treinamento e validação para o modelo de redes neurais usando o método LM com regularização – modelagem individual com os dados da SIX.

		RMSE	
Variável	Topologia	Treinamento	Validação
CONV	(5:7:1)	0,1167	0,0544
RGAS	(5:2:1)	0,3016	0,1688
RGLP	(5:10:1)	0,3116	0,3440
RGC	(5:10:1)	0,1350	0,1493
RLCO	(5:9:1)	0,3157	0,1413
ROD	(5:8:1)	0,0483	0,0302
RCOQ	(5:10:1)	0,3582	0,3292

5.2.2 PLS com Redes Neurais

Aqui também procurou seguir o procedimento descrito na Seção 5.1.5, os resultados obtidos para modelagem de cada variável de saída são apresentados na Tabela 5.16. Nesta tabela são encontrados os valores de *RMSE* para as etapas de treinamento e validação, bem como a topologia da rede obtida.

Tabela 5.16: *RMSE* para o conjunto 2 nas etapas de treinamento e validação para o método PLS com redes neurais artificiais – modelagem individual com os dados da SIX.

		RMSE	
Variável	Variáveis Latentes	Treinamento	Validação
CONV	4	0,3390	0,2230
RGAS	4	0,4895	0,4837
RGLP	4	0,6280	0,6042
RGC	3	0,4118	0,4165
RLCO	4	0,5387	0,3897
ROD	4	0,2837	0,1674
RCOQ	5	0,5966	0,5590

5.2.3 PCR com Redes Neurais

A abordagem proposta nesta Seção é utilizar a técnica de PCA para redução da dimensionalidade nas variáveis de entrada e em seguida fazer a modelagem de cada uma das variáveis de saída utilizando os escores obtidos pela técnica de PCA (representados pela matriz **T** na figura a seguir). A Figura 5.37 mostra o esquema da abordagem aplicada nesta Seção. Nela pode-se ver que após a PCA usamos os escores como entrada das redes que irão gerar as suas respectivas saídas.

Nesta modelagem, optou-se pelo método LM para treinamento da rede MLP visto que o mesmo apresenta boa performance de treinamento quando trabalha com redes de tamanho pequeno a moderado, ou seja, quando a rede possui uma quantidade de parâmetros (pesos e *bias*) pequenos (DEMUTH e BEALEM, 2001).

Os procedimentos tomados para treinamento e validação são os mesmos feitos na Seção 5.1.3, ou seja, utilizou-se a parada antecipada para o treinamento da rede, as funções de transferência são tangente hiperbólica na camada oculta e linear na camada de saída, o número de neurônios na camada oculta foi modificado de 1 até 15 e para cada número de neurônios na camada oculta foram feitos 30 treinamentos.

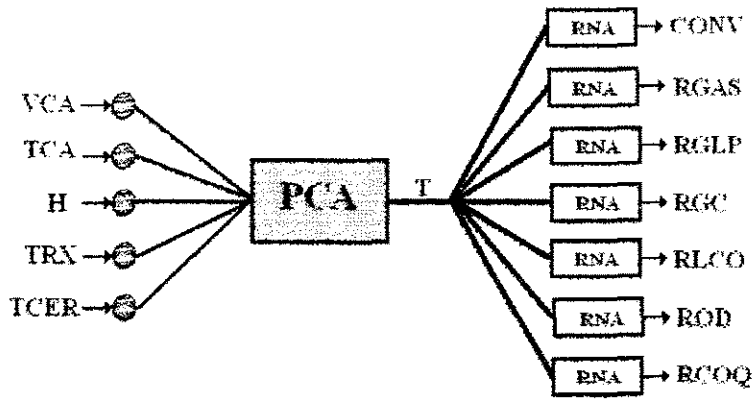


Figura 5.37: Esquema simplificado da modelagem de FCC utilizando redes neurais com PCA nos dados de entrada.

Na Seção 5.1.1 foi usada a análise dos componentes principais para fazer uma análise exploratória dos dados. Verificou-se (através da validação cruzada) que três componentes principais foram suficientes para descrever o sistema. Essas três componentes serão consideradas aqui e os escores obtidos pelo modelo de PCA foram usados aqui como entrada da rede.

A Tabela 5.17 mostra os resultados obtidos pelas redes para cada variável de saída. Na tabela encontra-se a topologia da rede que produziu o menor *RMSE* na etapa de validação e também na etapa de treinamento.

Tabela 5.17: *RMSE* para o conjunto 2 nas etapas de treinamento e validação para o método PCR-RNA – modelagem individual com os dados da SIX.

		RMSE	
Variável	Topologia	Treinamento	Validação
CONV	(3:7:1)	0,3647	0,2307
RGAS	(3:8:1)	0,4241	0,3458
RGLP	(3:4:1)	0,5678	0,5097
RGC	(3:3:1)	0,4979	0,3388
RLCO	(3:9:1)	0,5606	0,3402
ROD	(3:6:1)	0,2603	0,1738
RCOQ	(3:7:1)	0,2801	0,2781

5.2.4 Discussão sobre os modelos individuais

Comparando os resultados obtidos nas Tabela 5.15, 5.16 e 5.17 pode-se observar que o modelo de redes neurais usando o método LM com regulação (RNA-RB) forneceu os menores valores de *RMSE* para quase todas as variáveis de saída exceto para a variável rendimento em coque, na qual o modelo PCR-RNA forneceu o menor *RMSE* na validação.

As Figuras 5.38 – 5.44 mostram os gráficos dos desvios percentuais obtidos pelos modelos indicados na Tabela 5.18. Observa-se que, para a conversão, 100,00% das amostras estão com desvio (absoluto) inferior a 10%. Já para os rendimentos os resultados foram os seguintes: 100,00% para gasolina, 78,05% para GLP, 87,80% para gás combustível, 100,00% para LCO, 95,12% para óleo decantado e 96,34% para coque. Como era de se esperar, os resultados das modelagens individuais forneceram uma melhoria no poder de predição em comparação com o modelo global para quase todas as variáveis. A única exceção foi a variável RGLP que apresentou predições abaixo do que foi obtido pelo método LM com regularização.

Tabela 5.18: *RMSE* de validação e treinamento para as topologias de redes que tiveram os menores valores – conjunto 2.

			RMSE	
			Conjunto 1	Conjunto 2
Variável	Modelo	Topologia	Treinamento	Validação
CONV	RNA-RB	(5:7:1)	0,1167	0,0544
RGAS	RNA-RB	(5:2:1)	0,3016	0,1688
RGLP	RNA-RB	(5:10:1)	0,3116	0,3440
RGC	RNA-RB	(5:10:1)	0,1350	0,1493
RLCO	RNA-RB	(5:9:1)	0,3157	0,1413
ROD	RNA-RB	(5:8:1)	0,0483	0,0302
RCOQ	PCR-RNA	(3:7:1)	0,2801	0,2781

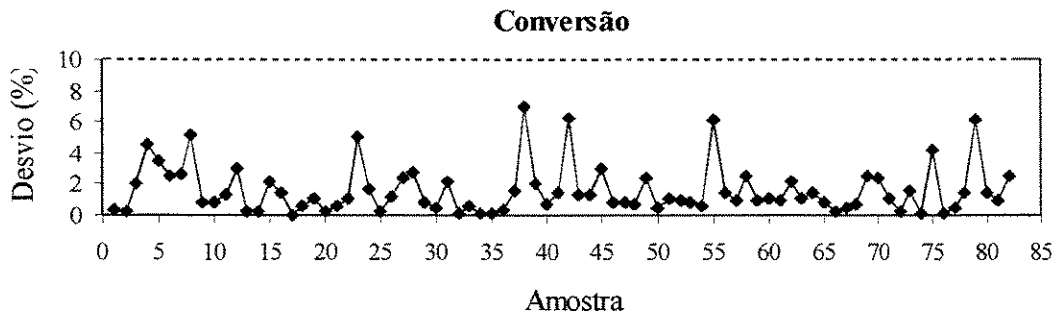


Figura 5.38: Desvios para a variável de saída conversão em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX.

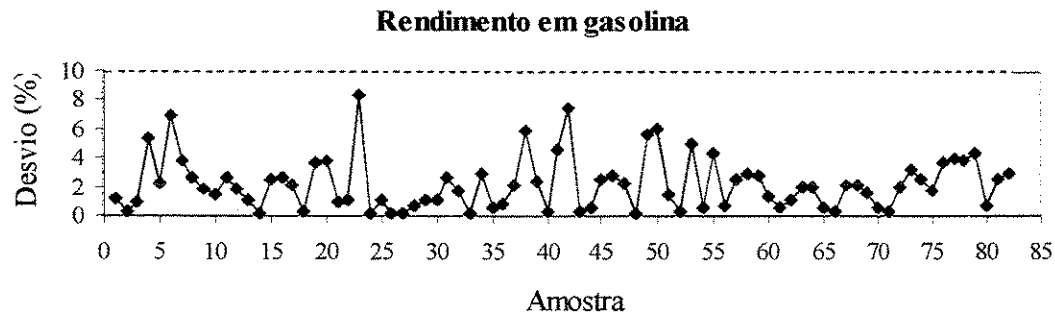


Figura 5.39: Desvios para a variável de saída rendimento em gasolina em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX.

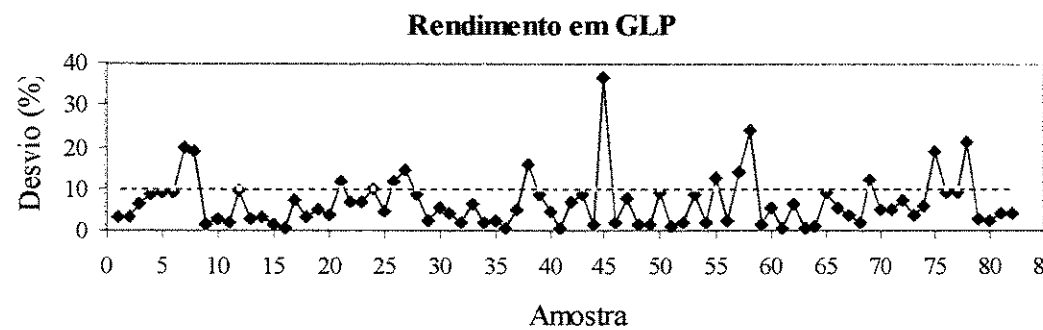


Figura 5.40: Desvios para a variável de saída rendimento em GLP em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX.

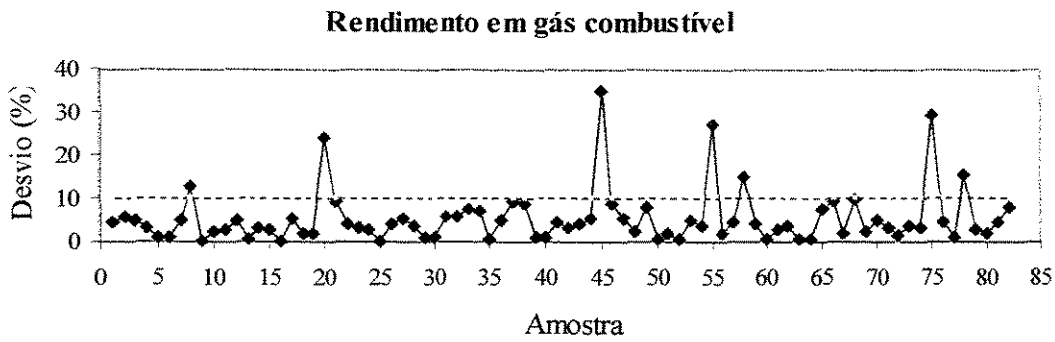


Figura 5.41: Desvios para a variável de saída rendimento em gás combustível em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX.

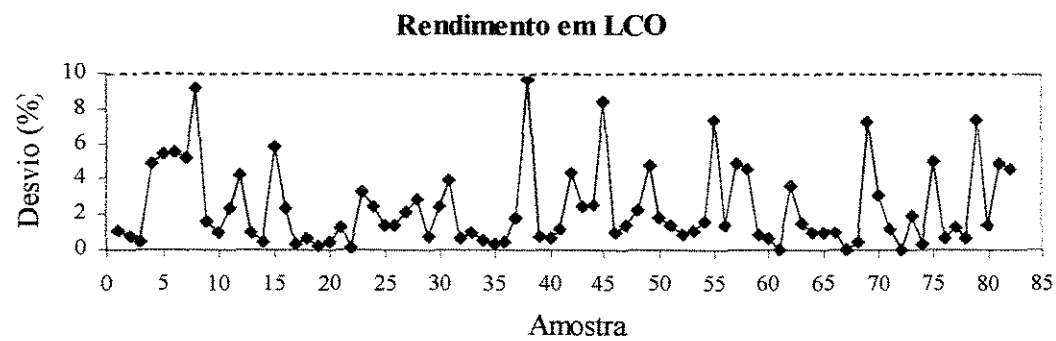


Figura 5.42: Desvios para a variável de saída rendimento em LCO em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX.

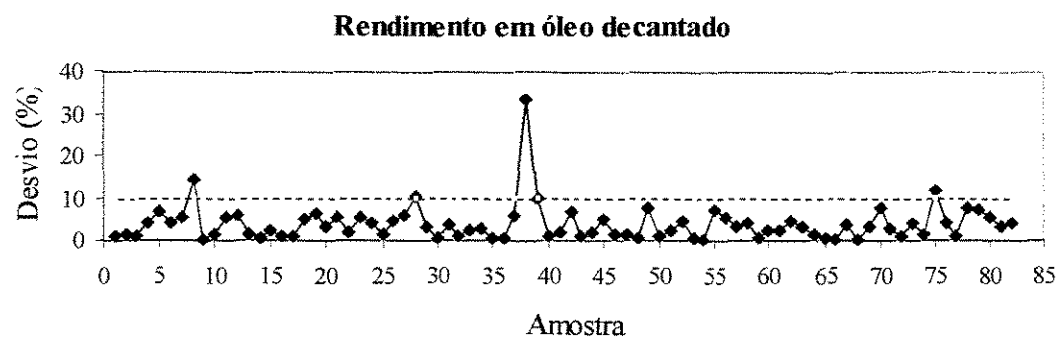


Figura 5.43: Desvios para a variável de saída rendimento em óleo decantado em função das amostras para o método LM com regularização Bayesiana na modelagem individual dos dados da SIX.

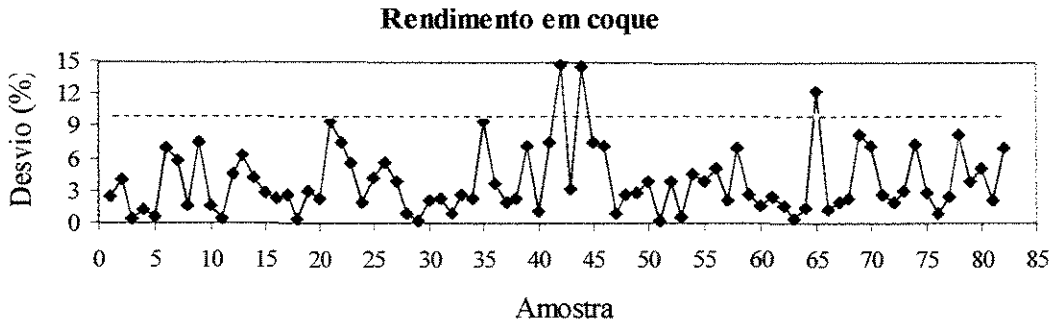


Figura 5.44: Desvios para a variável de saída rendimento em coque em função das amostras para o método PCR-RNA com parada antecipada na modelagem individual dos dados da SIX.

5.3 Modelagem Global – Dados da RLAM

A seguir são mostrados os resultados obtidos para uma unidade industrial de craqueamento catalítico da refinaria Landulpho Alves (RLAM) localizada em São Francisco do Conde – BA. As variáveis de entrada e saída foram apresentadas no capítulo 2.

O objetivo aqui é aplicar a metodologia utilizada na planta piloto para modelar a unidade industrial usando os modelos empíricos que apresentaram resultados satisfatórios para a planta piloto da SIX, ou seja, redes neurais com o método LM (utilizando validação cruzada e regularização Bayesiana) e redes neurais utilizando os escores do modelo de PCA (redução da dimensionalidade na entrada dos dados). Assim como na seção 5.1.1, inicialmente os dados passam por uma análise exploratória usando a técnica de PCA para detectar possíveis *outliers*. Após esta etapa é que o conjunto de dados restantes foi utilizado para obtenção dos modelos empíricos.

5.3.1 Análise Exploratória dos Dados

Seguindo a mesma metodologia da Seção 5.1.1, foi utilizada a técnica de PCA para fazer uma análise exploratória dos dados fornecidos pela RLAM. Os resultados a seguir

estão mais resumidos em relação aos resultados descritos para SIX, pois todas as discussões pertinentes já foram feitas.

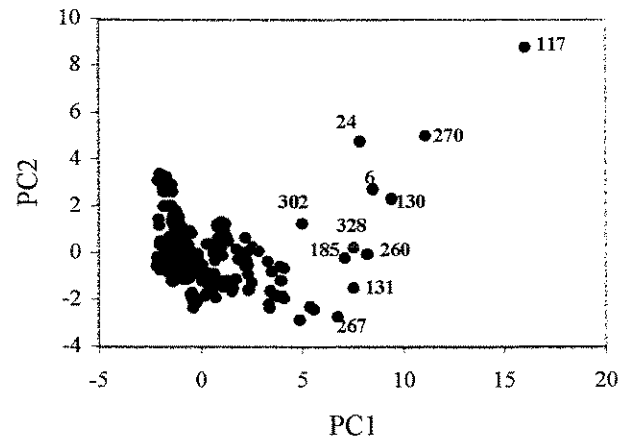
Os dados foram, inicialmente, pré-processados com o autoescalamento e o resultado da PCA mostrou que nove componentes principais são suficientes para descrever o conjunto de dados (através da variância acumulada). Para revelar amostras com possíveis comportamentos anômalos foram colocados em gráfico os escores das componentes principais. Também foram feitos os gráficos dos pesos para verificar se há variáveis que sejam correlacionáveis.

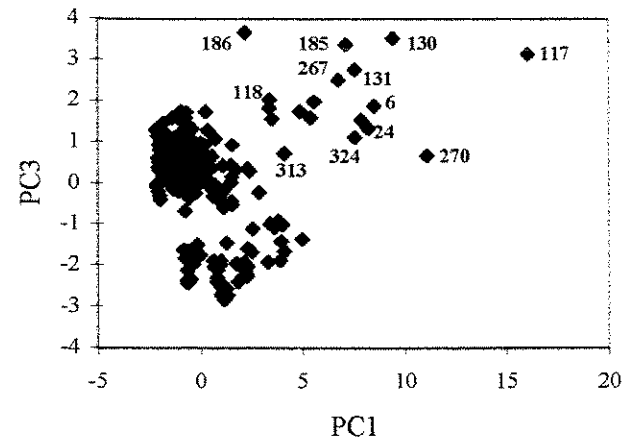
A Figura 5.45 mostra o gráfico dos escores para as componentes principais (PC) 1 e 2 (descrevem 53% da informação dos dados). A Figura 5.46 mostra o gráfico dos pesos das variáveis para as 4 primeiras componentes principais. Através dos gráficos dos escores observa-se que algumas amostras estão distantes do conjunto (por exemplo, amostras 6, 117, 130, 270, entre outras). Com relação aos gráficos dos pesos, pode-se observar que não há variáveis que sejam correlacionáveis como aconteceu com duas variáveis da SIX e que todas as variáveis contribuem na construção do modelo de PCA.

Para verificar a presença de outliers utilizou-se a estatística T^2 de Hotelling para remover amostras que estejam fora do modelo de PCA (com 95% de confiança). A Figura 5.47 indicou que 37 amostras estiveram foram do modelo de PCA e que foram removidas do conjunto de dados, reduzindo o mesmo para 256 amostras mantendo-se, porém, o mesmo número de variáveis de entrada.

Com este novo conjunto de dados (chamado de dados 2), novamente foi feito o pré-processamento dos dados e a PCA para este conjunto indicou que oito componentes eram suficientes.

A Figura 5.48 mostra o gráfico dos escores para as duas primeiras componentes principais onde ainda pode-se observar amostras distantes do conjunto de dados. Já a Figura 5.49 mostra o gráfico dos pesos de cada variável em função da primeira componente principal. Nesta figura se verifica que todas as variáveis foram importantes na construção do modelo.





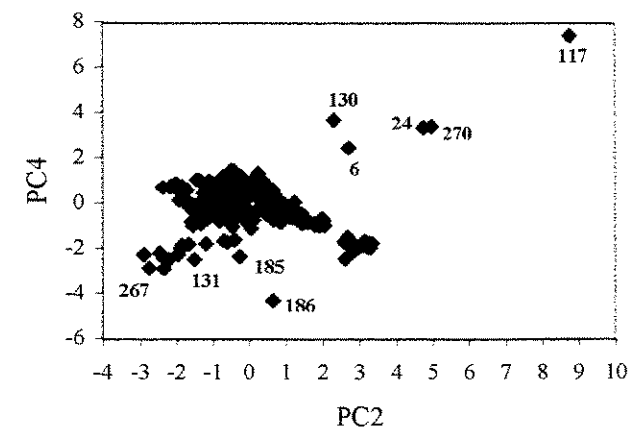


Figura 5.45: Gráficos dos escores em PC1, PC2, PC3 e PC4 - Dados 1.

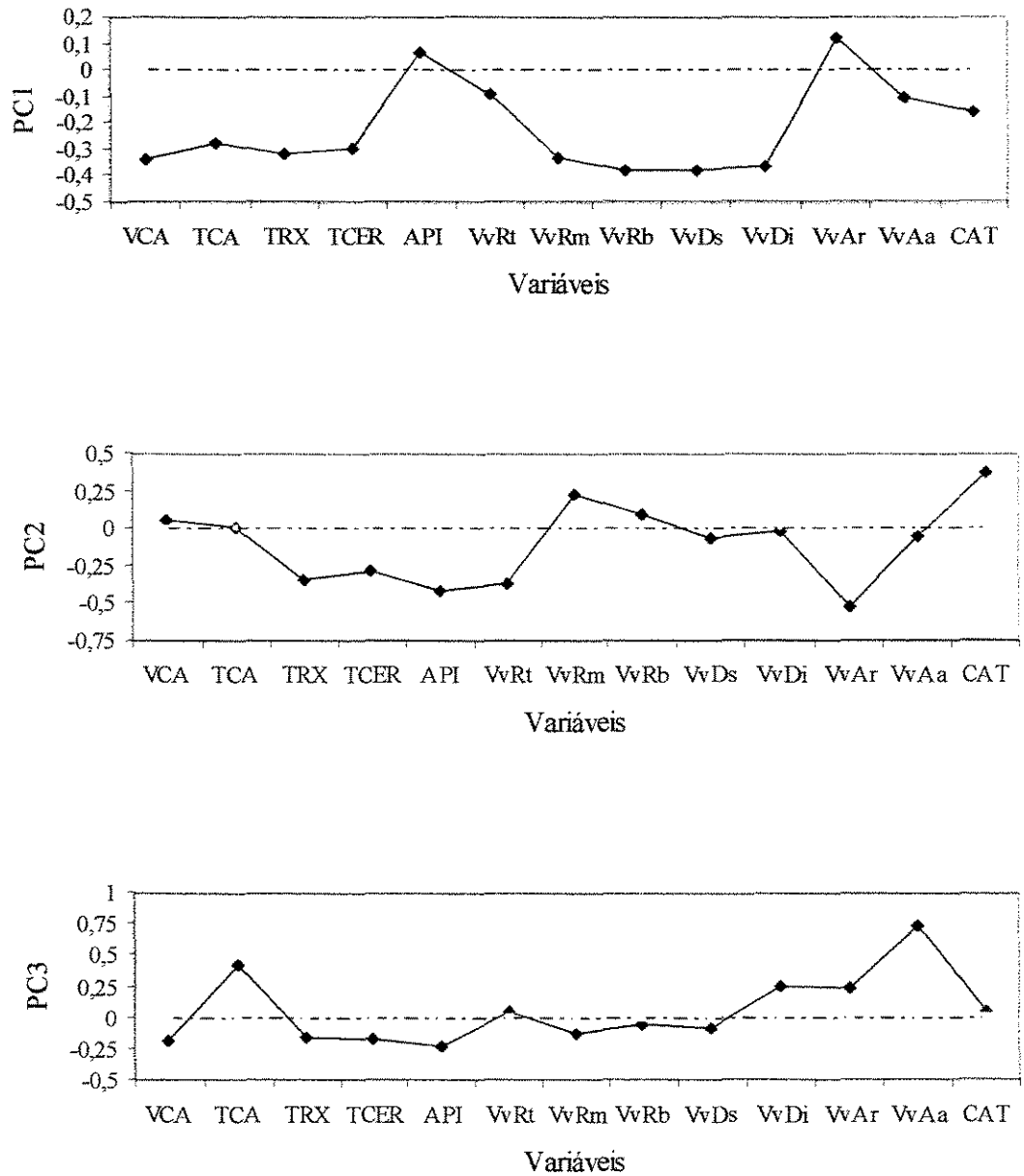


Figura 5.46: Gráficos dos pesos. PC1, PC2 e PC3 *versus* variáveis de saída - Dados 1.

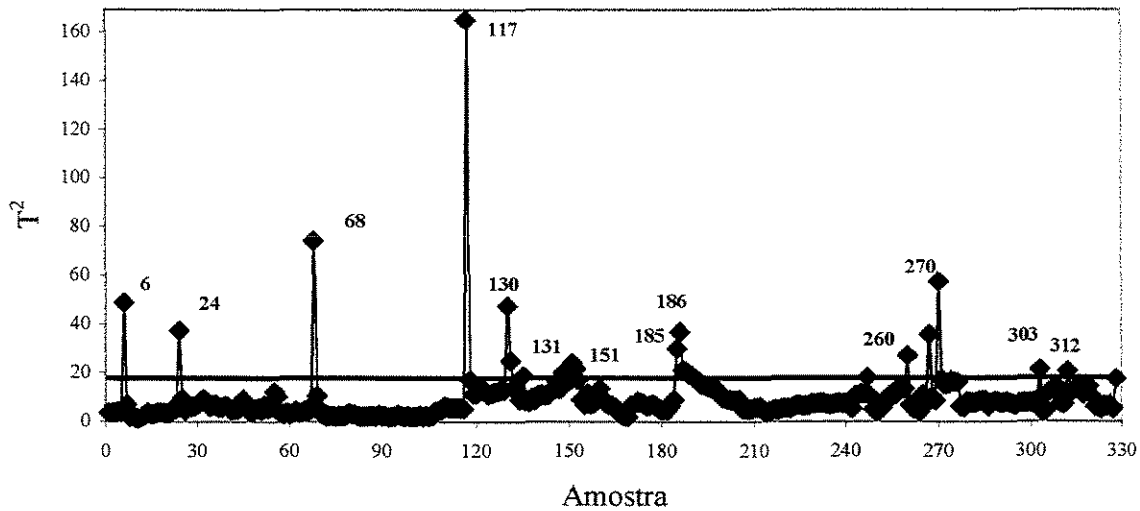


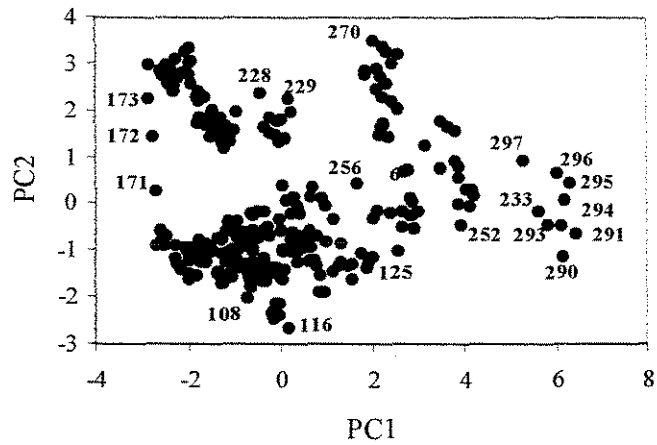
Figura 5.47: Gráfico de T^2 em função das amostras – Dados 1.

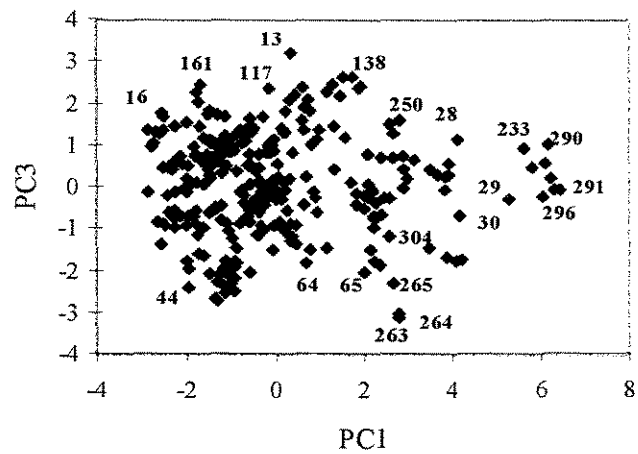
A Figura 5.50 mostra o gráfico de T^2 em função das amostras para os dados 2. Nesta figura nota-se que 48 amostras estão fora do modelo e que foram removidas formando um novo conjunto com 236 amostras e todas as variáveis de entrada originais (chamado de Dados 3).

Novamente utilizou a PCA para o conjunto que foi modificado (Dados 3) a qual indicou que oito componentes principais descrevem este conjunto (através da porcentagem de variância descrita por cada componente principal).

A Figura 5.51 mostra o gráfico dos escores no qual observa-se que pode haver amostras que ainda não esteja dentro do modelo de PCA. No entanto, através da Figura 5.52 que mostra o gráfico de T^2 em função das amostras, verifica-se que todas as amostras estão abaixo do limite (95% de confiança). Os gráficos dos pesos forneceram os mesmos resultados que os casos anteriores e está representado na Figura 5.53.

Os resultados indicaram que não houve um descarte de variáveis e de amostras para este último conjunto de dados. Assim, este último conjunto foi usado para treinamento, validação e teste dos seguintes modelos empíricos: redes neurais artificiais usando o método LM (validação cruzada e regularização) e redes neurais artificiais com os escores obtidos pela PCA (método PCR-RNA). A seguir são apresentados os resultados para a modelagem global e individual da unidade industrial de FCC.





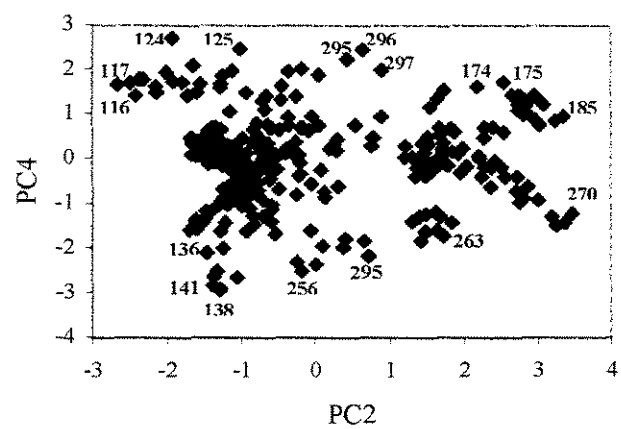


Figura 5.48: Gráficos dos escores em PC1, PC2, PC3 e PC4 - Dados 2.

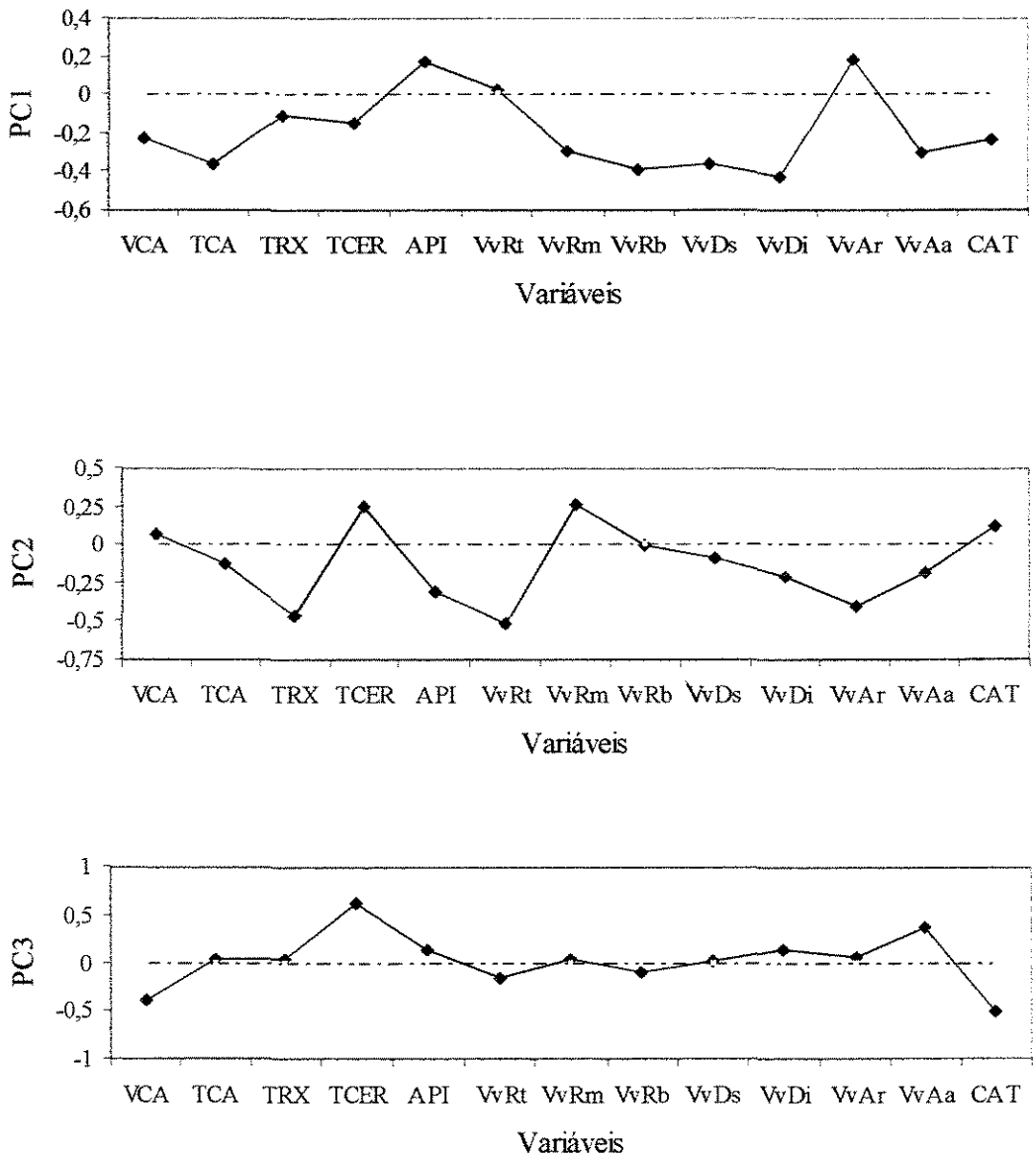


Figura 5.49: Gráficos dos pesos. PC1 *versus* variável de saída - Dados 2.

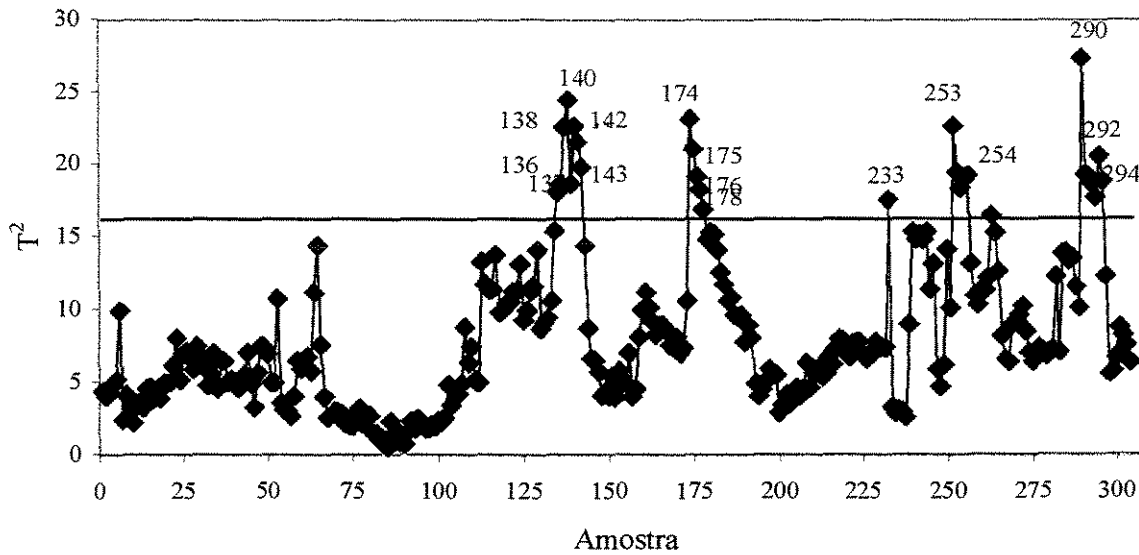


Figura 5.50: Gráfico de T^2 em função das amostras – Dados 2.

5.3.2 Resultados da Modelagem Global

O método LM foi usado para todos os modelos de redes neurais que foram aplicados aos dados da unidade industrial de FCC, pois este método havia apresentado melhores resultados para a modelagem da planta piloto.

A Tabela 5.19 mostra os valores de *RMSE* globais para as etapas de treinamento e validação dos modelos empíricos considerados, que são: redes neurais com validação cruzada (RNA-VC), redes neurais com regularização Bayesiana (RNA-RB) e redes neurais usando os escores obtidos pela análise dos componentes principais (PCR-RNA). O número de neurônios na camada oculta foi modificado de 1 a 25. Foram feitos 30 treinamentos para cada neurônio oculto apresentado nesta tabela e os resultados para cada neurônio oculto que aparecem na Tabela 5.19 referem-se ao menor valor de *RMSE* no conjunto de validação.

Na Tabela 5.19 observa-se que a topologia para o modelo RNA-VC foi (13:10:8), para o modelo RNA-RB foi (13:21:8) e para PCR-RNA foi (8:21:8). Para fazer uma comparação entre os modelos foi usado um conjunto de teste o qual indicou qual dos modelos apresentou um maior poder de predição.

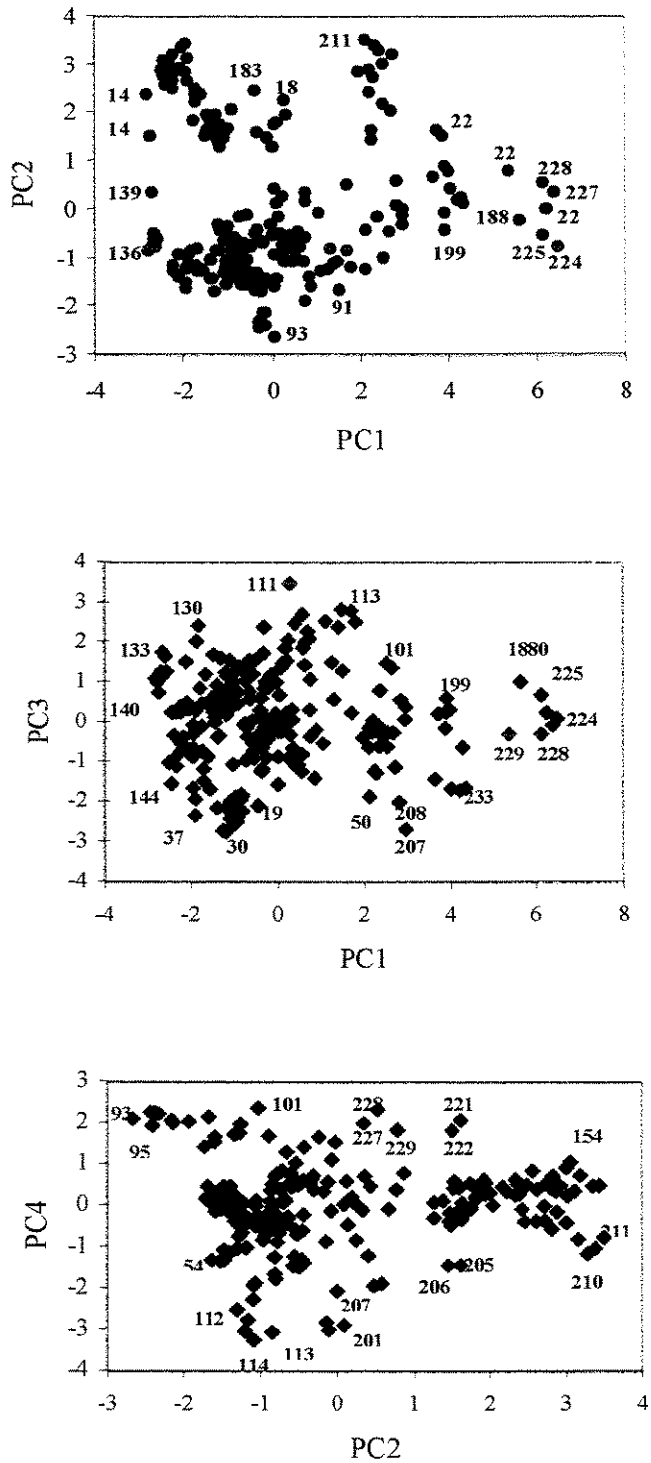


Figura 5.51: Gráficos dos escores em PC1, PC2, PC3 e PC4 - Dados 3.

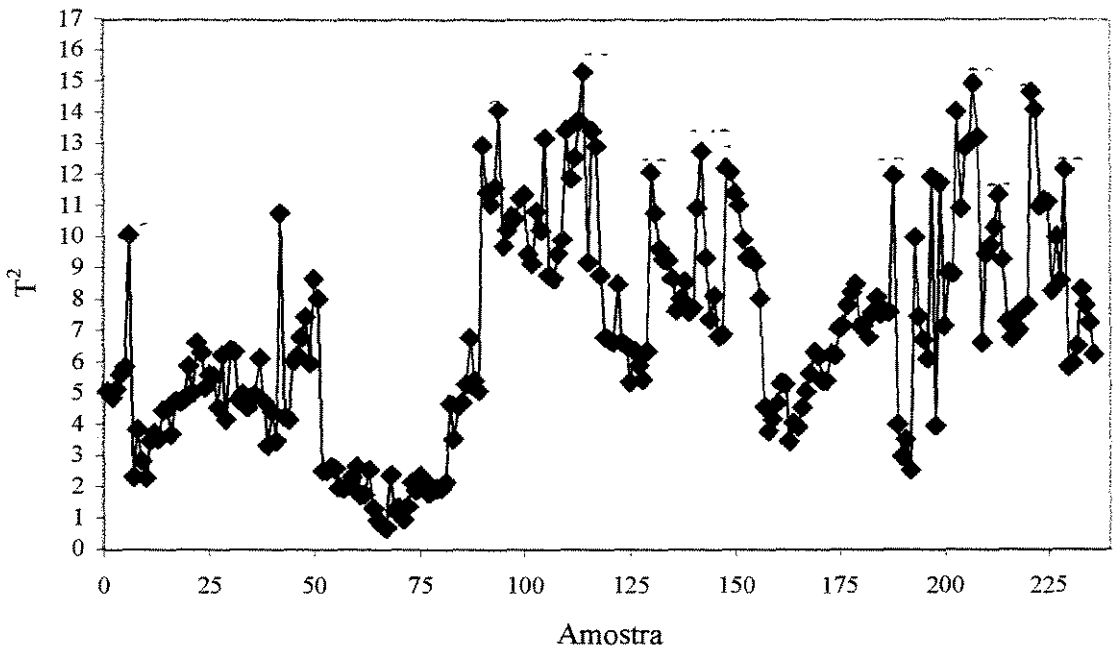


Figura 5.52: Gráfico de T^2 em função das amostras – Dados 3.

A Tabela 5.20 mostra os resultados de *RMSE* no conjunto de teste obtidos por cada modelo estudado nesta seção. Nota-se que o modelo PCR-RNA apresentou os menores *RMSE* para a maioria das variáveis de saída (VPE, VPA, VGLP, VGAS e VLCO). Já o modelo RNA-RB teve os menores *RMSE* para as variáveis VGC e VGA enquanto o modelo RNA-VC teve o menor *RMSE* para a variável VOD. Em outras palavras, o modelo PCR-RNA forneceu os melhores resultados pois apresentou uma maior capacidade de predição para novas amostras, principalmente para as variáveis de interesse que são vazão de gasolina (VGAS) e vazão de GLP (VGLP).

As Figuras 5.54 – 5.61 mostram os gráficos dos desvios obtidos pelo modelo PCR-RNA para as amostras de treinamento, validação e teste. Nesta figura observa-se que o modelo foi capaz de predizer 98,23% das amostras com desvio (absoluto) inferior a 10% para a variável vazão de gás combustível (VGC), e da mesma forma para vazão de gás ácido (VGA): 88,05%; vazão de propeno (VPE): 47,79%; vazão de propano: 30,97%; vazão de GLP (VGLP):77,43%; vazão de gasolina (VGAS): 95,13%; vazão de LCO (VLCO): 63,72% e vazão de óleo decantado (VOD): 45,13%.

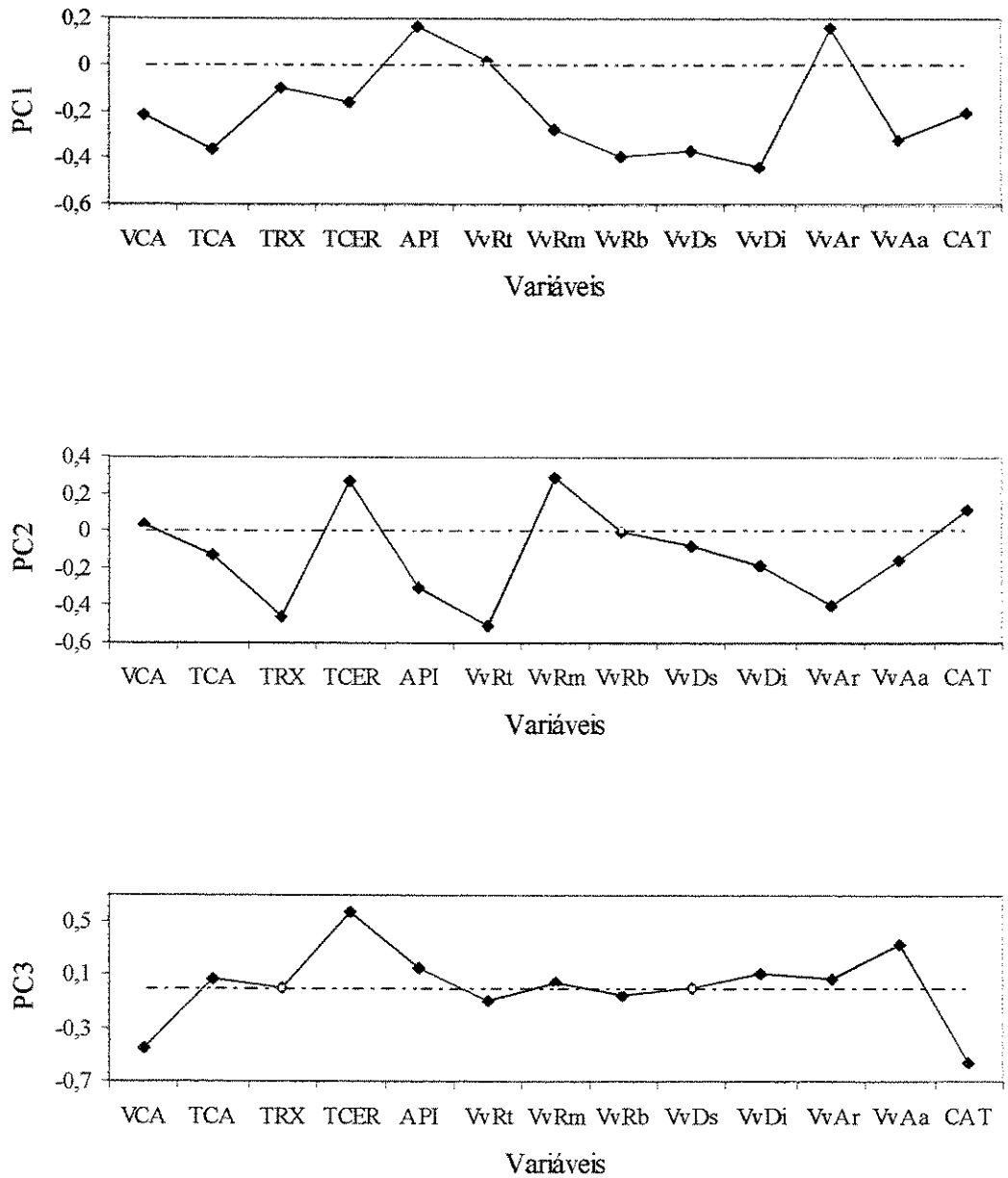


Figura 5.53: Gráficos dos pesos. PC1 *versus* variável de saída - Dados 3.

Tabela 5.19: *RMSE* globais para treinamento e validação de diferentes modelos empíricos globais – Dados da RLAM.

	RMSE					
	RNA-VC		RNA-RM		PCR-RNA	
Neurônios ocultos	Treinamento	Validação	Treinamento	Validação	Treinamento	Validação
1	0,9113	0,8527	0,9143	0,8609	0,9136	0,8815
2	0,7571	0,7668	0,7614	0,7693	0,7848	0,7872
3	0,6302	0,6516	0,6526	0,6569	0,6580	0,6564
4	0,5961	0,6229	0,5880	0,6262	0,6143	0,6262
5	0,5533	0,5977	0,5591	0,6096	0,5733	0,6006
6	0,5337	0,5961	0,5366	0,5919	0,5710	0,5836
7	0,5028	0,5860	0,5186	0,5828	0,5403	0,5631
8	0,4770	0,5838	0,4935	0,5670	0,5279	0,5678
9	0,4835	0,5832	0,4738	0,5631	0,5003	0,5506
10	0,4713	0,5714	0,4707	0,5661	0,4913	0,5528
11	0,4417	0,5784	0,4470	0,5627	0,4759	0,5442
12	0,4548	0,5754	0,4436	0,5605	0,4633	0,5472
13	0,4268	0,5751	0,4072	0,5530	0,4316	0,5370
14	0,4411	0,5800	0,4256	0,5482	0,4468	0,5299
15	0,4897	0,5947	0,4194	0,5521	0,4606	0,5309
16	0,4186	0,5896	0,4431	0,5455	0,4340	0,5256
17	0,3946	0,5836	0,4022	0,5512	0,4216	0,5312
18	0,4115	0,5876	0,4045	0,5403	0,3926	0,5275
19	0,4149	0,5889	0,3791	0,5474	0,3984	0,5344
20	0,3930	0,5924	0,3542	0,5412	0,3945	0,5312
21	0,3915	0,5934	0,3937	0,5361	0,4035	0,4934
22	0,3890	0,5941	0,3794	0,5421	0,3628	0,5226
23	0,3702	0,6096	0,3482	0,5471	0,3832	0,5270
24	0,3787	0,6192	0,3393	0,5457	0,3625	0,5273
25	0,4620	0,6251	0,3744	0,5485	0,3729	0,5211

Tabela 5.20: *RMSE* de cada variável de saída no conjunto de teste obtidos por diferentes modelos empíricos – Dados da RLAM.

	RMSE		
Variável de saída	RNA-VC	RNA-RM	PCR-RNA
gás combustível - (VGC)	0,3423	0,3239	0,4714
gás ácido - (VGA)	0,6277	0,5486	0,5976
Propeno - (VPE)	0,6295	0,7152	0,5207
Propano - (VPA)	0,4644	0,5899	0,4351
GLP - (VGLP)	0,4863	0,4360	0,3762
Gasolina - (VGAS)	0,3881	0,4532	0,4309
LCO - (VLCO)	0,6003	0,5796	0,5563
óleo decantado - (VOD)	0,5685	0,5937	0,6591

Os resultados citados anteriormente mostram que o modelo global apresentou um bom desempenho na predição de quatro variáveis (VGC, VGA, VGLP e VGAS), teve um resultado satisfatório para a variável VLCO e resultados que não foram satisfatórios para as variáveis VPE, VPA e VOD.

Apesar dos bons resultados para as variáveis VGLP e VGAS (as quais são de maior interesse), foram feitas modelagens para cada variável de saída com a finalidade de conseguir uma melhoria para as variáveis VPE, VPA e VOD. Na seção a seguir são apresentados os resultados obtidos por esses modelos.

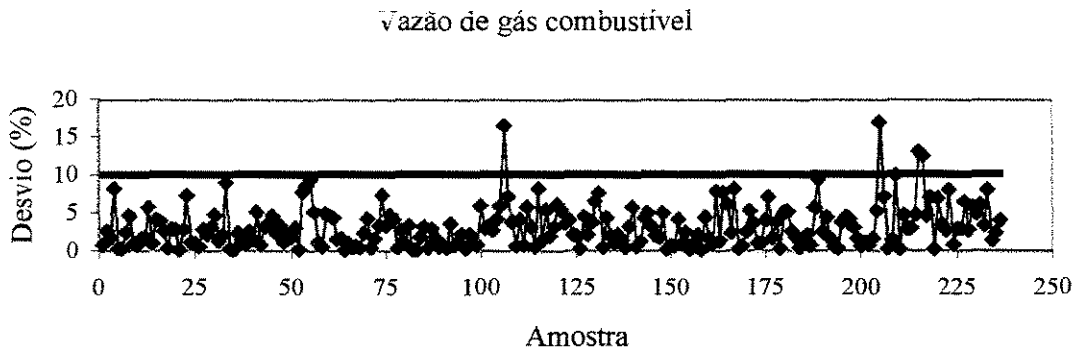


Figura 5.54 – Desvios para a variável de saída vazão de gás combustível em função das amostras para o modelo global PCR-RNA com dados da RLAM.

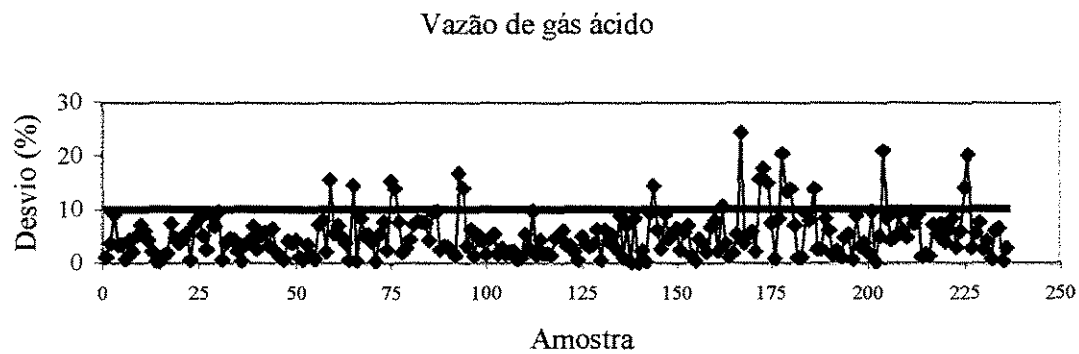


Figura 5.55 – Desvios para a variável de saída vazão de gás ácido em função das amostras para o modelo global PCR-RNA com dados da RLAM.

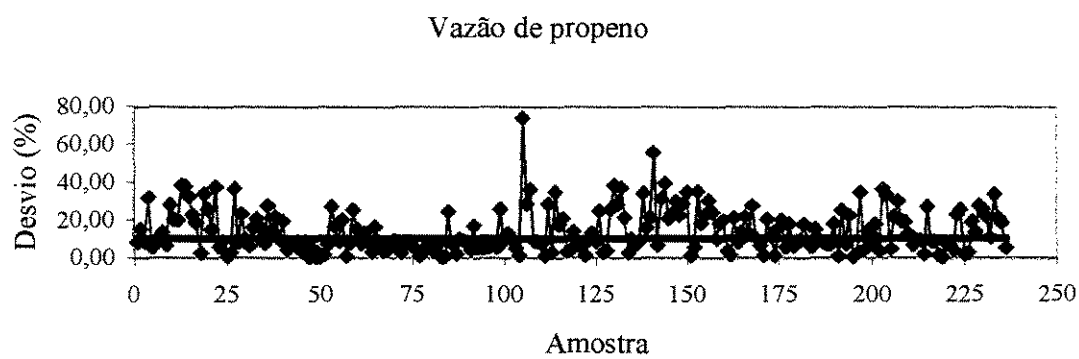


Figura 5.56 – Desvios para a variável de saída vazão de propeno em função das amostras para o modelo global PCR-RNA com dados da RLAM.

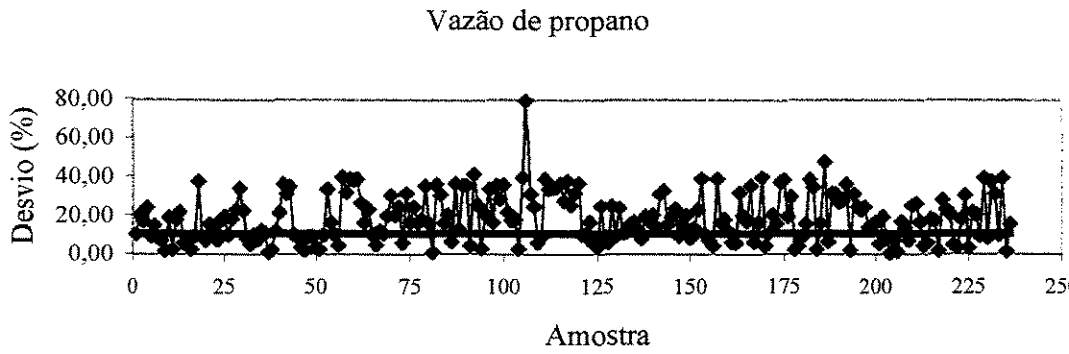


Figura 5.57 – Desvios para a variável de saída vazão de propano em função das amostras para o modelo global PCR-RNA com dados da RLAM.

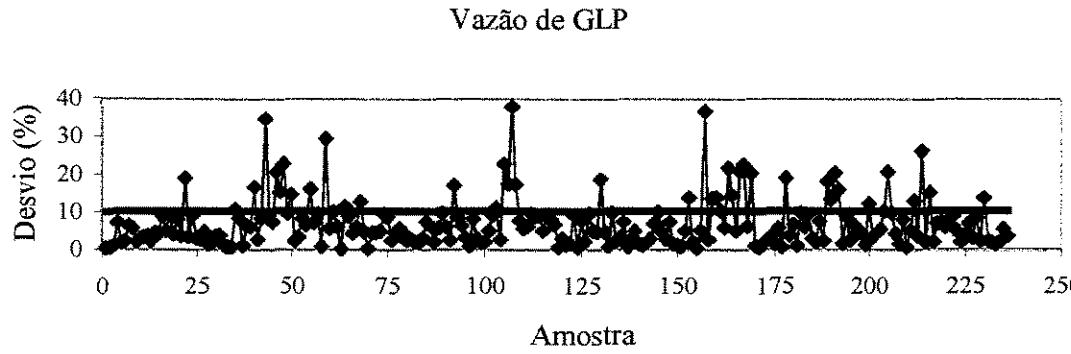


Figura 5.58 – Desvios para a variável de saída vazão de GLP em função das amostras para o modelo global PCR-RNA com dados da RLAM.

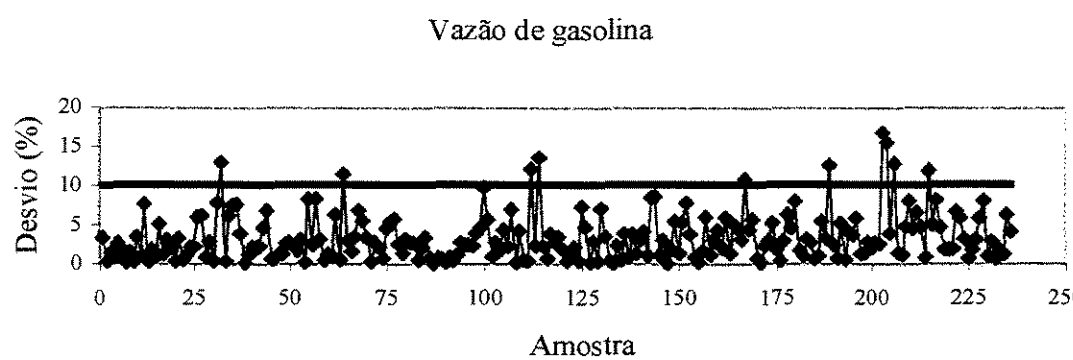


Figura 5.59 – Desvios para a variável de saída vazão de gasolina em função das amostras para o modelo global PCR-RNA com dados da RLAM.

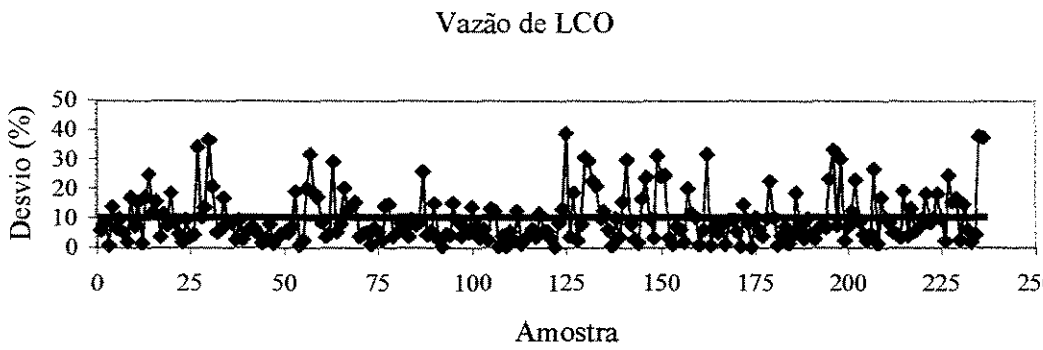


Figura 5.60 – Desvios para a variável de saída vazão de LCO em função das amostras para o modelo global PCR-RNA com dados da RLAM.

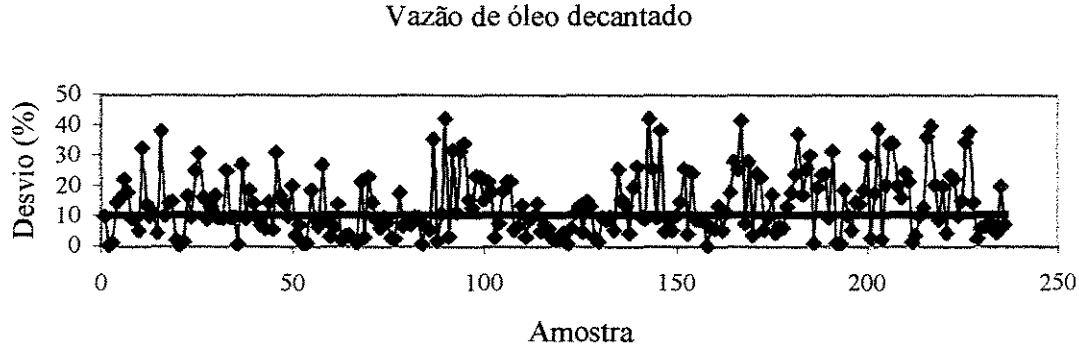


Figura 5.61 – Desvios para a variável de saída vazão de óleo decantado em função das amostras para o modelo global PCR-RNA com dados da RLAM.

5.4 Modelagem Individual – Dados da RLAM

Foi seguida a mesma metodologia adotada na Seção 5.3 para a modelagem individual dos dados da SIX, porém os modelos usados foram os mesmos da Seção 5.3, ou seja, os modelos RNA-VC, RNA-RB e PCR-RNA.

A Tabela 5.21 mostra os valores obtido para *RMSE* nas etapas de treinamento e validação para os modelos considerados. Também são mostradas as topologias das redes para cada variável de saída. No Apêndice A são fornecidos todos os valores de *RMSE*

obtidos, modificando o número de neurônios na camada oculta de 1 a 20 (melhores resultados de 30 repetições).

A Tabela 5.22 mostra os valores de *RMSE* para o conjunto de teste, os quais foram utilizados para comparar os modelos através da capacidade de predição de novas amostras. Nesta tabela pode-se observar que para as variáveis VGC e VGAS o modelo RNA-VC apresentou os melhores resultados, já o modelo RNA-RB teve os melhores resultados para as variáveis VGA e VOD. O modelo PCR-RNA foi o que apresentou os menores valores de *RMSE* no conjunto de teste para as variáveis VPE, VPA, VGLP e VLCO.

Tabela 5.21: *RMSE* globais para treinamento e validação de diferentes modelos empíricos globais – Dados da RLAM.

	RMSE					
	RNA-VC		RNA-RB		PCR-RNA	
Variáveis	Treinamento (topologia)	Validação	Treinamento (topologia)	Validação	Treinamento (topologia)	Validação
VGC	0,1518 (13:16:1)	0,2505	0,1488 (13:15:1)	0,2970	0,1872 (13:14:1)	0,2840
VGA	0,3697 (13:20:1)	0,4943	0,3369 (13:7:1)	0,4657	0,3146 (13:8:1)	0,5096
VPE	0,5344 (13:19:1)	0,7871	0,4098 (13:7:1)	0,7644	0,5449 (13:13:1)	0,7054
VPA	0,5146 (13:8:1)	0,6286	0,5829 (13:14:1)	0,6305	0,5079 (13:15:1)	0,5596
VGLP	0,2784 (13:5:1)	0,4126	0,3174 (13:4:1)	0,4368	0,3533 (13:6:1)	0,3866
VGAS	0,1854 (13:14:1)	0,3070	0,2409 (13:14:1)	0,3101	0,2671 (13:11:1)	0,3221
VLCO	0,3839 (13:9:1)	0,4912	0,3662 (13:9:1)	0,5513	0,3897 (13:5:1)	0,5046
VOD	0,2831 (13:16:1)	0,5189	0,2402 (13:12:1)	0,4933	0,2808 (13:13:1)	0,5161

Tabela 5.22: *RMSE* de cada variável de saída no conjunto de teste obtidos por diferentes modelos empíricos – Dados da RLAM.

	RMSE		
Variável de saída	RNA-VC	RNA-RM	PCA-RNA
gás combustível - (VGC)	0,2446	0,2899	0,6150
gás ácido - (VGA)	0,5019	0,5470	0,6032
Propeno - (VPE)	0,6488	0,7806	0,4519
Propano - (VPA)	0,5835	0,6171	0,3687
GLP - (VGLP)	0,4339	0,4751	0,4134
Gasolina - (VGAS)	0,3797	0,4094	0,5649
LCO - (VLCO)	0,5780	0,5971	0,6658
óleo decantado - (VOD)	0,5454	0,6118	0,6658

As Figuras 5.62 – 5.69 apresentam os gráficos dos desvios obtidos pelos modelos com melhor poder de predição para cada variável de saída da unidade de FCC da RLAM. As amostras de 1 a 124 foram as usadas para treinamento, 125 a 175 para validação e 176 a 226 para teste. Dessas figuras observa-se que o modelo foi capaz de predizer 99,12% das amostras com desvio (absoluto) inferior a 10% para a variável vazão de gás combustível (VGC), para a variável vazão de gás ácido (VGA) foi de 91,15%, para vazão de propeno (VPE) foi 48,67%, para vazão de propano foi 31,42%, vazão de GLP (VGLP) foi 82,74%, vazão de gasolina (VGAS) foi 96,46%, vazão de LCO (VLCO) foi 64,16% e vazão de óleo decantado (VOD) foi 63,27%.

Comparando com os resultados obtidos pela modelagem global, os resultados individuais forneceram uma pequena melhoria para quase todas as variáveis, exceto para a variável VOD a qual passou a ter um resultado satisfatório. Já as variáveis VPE e VPA continuaram fornecendo valores abaixo de 50% de predição das amostras. Este resultados para VPE e VPA indicam a necessidade de utilizar mais neurônios na camada oculta ou mesmo utilizar outros modelos empíricos que não foram testados na modelagem com dados da RLAM, como por exemplo o PLS com redes neurais.

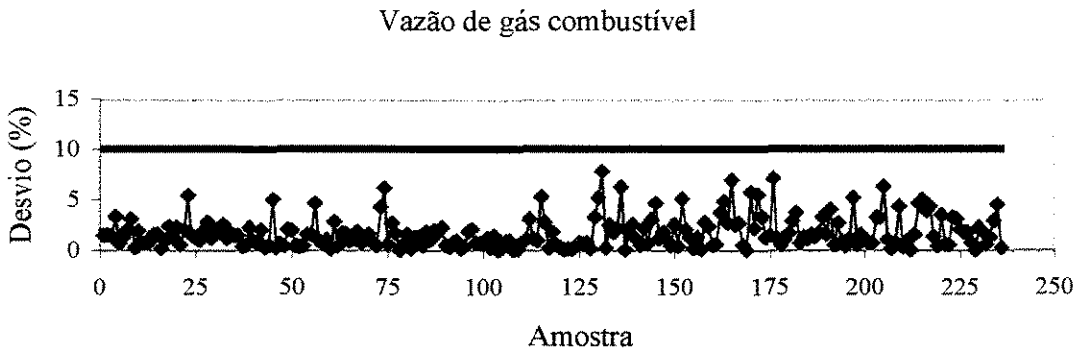


Figura 5.62: Desvios para a variável de saída vazão de gás combustível (VGC) em função das amostras para o modelo RNA-VC com dados da RLAM.

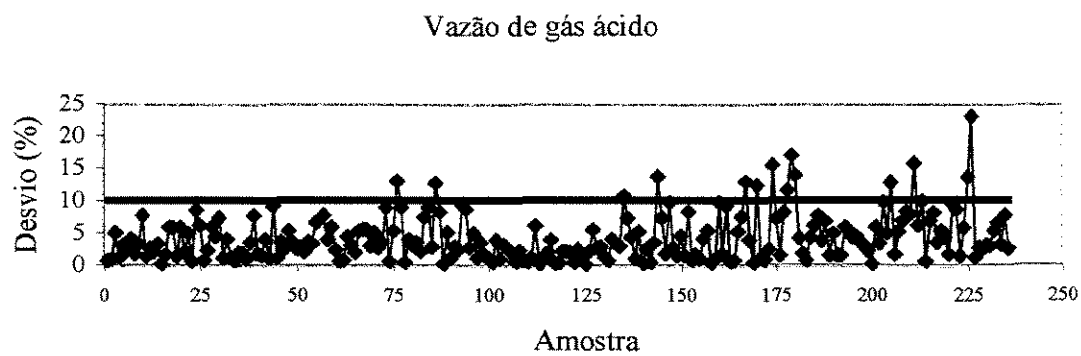


Figura 5.63: Desvios para a variável de saída vazão de gás ácido (VGA) em função das amostras para o modelo RNA-RB com dados da RLAM.

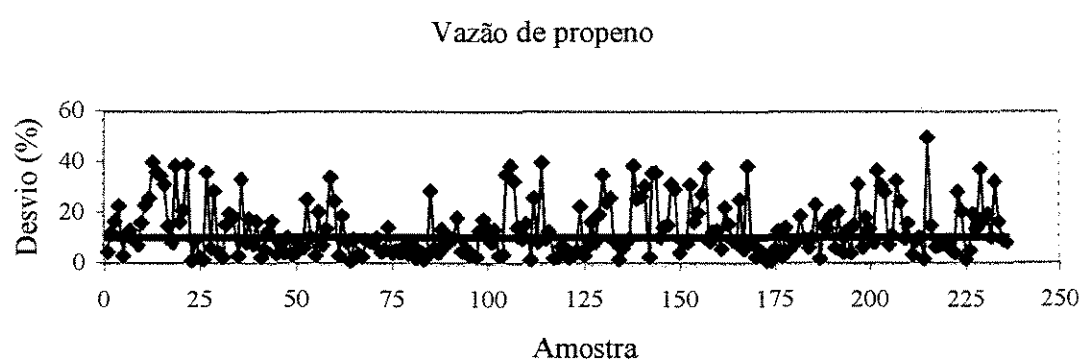


Figura 5.64: Desvios para a variável de saída vazão de propeno (VPE) em função das amostras para o modelo PCA-RNA com dados da RLAM.

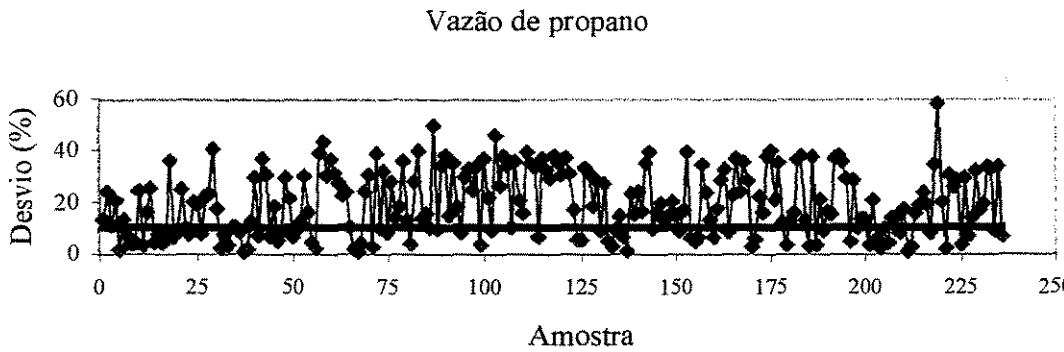


Figura 5.65: Desvios para a variável de saída vazão de propano (VPA) em função das amostras para o modelo PCA-RNA com dados da RLAM.

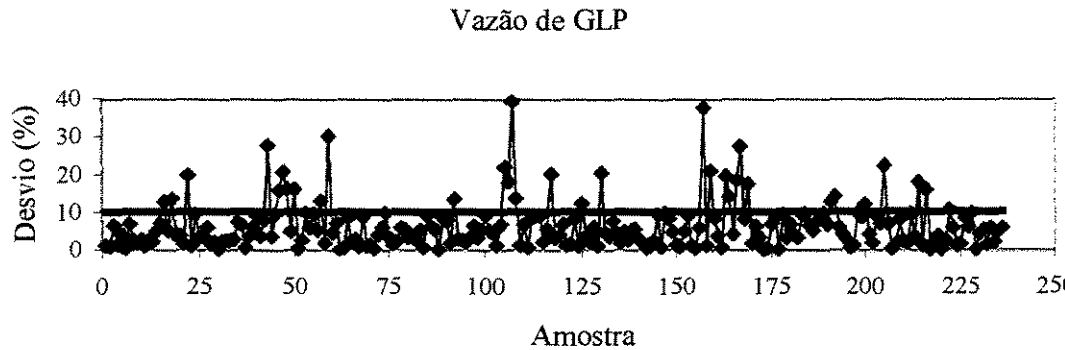


Figura 5.66: Desvios para a variável de saída vazão de gás liquefeito do petróleo (VGLP) em função das amostras para o modelo PCA-RNA com dados da RLAM.

Figura 5.67: Desvios para a variável de saída vazão de gasolina (VGAS) em função das amostras para o modelo RNA-VC com dados da RLAM.

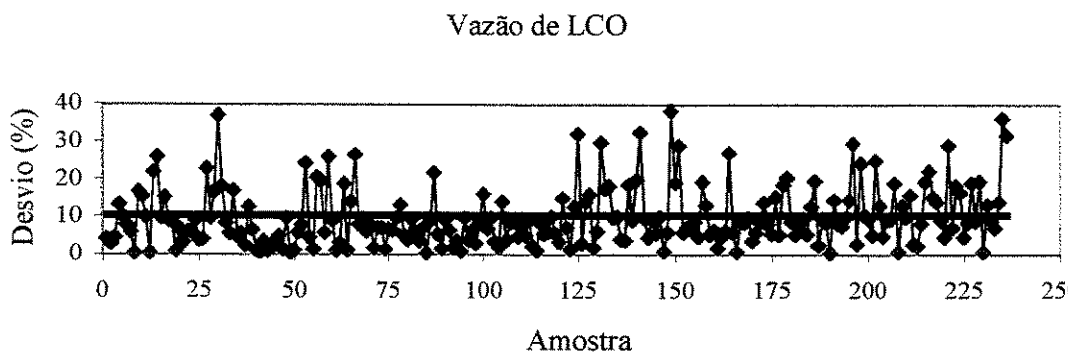


Figura 5.68: Desvios para a variável de saída vazão de óleo leve de reciclo (LCO) em função das amostras para o modelo PCA-RNA com dados da RLAM.

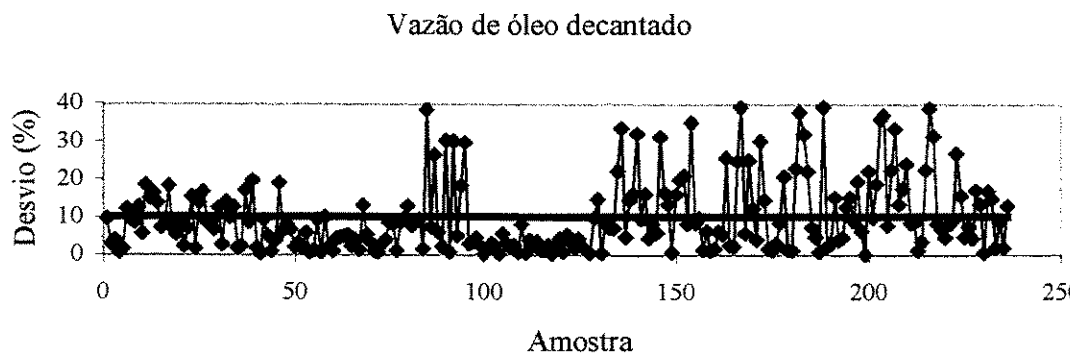


Figura 5.69: Desvios para a variável de saída vazão de óleo decantado (VOD) em função das amostras para o modelo RNA-RB com dados da RLAM.

CAPÍTULO 6

CONCLUSÕES

6.1 Conclusões Gerais

O presente trabalho teve como objetivo principal obter uma ferramenta que fosse capaz de fornecer a conversão e os produtos do craqueamento catalítico (em termos de rendimentos) em função das condições operacionais do processo. Esta ferramenta poderá ser utilizada pela SIX – Petrobras para as próximas etapas do seu projeto de aperfeiçoamento de *risers*.

Os modelos empíricos desenvolvidos possuem algumas limitações as quais estão associadas ao conjunto de dados experimentais fornecidos pela SIX – Petrobras. As principais limitações do modelo estão associadas as características da carga fornecida pela REPAR, cuja composição não foi alterada. Com a realização de novos testes sob diferentes condições de operação, incluindo cargas com diferentes características, as ferramentas aqui apresentadas poderão ser novamente utilizadas para obter um modelo empírico mais representativo do processo de craqueamento catalítico.

Os bons resultados com os dados da SIX (planta piloto) motivaram a aplicação da metodologia numa unidade industrial de FCC (RLAM). Neste caso, o objetivo foi obter uma ferramenta que prediga as vazões de saída dos produtos do processo.

Em ambos os casos, foram utilizados diferentes metodologias para modelar os processos. Ferramentas da quimiometria, de redes neurais e a combinação de ambas foram aplicadas e seus resultados foram discutidos e comparados. A seguir são apresentadas as conclusões específicas para as duas unidades de FCC modeladas.

6.2 Conclusões sobre os resultados obtidos para SIX

Neste trabalho foram realizadas as modelagens utilizando diferentes técnicas lineares e não lineares de regressão, tais como a regressão dos componentes principais (PCR), mínimos quadrados parciais (PLS) e redes neurais artificiais (RNA). Também foi utilizada a metodologia que combina as técnicas de PCA com RNA.

Antes das modelagens, foi feita uma análise exploratória dos dados na busca de possíveis variáveis que fossem altamente correlacionáveis e de amostras que tivessem comportamento muito diferente das amostras restantes. O uso da análise dos componentes principais, através dos gráficos dos pesos, permitiu a visualização de duas amostras que são altamente correlacionáveis (VCA e VvL) o que permitiu a utilização de apenas uma delas. No caso a escolhida para ser usada nas modelagens foi a variável VCA.

Através dos gráficos dos escores, verificou-se que duas amostras estavam muito distantes do conjunto dos dados e que seriam possíveis *outliers*. Através da estatística T^2 de Hotelling verificou-se que as amostras eram *outliers* e que não deveriam ser usadas no modelo. A retirada desta amostras fez com que uma das variáveis também fosse removida pois as variações observadas para estas amostras estavam associadas às duas únicas modificações feitas para a variável VvD. Nas demais amostras, VvD permaneceu constante e por isso foi necessário removê-la.

O uso da PCA como ferramenta de análise exploratória foi muito importante para este trabalho, pois esta ferramenta permite avaliar o conjunto de dados de uma forma global sem a necessidade de avaliar cada variável ou amostra de forma separada.

Mesmo com os resultados da análise exploratória, procurou-se desenvolver em paralelo a modelagem com todo o conjunto de dados (variáveis e amostras) para verificar a importância da estatística multivariada em processos com muitas variáveis (independentes e/ou dependentes) envolvidas. Desta forma, foram formados dois conjuntos: o conjunto 1 que possui 7 variáveis de entrada e 84 amostras; o conjunto 2 que possui 5 variáveis de entrada e 82 amostras. As amostras usadas na etapa de validação foram as mesmas para os dois conjuntos. Em todos os casos, os modelos empíricos para o conjunto 2 forneceram resultados melhores que aqueles obtidos pelo conjunto 1.

Os modelos de regressão linear deram os resultados com maiores valores de *RMSE*, o que mostra que, para o processo de craqueamento catalítico em questão, as técnicas de redução de dimensionalidades não aproximam o sistema para um comportamento linear. O PLS forneceu resultados um pouco melhores que o PCR.

Já o uso de redes neurais com a técnica de validação cruzada na modelagem global dos dados forneceu resultados melhores se comparados aos obtidos pelo PCR e PLS, pois trata-se de um processo altamente não-linear e as redes neurais produziram modelos satisfatórios para os dois conjuntos considerados. O método LM forneceu os menores *RMSE* para a maioria das variáveis de saída em relação aos outros métodos também estudados (BFGS e SCG) utilizando o conjunto 2.

Uma outra abordagem utilizada para melhorar a generalização da rede foi usada que é a regularização Bayesiana. Esta apresentou resultados um pouco melhores em relação as redes que trabalharam com a técnica de validação cruzada. A melhoria mais significativa foi para as variáveis: rendimento em GLP e rendimento em coque. Já o modelo que utiliza PLS com redes neurais nas suas relações internas não forneceu resultados melhores que os da regularização Bayesiana.

Na modelagem individual o conjunto 2 foi usado e diferentes modelos foram aplicados. Em quase todas as variáveis de saída o modelo de redes neurais com o método LM utilizando a regularização Bayesiana forneceu os melhores resultados e o método que utiliza os escores da técnica de análise dos componentes principais como entrada da rede neural forneceu o melhor resultado para variável rendimento em coque. De um modo geral, esta abordagem forneceu resultados melhores em relação aos resultados obtidos pela modelagem global.

6.3 Conclusões sobre os resultados obtidos para RLAM

Para a modelagem com os dados da unidade industrial foram utilizados os modelos que deram melhores resultados para a planta piloto de FCC: Rede neural com método LM usando duas diferentes técnicas de generalização (validação cruzada e regularização Bayesiana) e rede neural com os escores obtidos do conjunto de treinamento pela análise dos componentes principais (PCA-RNA).

Antes das modelagens foi feita a análise exploratória dos dados na busca de possíveis *outliers* e detecção de variáveis correlacionáveis. Os resultados indicaram a necessidade de remover 67 amostras do conjunto de dados e não houve retirada de nenhuma variável de entrada do processo.

Na modelagem global da unidade industrial (U-39 da RLAM) o modelo PCA-RNA forneceu os melhores resultados para a maioria das variáveis de saída. Neste caso, a PCA reduziu a dimensionalidade na entrada da rede fazendo com que a complexidade do modelo fosse reduzida também. Bons resultados foram obtidos para as vazões de gasolina, GLP, LCO, gás combustível e gás ácido. Resultados não satisfatórios foram obtidos para as vazões de propeno, propano e óleo decantado.

Já para a modelagem individual cada modelo aplicado teve resultados bons para algumas variáveis e ruins para outras. O modelo RNA-RB (regularização) teve melhores resultados para as vazões de gás ácido (VGA) e óleo decantado (VOD), o modelo RNA-VC (validação cruzada) teve melhores resultados para as vazões de gás combustível (VGC) e gasolina (VGAS) e o modelo PCA-RNA teve melhores resultados para as vazões de GLP (VGLP), LCO (VLCO), propano (VPA) e propeno (VPE). Os modelos individuais trouxeram melhoria na predição das variáveis de saída, porém as variáveis vazão de propano (VPA) e vazão de propeno (VPE) ainda tiveram predições abaixo do esperado.

SUGESTÕES PARA TRABALHOS FUTUROS

A seguir, são dadas algumas sugestões para dar continuidade ao trabalho aqui desenvolvido:

1- Realização de novos testes na planta piloto de FCC incorporando variáveis que não foram incluídas neste trabalho porque permaneceram constantes. Dessa forma, o modelo de redes neurais poderá ser novamente usado para treinamento e validação de um conjunto de dados mais representativo do processo.

2- Os resultados obtidos para a planta piloto de FCC da SIX podem ser modelados usando a técnica PCA-RNA com regularização Bayesiana. Os resultados apresentados na Seção 5.2.3 referem-se ao modelo PCA-RNA com validação cruzada. Isto pode trazer melhorias ao modelo individual.

3- Para unidade industrial de FCC (RLAM U-39) é sugerido modelar com outras ferramentas as variáveis VPA e VPE para se ter resultados mais satisfatórios, como por exemplo o PLS com redes neurais que não foi usado nesta parte do estudo. Pode-se ainda fazer novos treinamentos com os modelos aqui aplicados modificando outros fatores tais como: acrescentar mais neurônios na camada oculta, adição de mais uma camada oculta, etc.

4- De posse dos modelos que representem bem todas as variáveis do processo, a próxima etapa é a otimização do mesmo. NASCIMENTO *et al.* (2000) desenvolveram um modelo baseado em redes neurais artificiais para o processo industrial de polimerização do nylon. Com o modelo pronto, foi utilizado um otimizador na busca de condições operacionais que fossem capaz de aumentar a

produção do polímero. Segundo os autores esta abordagem é confiável, evita problemas numéricos típicos de otimização convencional e não consome muito tempo computacional.

REFERÊNCIAS BIBLIOGRÁFICAS

- ABADIE, E., "Craqueamento Catalítico", Petrobras S.A., 2. Versão, Rio de Janeiro (1997).

- ADEBIYI, O. A., CORRIPIO, A. B., "Dynamic Neural networks partial least squares (DNNPLS) identification of multivariate processes", *Computers and Chemical Engineering*, 27, 143-155 (2003).

- ALARADI, A. A., ROHANI, S., "Identification and control of a riser-type FCC unit using neural networks", *Computers and Chemical Engineering*, 26, 401-421 (2002).

- BAFFI, G., MARTIN, E. B., MORRIS, A. J., "Non-linear projection to latent structures revisited (the neural network PLS algorithm)", *Computers and Chemical Engineering* 23, 1293-1307, (1999).

- BARTHUS, R. C., "Aplicação de métodos quimiométricos para análise de controle de qualidade de óleos vegetais utilizando espectroscopia no infravermelho e Raman", Dissertação de Mestrado, Instituto de Química, UNICAMP (1999).

- BATISTA, L. M. F. L., "Desenvolvimento de software usando modelos determinísticos e redes neurais para o processo de craqueamento catalítico", Tese de Doutorado, Faculdade de Engenharia Química, UNICAMP (1996).

- BOLLAS, G. M., PAPADOKONSTADAKIS, S., MICHALOPOULOS, J., ARAMPATZIS, G., LAPPAS, A. A., VASSALOS, I. A., LYGEROS, A., "Using hybrid neural networks in scaling up na FCC model from a pilot plant to an industrial unit", *Chemical Engineering and Processing*, 42, 697-713, (2003).

- BORGES, J. A., "Modelagem e simulação de uma unidade de reforma catalítica da nafta utilizando redes neurais", Dissertação de Mestrado, Escola Politécnica da Universidade de São Paulo, USP (2001).

- BOLLAS G. M. , PAPADOKONSTANTAKIS S., MICHALOPOULOS J., ARAMPATZIS G., LAPPAS A. A., VASALOS I. A., LYGEROS A., "A computer-aided tool for the simulation and optimization of the combined HDS-FCC processes", *Chemical Engineering Research & Design* 82 (A7): 881-894, july (2004).

- BRAGA, A. P., LUDERMIR, T. B., CARVALHO, A. C. P. L. F., "Redes Neurais Artificiais – Teoria e Prática", LTC S.A., Rio de Janeiro (2000).

- BEEBE, K. R., PELL, R. J., SEASHOLTZ, M. B., "Chemometrics: A Pratical Guide", Wiley-Intercience Publication (1998).

- BULSARI, A. B., "Neural Networks for Chemical Engineers", Elsevier Science, Netherlands (1995).

- CHAPRA, S. C., CANALE, R. P., "Numerical Methods for Engineers: with software and programming", 4. ed., MacGraw-Hill, New York (2002).

- CHRISTENSEN, G., APELIAN, M. R., HICKEY, K. J., JAFFE, S. B., 1999, "Future directions in modelling the FCC process: An emphasis on product quality", *Chem Eng Science*, 54: 2753-2764.

- DECROOCQ, D., "Catalytic cracking of heavy petroleum fractions", Gulf Publishing Co., (1984).

- DEMUTH, H., BEALE, M., "Neural Network Toolbox – For use with Matlab, version 4", The MathWorks Inc. (2002).

- DESPAGNE, F., MASSART, D. L., "Neural Network in multivariate calibration", *Analyst*, 123, 157-178 (1998).

- DEWACHTERE, N. V, FROMENT, G. F, VASSALOS, I., MARKATOS, N., SKANDALIS, N., "Advanced modelling of riser-type catalytic cracking reactors", *Applied Thermal Engineering*, 17: 837-844 (1997).

- DONG, D., MCAVOY, T. J., "Nonlinear Principal Component Analysis – Based on Principal Curves and Neural Networks", *Computers and Chemical Engineering*, Vol. 27, n. 1, 65-78 (2003).

- EDGAR, T. F., HIMMELBLAU, D. M., LASDON, L. S., "Optimization of Chemical Processes", 2. ed., MacGraw-Hill, New York (2001).

- FERREIRA, M. M. C., ANTUNES, A. M., MELGO, M. S., VOLPE, P. L. O., "Quimiometria I: Calibração Multivariada, um Tutorial", *Química Nova*, vol. 22, n. 5, 724-731 (1999).

- FINSCHI L., "An Implementation of the Levenberg-Marquardt Algorithm", *TechnicalReport*, April (1996),
URL: http://www.ifor.math.ethz.ch/staff/finschi/Papers/LevMar.

- FISCHER, G. A., "Modelagem Matemática da Extração do Citocromo B5 via Redes Neurais", *Dissertação de Mestrado*, Faculdade De Engenharia Química – UNICAMP (2002).

- FORESEE, F. D., HAGAN, M. T., "Gauss-Newton approximation to Bayesian regularization," *Proceding of the 1997 International Joint Conference on Neural Networks*, 1930-1935 (1997).

- HAGAN, M. T., MENHAJ, M. B., "Training Feedforward Networks with the Marquardt Algorithm", *IEEE Transaction on Neural Networks*, Vol. 5, n° 6, 989 – 993, november (1994).

- HAGAN M.T., DEMUTH H.B. & BEALE M., "Neural Network Design", PWS Publishing Company (1995).

- HANU, A., SOHRAB, R., "Dynamic modelling and simulation of a riser-type fluid catalytic cracking unit", *Chem. Eng. Technology*, 20, 118- 130 (1997).

- HAYKIN, S., "Redes Neurais, Princípios e Práticas", 2. ed., Bookman, Porto Alegre (2001).

- HOLCOMB, T. R., MORARI, M., "PLS/neural networks", *Computers and Chemical Engineering*, Vol.16, n. 4, 393-411 (1992).

- HOSKINS, J. C., HIMMELBLAU, D. M., "Artificial Neural Networks Models of Knowledge Representation in Chemical Engineering", *Computers and Chemical Engineering*, Vol. 12, 881-890 (1988).

- JIA, F., MARTIN, E. B., MORRIS, A. J., "Nonlinear Principal Analysis for Process Faut Detection", *Computers and Chemical Engineering*, 22, 5851-5854 (1998).

- JOSEPH, B., WANG, F. H., SHIEH, D. S. S., "Exploratory Data Analyisis: A Comparation of Statistical Methods with Artificial Neural Networks", *Computers and Chemical Engineering*, Vol.16, n. 4, 413-423 (1992).

- KOURTI, T., MACGREGOR, J. F., "Multivariate SPC methods for process and product monitoring", *Journal of Quality Technology*, Vol. 28, n. 4, 409-428 (1996).

- KOURTI, T., MACGREGOR, J. F., "Process analysis, monitoring and diagnosis, using multivariate projection methods", *Chemometrics and Intelligent laboratory Systems*, 28, 3-21 (1995).

- KOLKANI, S. G., CHAUDHARY, A. K., NANDI, S., TAMBE, S. S., KULKARNI, B. D., "Modelling and Monitoring of Batch Processes Using Principal Component Analysis (PCA) Assisted Generalized Regression Neural Networks (GRNN)", *Biochemical Engineering Journal*, 18, 193-210 (2004).

- KODERLA, P., MOKANEK, T., "Comparation of two methods for the analysis of composite material.", *Journal of Materials Processing Technology*, 106, 87-93 (2000).

- KUMAN, S., CHADHA, A., GUPTA, R., SHARMA, R., "CATCRAK: A process simulator for an integrated FCC-regenerator system", *Ind EngChem Res*, 34: 3737-3748 (1995).

- LAQQA - LABORATÓRIO DE QUIMIOMETRIA EM QUÍMICA ANALÍTICA, (2004), URL: http://laqqa.iqm.unicamp.br

- MACKAY, D. J. C., "Bayesian Interpolation", *Neural Computation*, v. 4, 415-447 (1992).

- MALTHOUSE, E. C., TAMHANE, A. C., MAH, R. S. H., "Nonlinear Partial Least Squares", *Computers and Chemical Engineering*, Vol.21, n. 8, 875-890 (1997).

- MCGREAVY, C., LU, M. L., WANG, X. Z., KAM, E. K. T., "Characterization of the behaviour and product distribution in fluid catalytic cracking using neural networks", *Chem Eng Sci*, 49 (24A): 4717-4724 (1994).

- MCKEOWN, J. J., STELLA, F., HALL, G., "Some Numerical Aspects of the Training Problem for Feed-Forward Neural Nets", *Neural Networks*, vol. 10, n° 8, pp.1455-1463 (1997).

- MICHALOPOULOS, J., PAPADOKONSTADAKIS, S., ARAMPATZIS, G., LYGEROS, A., "Modelling of an industrial fluid catalytic cracking unit using neural networks", *Trans.IchemE.*, vol. 79, part A, 137-142, (2001).

- MOLLER, M. F., "A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning", *Neural Networks*, vol. 6, pp. 525-533 (1993).

- NARENDRA, K. S., PATHASARATHY K., "Identification and Control of Dynamical Systems Using Neural Networks", *IEEE Trans. on Neural Networks*, vol. 1, n. 1, pp. 4-27 (1990).

- Nascimento, C. A. O., Giudici, R., Guardani, R., "Neural network based approach for optimization of industrial chemical processes", *Computers & Chemical Engineering*, vol. 24, Issues 9-10, 2303-2314 (2000)

- JACOB, S. M., GROSS, B., VOLTZ S. E., WEEKMAN, V. W., 1976, A lumping and reaction scheme for catalytic cracking, *AIChEJ*, 22 (4): 701-713.

- OLIVEIRA, K. P. S., "Aplicação das técnicas de Redes Neurais e de análise de componentes principais na modelagem de uma lagoa aerada da RIPASA S/A", Tese de Mestrado, Faculdade de Engenharia Química, Universidade Estadual de Campinas (2000).

- OLIVEIRA-ESQUERRE, K. P. S. , "Aplicação das técnicas estatísticas multivariadas e de redes neurais na modelagem de um sistema de tratamento de efluentes industriais", Tese de Doutorado, Faculdade de Engenharia Química, Universidade Estadual de Campinas (2003).

- OTTO, M., "Chemometrics - Statistics and Computer Application in Analytical Chemistry", Wiley-VCH, Weinheim (1999).

- PRECHELT L., "Automatic Early Stopping Using Cross Validation: Quantifying the Criteria", *Neural Networks*, vol. 11, n° 4, pp. 761-767 (1998).

- PRECHELT L., "Early Stopping - but when?", *Technical Report* (1997), URL: http://wwwipd.ira.uka.de/~prechelt/Biblio/stop_tricks1997.ps.gz

- QIN, S. J., MCAVOY, T. J., "Nonlinear PLS modeling using neural networks", *Computers and Chemical Engineering*, 16, 379-391 (1992).

- RIBEIRO, F. A. L., "Aplicação de métodos de análise multivariada no estudo de hidrocarbonetos policíclicos aromáticos Dissertação de Mestrado, Instituto de Química, Universidade Estadual de Campinas (2001).

- RUMELHART D.E., MCCLELLAND J.L., AND THE PDP RESEARCH GROUP. "Parallel Distributed Processing: Exploration in the Microstructure of Cognition, vol. 1. MIT Press, Cambridge, Massachussetts (1986).

- SADEGHBEIGI, R., "Fluid Ctalytic Cracking Handbook – Design, Operatio and Troubleshooting of FCC Facilities", Gulf Publishing Company, Houston (1995).

- SANTEN, A., KOOT, G. L. M., ZULLO, L. C., "Statistical Data Analysis of a Chemical Plant", *Computers and Chemical Engineering*, Vol.21, Suppl., S1123-S1129 (1997).

- SANTOS, V. M. L., "Controle Preditivo Baseado em Redes Neurais: Aplicação a Unidades de Craqueamento Catalítico", Dissertação de Mestrado, Departamento de Engenharia Química, UFPE (2000).

- SARLE W. S., "Neural Networks and Statistical Models", *Proceedings of the Nineteenth Annual SAS Users Group International Conference*, (1994), URL:ftp://ftp.sas.com/pub/sugi19/neural/ neural1.ps.

- SARLE W. S., "Stopped Training and Other Remedies for Overfitting", *Proceedings ofthe Symposium on the Interface*, (1995), URL:ftp://ftp.sas.com/pub/neural/inter95.ps.Z.

- SILVA, L. N. C., "Análise e síntese de estratégia de aprendizado para redes neurais artificiais", Dissertação de Mestrado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas (1998).

- SWIGLER, K., "Applying Neural Networks: A Pratical Guide", Londres: Academic Press (1996).

- THEOLOGOS, K. N., NIKOU, I. D., LYGEROS, A. I., MARKATOS, N. C., "Simulation and Design of Fluid Catalytic Cracking Riser – Type Reactors", *Computers and Chemical Engineering*, Vol. 20, Suppl., S757-S762 (1996).

- TURNER, P., MONTAGUE, G., MORRIS, J., "Nonlinear and Direction-dependent dynamic process modelling using neural networks", *IEEE Proc.-Control Theory Appl.*, vol. 143, n. 1, 44-48, january (1996).

- VAN DER SMAGT, P., "Minimization Methods for Training Feedforward Neural networks," *Neural Networks*, vol 1, n° 7 (1994)

- VAN LANDEGHEM, F., NEVICATO, D., PITAULT, I., FORISSIER, M., TURLIER, P., DEROUIN, C., BERNARD, J. R., "Fluid Catalytic Cracking: Modelling of an Industrial Riser", *Applied Catalysis A: General*, 138, 381-405 (1996).

- VIEIRA,W. G., "FCC: Controle Preditivo e Identificação Via Redes Neurais", Tese de Doutorado, Faculdade de Engenharia Química, UNICAMP (2002).

- WISE, B. M., GALLAGHER, N. B., BRO, R., SHAVER, J. M., "PLS_Toolbox 3.0 - for use with MATLAB", Eigenvector Research , Inc. (2003).

- WOLD, S., ESBENSEN, K., GELADI, P., "Principal Component Analysis", *Chemometrics and Intelligent Systems*, 2, 37-52 (1987).

- WOLD, S., SJÖSTRÖM, M., ERIKSSON, L., "PLS-regression: A Basic Tool of Chemometrics", *Chemometrics and Intelligent laboratory Systems*, 58, 109-130 (2001).

- ZUPAN, Z., GASTEIGER, J., "Neural Networks for Chemists – An Introduction", VHC - New York (1993).

- ZHANG, J., MARTIN, E. B., MORRIS, A. J., "Process Monitoring Using Non-linear Statistical Techniques", *Chemical Engineering Journal*, 67, 181-189 (1997).

APÊNDICE A

RESULTADOS DA MODELAGEM INDIVIDUAL FEITA PARA OS DADOS DA RLAM

A seguir são mostradas as tabelas contendo os valores de *RMSE* nas etapas de treinamento e validação para cada método estudado na Seção 5.4.

Tabela A.1: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGC, VGA, VPE e VPA. Método LM com validação cruzada.

	RMSE											
	VGC			VGA			VPE			VPA		
	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste
1	0,4212	0,3824	0,3945	0,6157	0,5859	0,6014	0,7539	0,8324	0,8108	0,5990	0,6273	0,6072
2	0,3394	0,3427	0,3744	0,4562	0,5230	0,5604	0,6279	0,8219	0,8288	0,7182	0,6352	0,6963
3	0,2618	0,2761	0,3009	0,4330	0,5064	0,5535	0,5776	0,7575	0,7839	0,5092	0,6033	0,6559
4	0,3111	0,3145	0,3283	0,4059	0,4903	0,5586	0,7236	0,7965	0,8015	0,6491	0,6382	0,6349
5	0,2901	0,3045	0,3599	0,4809	0,5022	0,5903	0,7111	0,7426	0,8180	0,5740	0,6575	0,6401
6	0,1781	0,2910	0,4779	0,3942	0,5068	0,5261	0,3778	0,7945	0,7428	0,5096	0,6352	0,6120
7	0,1953	0,2636	0,3124	0,4234	0,4905	0,5140	0,5399	0,7722	0,7745	0,5527	0,6481	0,6082
8	0,1943	0,2776	0,3652	0,3072	0,4871	0,5858	0,4627	0,7882	0,8585	0,5146	0,6286	0,5835
9	0,1941	0,2780	0,2893	0,3880	0,4956	0,5173	0,5961	0,7847	0,7485	0,5326	0,6135	0,6090
10	0,1679	0,2799	0,2579	0,3715	0,5006	0,5575	0,4849	0,8071	0,7624	0,5156	0,6319	0,6388
11	0,2295	0,2944	0,3097	0,4262	0,5013	0,5709	0,4998	0,7910	0,8785	0,3752	0,6524	0,7823
12	0,3596	0,2987	0,3209	0,3294	0,4932	0,5383	0,4314	0,7876	0,7124	0,5421	0,6196	0,6245
13	0,1857	0,2902	0,2784	0,3460	0,4834	0,5696	0,5582	0,7761	0,7556	0,4620	0,6573	0,6229
14	0,1277	0,2987	0,3671	0,3739	0,4783	0,6195	0,4469	0,7927	0,8020	0,4491	0,6502	0,6534
15	0,2616	0,3233	0,4391	0,3888	0,4870	0,5430	0,3346	0,7729	0,8046	0,3595	0,6502	0,7878
16	0,1518	0,2505	0,2446	0,3054	0,4706	0,5781	0,4038	0,7908	0,7867	0,4217	0,6550	0,6957
17	0,1604	0,3059	0,2708	0,2693	0,4729	0,5934	0,4093	0,7601	0,7997	0,5024	0,6489	0,6706
18	0,2277	0,2565	0,2924	0,4337	0,5122	0,5762	0,4573	0,7823	0,7990	0,4971	0,6179	0,6555
19	0,1584	0,2896	0,2777	0,3839	0,5049	0,6053	0,5344	0,7871	0,6488	0,5417	0,6577	0,6529
20	0,1594	0,3348	0,3682	0,3697	0,4943	0,5019	0,5232	0,7941	0,7403	0,4130	0,6521	0,7974

Tabela A.2: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGLP, VGAS, VLCO e VOD. Método LM com validação cruzada.

	RMSE											
	VGLP			VGAS			VLCO			VOD		
	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste
1	0,4973	0,5083	0,5651	0,6071	0,5139	0,6074	0,6081	0,6380	0,5983	0,6794	0,6332	0,6760
2	0,4076	0,4607	0,5412	0,3715	0,4000	0,3960	0,5201	0,5739	0,5922	0,5394	0,5814	0,6472
3	0,3538	0,4799	0,5200	0,3220	0,3620	0,4271	0,4579	0,5519	0,6070	0,5235	0,5464	0,6650
4	0,3367	0,4311	0,4470	0,2884	0,3309	0,4352	0,4732	0,5031	0,6550	0,5317	0,5348	0,6009
5	0,2784	0,4126	0,4339	0,2411	0,3417	0,4710	0,4065	0,5363	0,5980	0,4040	0,5140	0,6447
6	0,3665	0,4708	0,4896	0,3190	0,3717	0,4070	0,4431	0,5396	0,5607	0,5108	0,5096	0,6434
7	0,3524	0,4509	0,5368	0,2986	0,3418	0,3972	0,3922	0,5142	0,6216	0,3381	0,5060	0,6209
8	0,2325	0,4547	0,5086	0,2400	0,3492	0,4221	0,4327	0,5117	0,5976	0,3924	0,4821	0,6301
9	0,3367	0,4567	0,4973	0,2687	0,3264	0,3876	0,3839	0,4912	0,5780	0,2515	0,5500	0,7506
10	0,2616	0,4541	0,5544	0,2901	0,3231	0,3694	0,4472	0,5370	0,6034	0,3329	0,5065	0,6327
11	0,3379	0,4358	0,4675	0,2197	0,3260	0,3557	0,4095	0,5101	0,5434	0,2891	0,5151	0,6583
12	0,3716	0,4851	0,4769	0,2093	0,3695	0,4890	0,4130	0,5359	0,5518	0,2947	0,4961	0,6605
13	0,2430	0,4533	0,4075	0,1738	0,3453	0,3970	0,3488	0,5552	0,5944	0,3961	0,4794	0,6184
14	0,3051	0,4720	0,5362	0,1854	0,3070	0,3797	0,4353	0,5374	0,5521	0,4036	0,4912	0,5762
15	0,3619	0,4770	0,5208	0,1545	0,3425	0,3780	0,3841	0,5450	0,5375	0,4264	0,5096	0,5980
16	0,2428	0,4949	0,4859	0,1671	0,3327	0,3692	0,3559	0,5387	0,5624	0,2831	0,5189	0,5454
17	0,2887	0,4644	0,5236	0,2582	0,3300	0,3519	0,4713	0,5423	0,5448	0,3259	0,4955	0,6321
18	0,4369	0,4715	0,4991	0,2257	0,3510	0,3643	0,4733	0,5648	0,5806	0,1984	0,4849	0,6711
19	0,1683	0,4769	0,4820	0,2225	0,3166	0,4553	0,4594	0,5491	0,5896	0,1988	0,4912	0,6033
20	0,2165	0,4941	0,5150	0,2457	0,3363	0,3854	0,3357	0,5170	0,6393	0,2992	0,5150	0,6183

Tabela A.3: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGC, VGA, VPE e VPA. Método LM com regularização Bayesiana.

	RMSE											
	VGC			VGA			VPE			VPA		
	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste
1	0,4297	0,3909	0,3983	0,6285	0,6008	0,6040	0,7890	0,8662	0,7720	0,5955	0,6363	0,6018
2	0,3439	0,3281	0,3824	0,4662	0,5006	0,5561	0,6224	0,8414	0,7565	0,5912	0,6176	0,6126
3	0,3110	0,3485	0,3238	0,4446	0,4922	0,5260	0,5567	0,8075	0,6911	0,5860	0,6298	0,6144
4	0,2417	0,3161	0,3196	0,3865	0,4698	0,5626	0,5096	0,8089	0,7110	0,5886	0,6304	0,6125
5	0,2262	0,3164	0,2946	0,3344	0,4817	0,5509	0,4426	0,7748	0,8057	0,5875	0,6298	0,6147
6	0,1654	0,3125	0,2985	0,3586	0,4689	0,5424	0,4167	0,7654	0,7829	0,5869	0,6280	0,6169
7	0,1471	0,3308	0,2943	0,3369	0,4657	0,5470	0,4098	0,7644	0,7806	0,5830	0,6282	0,6176
8	0,2359	0,3262	0,3064	0,3428	0,4690	0,5341	0,4162	0,7659	0,7800	0,5848	0,6284	0,6162
9	0,1787	0,3104	0,2821	0,3138	0,4708	0,5566	0,4138	0,7647	0,7946	0,5832	0,6290	0,6162
10	0,1965	0,3093	0,2824	0,3115	0,4667	0,5523	0,3768	0,7790	0,7921	0,5849	0,6282	0,6179
11	0,1810	0,3020	0,2865	0,3280	0,4696	0,5658	0,4085	0,7689	0,7845	0,5804	0,6296	0,6167
12	0,1512	0,3243	0,2796	0,3143	0,4754	0,5617	0,4547	0,7775	0,7622	0,5908	0,6310	0,6148
13	0,1405	0,3304	0,3092	0,3127	0,4732	0,5745	0,3642	0,7830	0,8165	0,5726	0,6156	0,6138
14	0,1493	0,3329	0,2912	0,3056	0,4712	0,5588	0,3873	0,7889	0,7574	0,5829	0,6305	0,6171
15	0,1488	0,2970	0,2899	0,3456	0,4745	0,5727	0,4452	0,8007	0,7343	0,5807	0,6282	0,6184
16	0,1282	0,3029	0,2954	0,3271	0,4704	0,5650	0,4903	0,8110	0,7254	0,5755	0,6326	0,6186
17	0,2173	0,3158	0,2726	0,3201	0,4783	0,5677	0,4524	0,7897	0,7571	0,5699	0,6316	0,6162
18	0,1283	0,3184	0,3127	0,3299	0,4743	0,5739	0,5053	0,8157	0,7000	0,5731	0,6347	0,6153
19	0,1416	0,3098	0,3005	0,3412	0,4745	0,5528	0,5005	0,8097	0,7191	0,5753	0,6284	0,6197
20	0,1813	0,3103	0,2778	0,3262	0,4676	0,5700	0,4263	0,7820	0,7689	0,5798	0,6280	0,6186

Tabela A.4: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGLP, VGAS, VLCO e VOD. Método LM com regularização Bayesiana.

	RMSE											
	VGLP			VGAS			VLCO			VOD		
	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste
1	0,4874	0,5523	0,5674	0,5784	0,5245	0,5715	0,6422	0,6485	0,6090	0,6709	0,6462	0,6860
2	0,3993	0,4890	0,4977	0,4175	0,3746	0,5204	0,5176	0,5688	0,6174	0,5591	0,5819	0,6474
3	0,3587	0,4667	0,5102	0,2922	0,3413	0,3551	0,4574	0,5702	0,5749	0,4610	0,5873	0,6517
4	0,3174	0,4368	0,4751	0,3072	0,3480	0,3548	0,3920	0,5365	0,6050	0,4587	0,5701	0,6403
5	0,2806	0,4686	0,4924	0,2524	0,3425	0,3652	0,3768	0,5420	0,5941	0,3725	0,5489	0,6211
6	0,3141	0,4639	0,4793	0,2784	0,3350	0,3584	0,3760	0,5437	0,5956	0,2868	0,5077	0,5845
7	0,2474	0,4681	0,5071	0,2297	0,3278	0,3818	0,3730	0,5394	0,5929	0,3383	0,5225	0,6146
8	0,3017	0,4584	0,4927	0,2342	0,3418	0,4263	0,3108	0,5435	0,5906	0,2851	0,5346	0,5975
9	0,3631	0,4632	0,4783	0,2833	0,3268	0,4084	0,3662	0,5513	0,5971	0,2499	0,5208	0,6373
10	0,2562	0,4687	0,4901	0,1974	0,3294	0,3959	0,3743	0,5538	0,5882	0,3704	0,5412	0,5992
11	0,2461	0,4774	0,4915	0,2993	0,3287	0,4028	0,3114	0,5530	0,6277	0,2677	0,5210	0,6611
12	0,3643	0,4619	0,4580	0,2979	0,3225	0,4250	0,4106	0,5622	0,6098	0,2402	0,4933	0,6118
13	0,3269	0,4687	0,4698	0,2027	0,3286	0,3990	0,4208	0,5625	0,5935	0,2062	0,5154	0,6799
14	0,3999	0,4591	0,4704	0,2409	0,3101	0,4094	0,4417	0,5639	0,6045	0,3240	0,5353	0,6261
15	0,3062	0,4590	0,4844	0,2118	0,3351	0,4318	0,4349	0,5637	0,6095	0,3096	0,5181	0,6148
16	0,3340	0,4714	0,4629	0,2767	0,3294	0,4228	0,4425	0,5569	0,6016	0,3092	0,5347	0,6010
17	0,3256	0,4802	0,5047	0,2060	0,3271	0,4015	0,4493	0,5580	0,5995	0,3301	0,5433	0,6593
18	0,3597	0,4763	0,4616	0,1872	0,3183	0,3893	0,4454	0,5593	0,6041	0,2937	0,5220	0,6505
19	0,3383	0,4495	0,4476	0,2328	0,3357	0,3797	0,4666	0,5583	0,5829	0,3135	0,5335	0,5833
20	0,3303	0,4680	0,4661	0,2308	0,3272	0,4104	0,4081	0,5609	0,5895	0,2975	0,5132	0,6260

Tabela A.5: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGC, VGA, VPE e VPA. Método PCA-RNA.

	RMSE											
	VGC			VGA			VPE			VPA		
	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste
1	0,5344	0,4597	0,6057	0,6578	0,6582	0,7115	0,7283	0,7921	0,6675	0,6158	0,5582	0,5113
2	0,3999	0,4025	0,5975	0,5552	0,6177	0,6594	0,6483	0,7460	0,6625	0,4966	0,5539	0,5157
3	0,3108	0,3797	0,5728	0,4843	0,6054	0,6609	0,5842	0,7129	0,6343	0,5022	0,5754	0,4969
4	0,2731	0,3178	0,5275	0,4326	0,5704	0,5988	0,5759	0,7016	0,6043	0,5088	0,5744	0,4925
5	0,2641	0,3110	0,5067	0,4428	0,5722	0,6000	0,5693	0,7033	0,6208	0,5040	0,5734	0,4889
6	0,2210	0,3075	0,4957	0,3405	0,5411	0,6145	0,5837	0,7057	0,6159	0,5113	0,5687	0,4718
7	0,2369	0,3104	0,4814	0,3885	0,5647	0,5930	0,5764	0,7085	0,6524	0,4967	0,5732	0,4602
8	0,2273	0,2841	0,5160	0,3146	0,5096	0,6150	0,5871	0,7107	0,6363	0,5068	0,5666	0,4625
9	0,2476	0,3087	0,4669	0,3313	0,5497	0,5669	0,5339	0,7069	0,6125	0,5238	0,5685	0,4732
10	0,2339	0,2891	0,4601	0,3425	0,5466	0,5800	0,5835	0,7082	0,6502	0,4987	0,5678	0,4616
11	0,2360	0,3007	0,4546	0,3643	0,5450	0,6116	0,5663	0,7111	0,6488	0,5147	0,5584	0,4644
12	0,2546	0,3069	0,5074	0,3032	0,5357	0,5956	0,5800	0,7127	0,6330	0,5230	0,5609	0,4643
13	0,2648	0,3001	0,5252	0,3801	0,5755	0,6161	0,5449	0,7054	0,6032	0,5121	0,5526	0,4649
14	0,1872	0,2840	0,4556	0,3266	0,5460	0,5791	0,6165	0,7205	0,6185	0,5178	0,5602	0,4642
15	0,1840	0,2954	0,4693	0,3245	0,5526	0,6233	0,5791	0,7101	0,6444	0,5079	0,5596	0,4519
16	0,1566	0,2980	0,4819	0,4458	0,5733	0,5907	0,5622	0,7100	0,6553	0,5227	0,5590	0,4641
17	0,2419	0,2982	0,5064	0,3922	0,5670	0,5699	0,6085	0,7213	0,6351	0,5145	0,5621	0,4575
18	0,2634	0,3006	0,5108	0,4600	0,5804	0,6064	0,5793	0,7163	0,6371	0,5122	0,5612	0,4566
19	0,2216	0,3125	0,5070	0,3095	0,5488	0,6327	0,5854	0,7095	0,6465	0,5158	0,5543	0,4644
20	0,2528	0,3004	0,5240	0,3685	0,5654	0,5761	0,5724	0,7082	0,6556	0,5099	0,5595	0,4563

Tabela A.6: *RMSE* de treinamento, validação e teste para as variáveis de saída: VGLP, VGAS, VLCO e VOD. Método PCA-RNA.

	RMSE											
	VGLP			VGAS			VLCO			VOD		
	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste	Treino	Valid	Teste
1	0,5792	0,5302	0,5261	0,6342	0,5080	0,7425	0,6421	0,6556	0,5534	0,7943	0,6972	0,9391
2	0,5161	0,4369	0,4661	0,4876	0,3889	0,5569	0,5740	0,5463	0,5236	0,6009	0,5959	0,6541
3	0,4107	0,4292	0,4263	0,3519	0,3861	0,4846	0,4565	0,5723	0,5502	0,5122	0,6094	0,6930
4	0,3940	0,4266	0,4007	0,3063	0,3350	0,4584	0,4370	0,5441	0,5507	0,5108	0,5832	0,6718
5	0,3236	0,3794	0,3930	0,3033	0,3311	0,4490	0,3897	0,5046	0,5649	0,4880	0,5796	0,7273
6	0,3533	0,3866	0,3687	0,2921	0,3293	0,4462	0,4220	0,5355	0,5409	0,4018	0,5631	0,7026
7	0,3976	0,4311	0,4117	0,2842	0,3275	0,4380	0,3826	0,5396	0,5880	0,4025	0,5782	0,6533
8	0,3461	0,4223	0,3939	0,2823	0,3415	0,4496	0,3808	0,5411	0,5895	0,3531	0,5368	0,6445
9	0,3912	0,4202	0,4325	0,2528	0,3262	0,4336	0,3694	0,5270	0,5413	0,3122	0,5673	0,6487
10	0,3057	0,4050	0,4001	0,2864	0,3438	0,4656	0,3791	0,5401	0,5881	0,3126	0,5598	0,6435
11	0,3328	0,3703	0,4411	0,2671	0,3221	0,4134	0,3880	0,5244	0,5287	0,2915	0,5218	0,6399
12	0,3007	0,3969	0,4232	0,2328	0,3281	0,4321	0,3750	0,5299	0,5573	0,3293	0,5623	0,6518
13	0,3769	0,3685	0,4025	0,3263	0,3479	0,4375	0,3978	0,5368	0,5357	0,2808	0,5161	0,6658
14	0,2542	0,3904	0,3706	0,2231	0,3363	0,4280	0,3676	0,5288	0,5497	0,3404	0,5651	0,6543
15	0,3753	0,3687	0,4028	0,2361	0,3497	0,4462	0,4560	0,5500	0,5346	0,3493	0,5908	0,7302
16	0,3236	0,3925	0,4250	0,2165	0,3507	0,3913	0,3969	0,5312	0,5402	0,3298	0,5619	0,6537
17	0,3007	0,3939	0,3760	0,2695	0,3442	0,4674	0,3954	0,5302	0,5382	0,3088	0,5512	0,6534
18	0,3486	0,3726	0,4164	0,2884	0,3281	0,4529	0,3677	0,5134	0,5266	0,3058	0,5651	0,6673
19	0,2697	0,3824	0,4052	0,3163	0,3631	0,4839	0,4274	0,5142	0,5252	0,3413	0,5476	0,6853
20	0,3836	0,4175	0,4266	0,1661	0,3460	0,4289	0,4538	0,5576	0,5228	0,3145	0,5765	0,6448