

UNIVERSIDADE ESTADUAL DO NORTE FLUMINENSE  
LABORATÓRIO DE ENGENHARIA E EXPLORAÇÃO DE PETRÓLEO

PROJETO ENGENHARIA  
DESENVOLVIMENTO DO SOFTWARE:  
AVALIAÇÃO DAS CONDIÇÕES EXPERIMENTAIS PARA  
SEPARAÇÃO DE COMPONENTES DO PETRÓLEO POR  
CROMATOGRAFIA LÍQUIDA UTILIZANDO PLANEJAMENTO  
FATORIAL  
TRABALHO DA DISCIPLINA PROGRAMAÇÃO PRÁTICA

Versão 1:  
JHONNY COSTA CARVALHO  
Prof. André Duarte Bueno

MACAÉ - RJ  
Versão 2018 - Semestre 2019/1

# Contents

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Escopo do problema . . . . .	1
1.2	Objetivos . . . . .	2
<b>2</b>	<b>Especificação</b>	<b>4</b>
2.1	Especificação do software - requisitos . . . . .	4
2.1.1	Definições da interface . . . . .	4
2.1.2	Entrada de dados . . . . .	4
2.2	Nome do sistema/produto . . . . .	5
2.2.1	Requisitos funcionais . . . . .	5
2.2.2	Requisitos não funcionais . . . . .	5
2.3	Casos de uso . . . . .	6
2.4	Diagrama de caso de uso geral . . . . .	6
2.5	Diagrama de caso de uso Específico . . . . .	7
<b>3</b>	<b>Elaboração</b>	<b>8</b>
3.1	Análise de domínio . . . . .	8
3.2	Formulação teórica . . . . .	8
3.2.1	Planejamentos fatoriais com ponto central . . . . .	9
3.2.2	Modelos empíricos em estudo de triagem . . . . .	10
3.2.3	Estimativa dos erros para os efeitos através de repetições . . . . .	12
3.2.4	Estimativa dos erros para os efeitos, sem repetições . . . . .	13
3.2.5	Estimativa dos erros para os efeitos e coeficientes a partir das repetições no ponto central . . . . .	14
3.2.6	Identificação dos efeitos e coeficientes significativos . . . . .	15
3.2.7	Avaliação do modelo . . . . .	16
3.3	Diagrama de pacotes – assuntos . . . . .	20
<b>4</b>	<b>AOO – Análise Orientada a Objeto</b>	<b>21</b>
4.1	Diagramas de classes . . . . .	21
4.1.1	Dicionário de classes . . . . .	22
4.2	Diagrama de seqüência – eventos e mensagens . . . . .	22

4.2.1	Diagrama de sequência geral . . . . .	23
4.3	Diagrama de comunicação – colaboração . . . . .	23
4.4	Diagrama de máquina de estado . . . . .	24
4.5	Diagrama de atividades . . . . .	25
<b>5</b>	<b>Projeto</b>	<b>27</b>
5.1	Projeto do sistema . . . . .	27
5.2	Projeto orientado a objeto – POO . . . . .	28
5.3	Diagrama de componentes . . . . .	29
5.4	Diagrama de implantação . . . . .	30
<b>6</b>	<b>Implementação</b>	<b>32</b>
6.1	Código fonte . . . . .	32
<b>7</b>	<b>Teste</b>	<b>48</b>
7.1	Teste 1: Descrição . . . . .	48
7.2	Teste 2: Resultados . . . . .	48
<b>8</b>	<b>Documentação</b>	<b>50</b>
8.1	Documentação do usuário . . . . .	50
8.1.1	Como instalar o software . . . . .	50
8.2	Documentação para desenvolvedor . . . . .	51
8.2.1	Dependências . . . . .	51
8.2.2	Como gerar a documentação usando doxygen . . . . .	51

# List of Figures

2.1	Diagrama de caso de uso – Caso de uso geral . . . . .	7
2.2	Diagrama de caso de uso específico . . . . .	7
3.1	Planejamento $2^2$ com ponto central. . . . .	10
3.2	Representação gráfica do valor p para um teste unilateral. . . . .	16
3.3	Diagrama de Pacotes . . . . .	20
4.1	Diagrama de classes . . . . .	22
4.2	Diagrama de seqüência . . . . .	23
4.3	Diagrama de comunicação. . . . .	24
4.4	Diagrama de máquina de estado para a classe CFPC22_02. . . . .	25
4.5	Diagrama de atividades do metodo minimos quadrados da classe CFP22_02. . . . .	26
5.1	Diagrama de componentes. . . . .	30
5.2	Diagrama de implantação . . . . .	31
7.1	Tela do programa mostrando xxx . . . . .	49
7.2	Tela do programa mostrando resultados . . . . .	49
8.1	Documentacao do doxgem . . . . .	51
8.2	Documentacao doxyen . . . . .	52
8.3	Documentacao do doc.gen . . . . .	52

# List of Tables

2.1	Tabela do planejamento Fatorial . . . . .	5
2.2	Exemplo de caso de uso . . . . .	6
3.1	Análise de variância para regressão múltipla . . . . .	19

# Chapter 1

## Introdução

Neste trabalho desenvolve-se o software para avaliação das melhores condições experimentais para a obtenção da fração de hidrocarbonetos saturados do petróleo por meio de planejamento fatorial com ponto central com o intuito de minimizar o uso de solventes e adsolventes e obter um maior rendimento na separação desta fração.

### 1.1 Escopo do problema

O petróleo bruto inalterado contém carbono ( $\sim 80 - 87\%$ ), hidrogênio ( $\sim 12 - 15\%$ ), enxofre ( $\sim 0,1-5\%$ ), oxigênio ( $0,1-4,5\%$ ) e nitrogênio ( $0,1 - 1,5\%$ ). Grande parte do nitrogênio, enxofre e oxigênio estão associados às resinas e asfaltenos e são frequentemente denominados compostos NSO. Outros elementos presentes no petróleo incluem vários metais, em particular níquel e vanádio, que geralmente estão presentes em pequenas quantidades. A composição do petróleo bruto contém em média 57% de hidrocarbonetos saturados (S), 29% de hidrocarbonetos aromáticos (A) e 14% de resinas (R) e asfaltenos (A) [Killops and Killops, 2013]. No entanto, esta composição média pode variar significativamente devido, por exemplo, à origem da matéria orgânica, à história térmica e às alterações secundárias ocorridas em reservatório após acumulação.

O petróleo contém também vários componentes lipídicos preservados, designados biomarcadores, os quais podem ser associados com compostos precursores biológicos, porque seus esqueletos básicos são preservados durante a diagênese e grande parte da catagênese. Esses biomarcadores podem estar presentes nas frações de hidrocarbonetos saturados e aromáticos, tais como os esteranos normais, hopanos e os esteróides aromáticos. Porém, geralmente representam menos de 1% em peso de óleos brutos inalterados [Killops and Killops, 2013].

A análise composicional confiável das frações do petróleo é muito importante porque fornece informações que afetam desde a exploração de petróleo até a produção, transporte e refino [Bissada et al., 2016]. Por exemplo, na análise do sistema petrolífero, em correlações com a rocha geradora, na previsão de reservatórios com potencial para precipitar as-

faltos, na resolução de problemas ambientais, dentre outros [Ashoori and Vanini, 2016]. Assim, para fornecer adequadamente estas informações efetua-se a separação dos componentes SARA para determinar a concentração relativa de cada grupo químico e obter frações de alta pureza para posteriores análises por cromatografia gasosa acoplada a espectrometria de massas (CG-EM) e análise isotópica, por exemplo [Bissada et al., 2016].

Vários procedimentos podem ser adotados para separação das frações SARA, tais como cromatografia em fase líquida clássica (CLC), cromatografia em camada delgada por detecção por ionização em chama (do inglês *thin layer chromatography - flame ionization detection*, TLC-FID), cromatografia líquida de alta performance (HPLC), dentre outros [Karlsen and Larter, 1991, Fan and Buckley, 2002, Jiang et al., 2008]. A CLC, apesar de ser uma técnica simples, é amplamente utilizada como etapa preliminar para análises instrumentais mais sensíveis como CG-EM especialmente quando não se dispõe de outros métodos de separação. Vale ressaltar, no entanto, que este método requer quantidades significativas de solventes, adsorventes e estes ainda podem interferir na adequada separação das frações presentes na matriz analisada [Collins et al., 2006].

Assim, aliada à necessidade da indústria, faz-se necessário a busca por processos que otimizem a utilização desses materiais (solventes orgânicos e adsorventes) usados na caracterização do petróleo. Uma ferramenta capaz de determinar melhores condições experimentais com consequente diminuição da quantidade de solvente utilizado é o planejamento fatorial [Brasil and Araújo, 2014].

Dada a importância desta etapa para análises posteriores, uma vez que a separação das frações do petróleo pode ser influenciada pelas quantidades de adsorvente e de solvente utilizados, este trabalho tem como objetivo o desenvolvimento de um software com o auxílio de planejamento fatorial, com ponto central, determinar as melhores condições experimentais para a obtenção da fração de saturados do petróleo.

## 1.2 Objetivos

Os objetivos deste projeto de engenharia são:

- Objetivo geral:
  - Determinar, a partir do planejamento fatorial com ponto central, as melhores condições experimentais para a obtenção da fração de hidrocarbonetos saturados do petróleo utilizando a técnica de cromatografia em fase líquida clássica em coluna aberta, visando diminuir a quantidade de solvente e adsorvente utilizado, aumentar a porcentagem de recuperação da fração e, como consequência, reduzir a quantidade de resíduos líquidos e sólidos gerados ao meio ambiente
- Objetivos específicos:

- Calcular desvio padrao e variância;
- Utilizar o planejamento fatorial para avaliação preliminar dos parâmetros da análise SARA;
- Determinar as melhores condições para separação em coluna cromatográfica clássica a partir de Planejamento Composto Central e teste F;
- Plotar os resultados em forma de tabela e/ou gráficos para melhor compreensão.



# Chapter 2

## Especificação

Apresenta-se neste capítulo do projeto de engenharia a concepção, a especificação do sistema a ser modelado e desenvolvido.

### 2.1 Especificação do software - requisitos

Construir um software para fins acadêmicos que será livre, licença GNU/GPL v2.0, multiplataforma e orientado a objeto em C++, de modo a facilitar futuras modificações. Este será em modo texto, pois há pouca interação do usuário com o software. Inicialmente, o usuário definirá os parâmetros de entrada que já foram previamente obtidos em laboratório. Estes parâmetros serão inseridos manualmente via teclado ou pela leitura de um arquivo de texto (formato ASCII) usando essas informações o software irá calcular a variância, desvio padrão, teste t e outras estatísticas para avaliação dos coeficientes significativos. O software apresentará ao usuário valores de acordo com a ANOVA, os resultados serão exibidos na forma de gráficos e texto.

#### 2.1.1 Definições da interface

O software tem interface de entrada em modo texto e saídas em modo texto e gráfica.

#### 2.1.2 Entrada de dados

Entrada dos valores em porcentagens obtidos em laboratório com o primeiro nível de sílica(-) solvente (+), o segundo nível sílica(-) solvente (-), o terceiro nível sílica(+) solvente (+), o quarto nível sílica(+) solvente (-) e três últimos sendo o ponto central sílica(0) solvente (0).

Experimento	Sílica (g)	Solvente (ml)
1	1,0(-)	13,0(-)
2	1,0(-)	7,0(-)
3	2,0(+)	13,0(+)
4	2,0(+)	7,0(-)
5,6 e 7	1,5(0)	10,0(0)

Table 2.1: Tabela do planejamento Fatorial

## 2.2 Nome do sistema/produto

<b>Nome</b>	Avaliação das condições experimentais para separação de componentes do petróleo por cromatografia líquida utilizando planejamento fatorial 2 <sup>2</sup> .
<b>Componentes principais</b>	Caracterizar as amostras de óleos a partir dos parametros de entradas que informam a densidade e os saturados.
<b>Missão</b>	Otimizar o uso de solvente e adsolvente na identificação dos hidrocarbonetos, tendo em vista a preocupação com o meio ambiente e tornar o processo menos dispendioso.

### 2.2.1 Requisitos funcionais

Apresenta-se a seguir os requisitos funcionais.

<b>RF-01</b>	O usuário deverá ter liberdade para escolher o tipo de entrada de dados (teclado ou disco).
--------------	---

<b>RF-02</b>	Deve permitir o carregamento de arquivos criados pelo software
--------------	--

<b>RF-03</b>	O usuário terá disponível os resultados em gráficos e em arquivos de texto.
--------------	---

### 2.2.2 Requisitos não funcionais

<b>RNF-01</b>	Os cálculos devem ser feitos utilizando-se Porcentagem dos resultados obtidos.
---------------	--

<b>RNF-02</b>	O programa deverá ser multi-plataforma, podendo ser executado em <i>Windows</i> , <i>GNU/Linux</i> .
---------------	--

## 2.3 Casos de uso

Um caso de uso descreve um ou mais cenários de uso de software, exemplos de uso, como o sistema interage com usuários externos (atores). Ademais, ele deve representar uma sequência típica de uso do programa (a execução de determinadas tarefas padrão). Também deve representar as exceções, casos em que o usuário comete algum erro, em que o sistema não consegue realizar as tarefas solicitadas.

A Tabela 2.1 que exemplifica um caso de uso do sistema, bem como os diagramas de casos de uso

Table 2.2: Exemplo de caso de uso

Nome do caso de uso:	Cálculo do planejamento fatorial de $2^2$ com ponto central.
Resumo/descrição:	Geração de coeficientes correspondentes as suas variaves no qual informará se são ou não significativa para o processo de obtenção da fração de saturados.
Etapas:	<ol style="list-style-type: none"> <li>1. Calcular estimativas para y.</li> <li>2. Calcular os efeito dos coeficientes significativos.</li> <li>3. Calcular os erros.</li> <li>4. Calcular resíduos.</li> <li>5. Calcular t Student.</li> <li>6. Calcular F fischer</li> <li>7. Gerar gráficos.</li> <li>8. Analisar resultados.</li> </ol>
Cenários alternativos:	Um cenário alternativo envolve uma entrada errada do usuário (por exemplo, um planejamento fatorial de grau maior que 2.

## 2.4 Diagrama de caso de uso geral

O diagrama de caso de uso geral da Figura 2.1 mostra o usuário entrando com os valores de entradas após ter escolhido o tipo de Planejamento, com esses dados o software efetua os calculandos da ANOVA, posteriormente analisando os resultados.

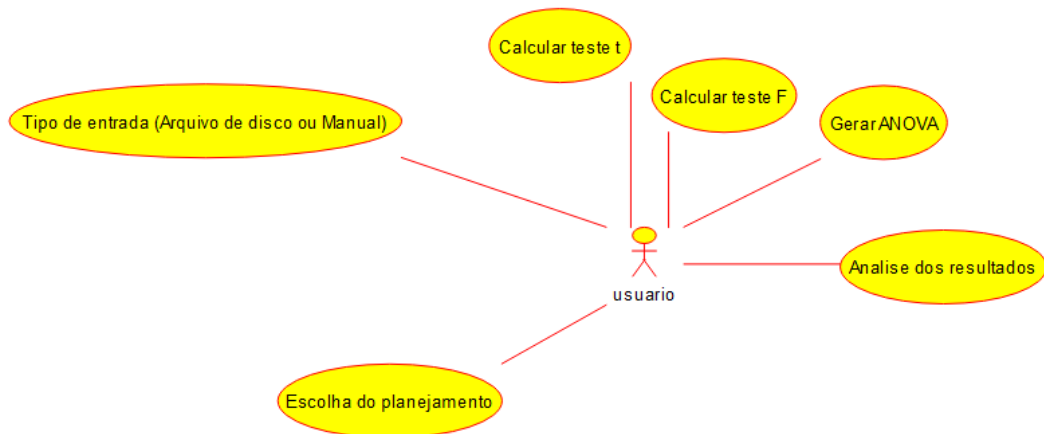


Figure 2.1: Diagrama de caso de uso – Caso de uso geral

## 2.5 Diagrama de caso de uso Específico

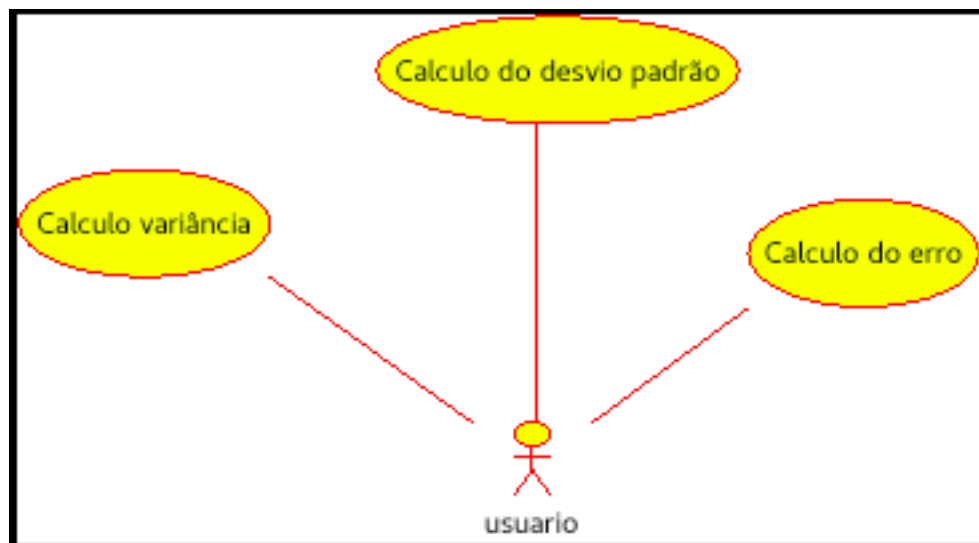


Figure 2.2: Diagrama de caso de uso específico

# Chapter 3

## Elaboração

Depois da definição dos objetivos, da especificação do software e da montagem dos primeiros diagramas de caso de uso, a equipe de desenvolvimento do projeto de engenharia passa por um processo de elaboração que envolve o estudo de conceitos relacionados ao sistema a ser desenvolvido, a análise de domínio e a identificação de pacotes.

Na elaboração fazemos uma análise dos requisitos, ajustando os requisitos iniciais de forma a desenvolver um sistema útil, que atenda às necessidades do usuário e, na medida do possível, permita seu reuso e futura extensão.

Eliminam-se os requisitos "impossíveis" e ajusta-se a idéia do sistema de forma que este seja flexível, considerando-se aspectos como custos e prazos.

### 3.1 Análise de domínio

- Área relacionada:
- A principal área relacionada a este projeto é a estática aplicada a Geoquímica do petróleo.
- Sub-área relacionada:
- Planejamento Fatorial

### 3.2 Formulação teórica

De acordo com Breitz et al., (2014), independente da área de pesquisa, é necessário conhecer as variáveis (ou fatores) que afetam o sistema em estudo e a extensão desta influência, a fim de melhorar o processo de uma maneira geral, com o objetivo de reduzir tempo e custos. Assim, no método multivariado realiza-se o menor número de experimentos necessários para extrair o máximo de informação dos dados coletados de modo a avaliar e/ou otimizar um produto ou processo.

Quando se realiza um planejamento fatorial o objetivo é estudar variáveis estatisticamente independentes, ou seja, aquelas que podem ter seus valores alterados e controlados de forma independente umas das outras ([Breitkreitz et al., 2014]). Assim, com a realização do planejamento fatorial pode-se determinar as quantidades de ensaios a serem realizados de acordo com os números de variáveis envolvidas no processo, é possível identificar as variáveis que apresentarão respostas significativas para o experimento, além de otimizar tempo, capital e reagentes químicos, gerando assim menos resíduos ao meio ambiente.

Adicionalmente, o planejamento fatorial avalia os parâmetros que têm efeitos significativos em seus resultados experimentais, avaliando simultaneamente as variáveis envolvidas no processo. Essas interações são as principais componentes de muitos processos de otimização, levando às condições de relevância dos parâmetros em estudo ([Teófilo et al., 2006]).

O tipo de planejamento fatorial que será utilizado é uma escolha do pesquisador, mas de forma geral, para um estudo inicial com o intuito de se fazer uma triagem de variáveis e identificar aquelas que possuem maior influência na resposta, sugere-se o uso de planejamentos fracionários ou completos ([Breitkreitz et al., 2014]).

O planejamento fatorial normalmente é adequado para estudar, de forma eficiente e econômica, o efeito conjunto de vários fatores sobre uma variável resposta de interesse.

O planejamento fatorial do tipo  $2^2$  é considerável o mais comum, por praticar menores números de níveis, no entanto, se mostra vantajoso, pois, indica tendências de resultados para análises de métodos posteriores ([Cunico et al., 2008]). Uma boa ferramenta de escolha após os resultados obtidos do planejamento fatorial é o gráfico de superfície de resposta, onde conseguimos obter o ponto máximo de obtenção das frações do petróleo através da cromatográfica líquida ([Teófilo et al., 2006]).

Quando se pretende otimizar custos e tempo de pesquisa é interessante o implemento do planejamento fatorial com ponto central. Através dele consegue-se estimar o desvio padrão dos resultados sem a necessidade de realizar repetições para cada resultado obtido. O cálculo da variância em experimentos feitos com ponto central se dá através da média das variáveis envolvidas ([Teófilo et al., 2006]).

O objetivo é determinar, a partir do planejamento fatorial com ponto central, as melhores condições experimentais para a obtenção da fração de hidrocarbonetos saturados do petróleo utilizando a técnica de cromatografia em fase líquida clássica em coluna aberta, visando diminuir a quantidade de solvente e adsorvente utilizado, aumentar a porcentagem de recuperação da fração e, como consequência, reduzir a quantidade de resíduos líquidos e sólidos gerados ao meio ambiente

### 3.2.1 Planejamentos fatoriais com ponto central

Em muitos casos, a realização de repetições autênticas pode ser algo inconveniente por diversas razões. Para contornar este infortúnio e obter uma boa estimativa dos erros, um

experimento é normalmente incluído no centro do planejamento, em que o valor médio dos níveis de todas as variáveis é empregado. São os conhecidos experimentos no ponto central (nível zero). Deste modo, é possível avaliar a significância dos efeitos ou coeficientes, tanto em planejamentos de triagem (completos ou fracionários) como em metodologias de superfície de resposta (discutidos mais adiante). Além desta vantagem, recomenda-se este tipo de experimento pelas seguintes razões: o risco de perder a relação não linear entre os intervalos é minimizado e é possível estimar um modelo razoável e verificar se há falta de ajuste.

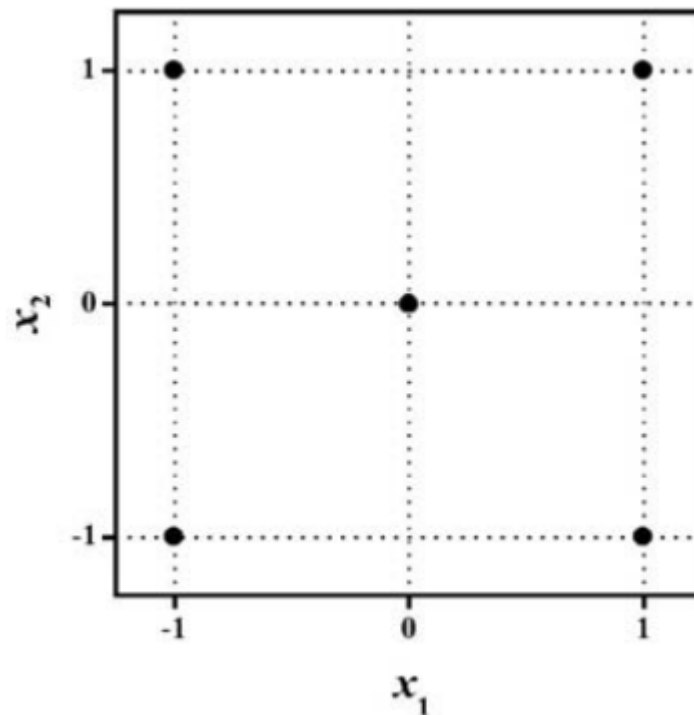


Figure 3.1: Planejamento  $2^2$  com ponto central.

Logicamente não há como fugir das repetições, mas o número destas, na maioria dos casos, é significativamente reduzido. No entanto, esta metodologia é possível de ser aplicada somente quando se utilizam variáveis quantitativas, visto que, para variáveis qualitativas, não há como adicionar níveis no ponto central.

Os cálculos para estimativa de efeitos e coeficientes relacionados aos planejamentos com ponto central serão discutidos no item Modelos empíricos em estudo de triagem.

### 3.2.2 Modelos empíricos em estudo de triagem

Pode-se assumir, desde o início do experimento, que o sistema estudado (domínio experimental) é regido por alguma função que é descrita pelas variáveis experimentais. Normalmente esta função pode ser aproximada por um polinômio, o qual pode fornecer uma boa descrição entre os fatores e a resposta obtida. A ordem deste polinômio é limitada pelo tipo de planejamento usado.

Planejamentos fatoriais de dois níveis, completos ou fracionários, podem estimar apenas efeitos principais e interações.

Planejamentos fatoriais de três níveis (ponto central) podem estimar, além disso, o grau de curvatura na resposta.

Para descrever tais modelos em um estudo de triagem, são utilizados os polinômios mais simples, ou seja, aqueles que contêm apenas termos lineares. Considerando um exemplo para três variáveis,  $x_1$ ,  $x_2$  e  $x_3$ , dois polinômios seriam:

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + e \quad (3.1)$$

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_{12}x_1x_2 + b_{13}x_1x_3 + b_{23}x_2x_3 + b_{123}x_1x_2x_3 + e \quad (3.2)$$

Segundo a Equação 1, o coeficiente  $b_0$  é o valor populacional da média de todas as respostas obtidas,  $b_1$ ,  $b_2$  e  $b_3$  são os coeficientes relacionados com as variáveis  $x_1$ ,  $x_2$  e  $x_3$ , respectivamente, e  $e$  é o erro aleatório associado ao modelo e, para o caso da Equação 2,  $b_{12}$ ,  $b_{13}$ ,  $b_{23}$  são os coeficientes para as interações  $x_1x_2$ ,  $x_1x_3$ ,  $x_2x_3$  e  $b_{123}$  é o coeficiente para a interação  $x_1x_2x_3$ .

A partir do planejamento montado, pode-se construir a matriz de coeficientes de contraste, de acordo com a Tabela 1. A matriz de coeficientes de contraste  $X$ , juntamente com o vetor de respostas  $y$ , obtido experimentalmente, será utilizada para cálculo dos coeficientes do modelo, conhecidos também como vetor de regressão.

O modelo procurado, descrito pelas Equações 1 e 2 pode ser representado matricialmente pela Equação 3

$$\hat{y} = Xb \quad (3.3)$$

em que  $\hat{y}$  é o vetor das respostas estimadas pelo modelo e  $b$  o vetor de regressão.

Uma maneira de determinar o vetor de regressão  $b$  é através do método dos quadrados mínimos, definido pela Equação 4, em que  $X^t$  indica a transposta de  $X$

$$b = (X^tX)^{-1}X^ty \quad (3.4)$$

$(X^tX)^{-1}$  é a matriz inversa do produto da transposta da matriz  $X$  com ela mesma.

Para um planejamento fatorial completo, a matriz  $X$  é quadrada e ortogonal, onde  $n$  corresponde ao número de ensaios. Isto ocorre pois as colunas de  $X$  não estão normalizadas, portanto  $X^{-1} = X^t$ . Neste caso, a Equação 8 se resume à Equação 5

$$b = \left(\frac{1}{n}\right)X^ty \quad (3.5)$$

A Equação 4 é geral e pode ser empregada para se fazer a estimativa de efeitos e



coeficientes para todos os planejamentos descritos neste artigo, no entanto a Equação 5 é específica para o planejamento fatorial completo, não sendo aplicável aos outros planejamentos descritos.

Como as variáveis são estudadas em dois níveis codificados, cada efeito satisfaz à variação de duas unidades da variável correspondente. Se considerarmos os fatores individualmente, pode-se provar que o valor de cada coeficiente do modelo é a metade do valor do efeito correspondente, exceto para  $b_0$ , cujo valor é o mesmo do calculado para seu efeito. Desta maneira, o modelo empregado para descrever as respostas é elaborado em função dos efeitos por unidade de variação individual.

### 3.2.3 Estimativa dos erros para os efeitos através de repetições

Normalmente os resultados obtidos em experimentos de planejamento fatorial completo ou fracionário com repetições consistem de uma pequena amostra de um hipotético conjunto maior, representado por uma população. Destes dados podemos obter a média amostral, a variância amostral e o desvio padrão amostral, como se segue:

$$\bar{y} = \sum_{i=1}^r y_i \quad (3.6)$$

$$s^2 = \frac{\sum_{i=1}^r (y_i - \bar{y})^2}{(r - 1)} \quad (3.7)$$

$$s = \pm \sqrt{\frac{\sum_{i=1}^r (y_i - \bar{y})^2}{(r - 1)}} \quad (3.8)$$

em que  $r$  é o número de replicatas, isto é, o número de ensaios realizados em um mesmo ponto experimental (nível),  $y_i$  são os valores de cada observação individual,  $\bar{y}$  é o valor médio,  $s^2$  é a variância e  $s$  o desvio padrão.

A soma dos desvios da média amostral de  $r$  replicatas é necessariamente zero. Isto requer que  $\sum (y_i - \bar{y}) = 0$  constitua uma restrição linear nos desvios usados no cálculo de  $s^2$ . Está subentendido que com  $r-1$  replicatas é possível determinar a que falta. Os  $r$  resíduos  $y_i - \bar{y}$  e, conseqüentemente, a soma dos seus quadrados juntamente com a variância amostral são ditas ter, por esta razão,  $r-1$  graus de liberdade. A perda de um grau de liberdade está associada à necessidade de substituir a média populacional pela média amostral derivada dos dados. Desta maneira, quando repetições genuínas são realizadas em uma série de condições experimentais, a variação entre suas observações pode ser usada para estimar o desvio padrão de uma simples observação e, conseqüentemente, o desvio dos efeitos. Todos os ensaios, inclusive repetições, devem ser realizados aleatoriamente, refazendo todas as etapas do experimento. As repetições realizadas desta maneira são

consideradas genuínas.

Em um experimento em que cada ensaio foi realizado  $r$  vezes, se o valor der for pequeno, por exemplo,  $2 \leq r \leq 10$ , haverá poucos graus de liberdade para o cálculo da variância. Uma maneira para se obter um maior número de graus de liberdade é realizando uma estimativa conjunta das variâncias, conforme a Equação 9

$$s^2 = \frac{\sum_{i=1}^n (r_i - 1) s_i^2}{\sum_{i=1}^n (r_i - 1)} \quad (3.9)$$

$$\nu = \sum_{i=1}^n (r_i - 1) \quad (3.10)$$

O resultado da Equação 9 reflete a variância conjunta ( $s_c^2$ ) de cada observação individual  $y_i$  sendo o erro padrão igual à raiz quadrada do mesmo. A Equação 10, que é exatamente o denominador da Equação 9 é, portanto, o número de graus de liberdade da estimativa conjunta, designada como  $\nu$ .

Considerando a realização de repetições autênticas, que cada efeito é uma combinação linear dos valores de  $y_i$  dos  $n$  ensaios, com coeficientes  $c$  iguais a  $+2/n$  e  $-2/n$ . Levando em consideração a aleatoriedade dos ensaios, tais valores são estatisticamente independentes e apresentam a mesma variância populacional  $\sigma^2$ . Neste caso, por definição, as correlações entre todas as variáveis são nulas, e a variância da combinação linear das variáveis aleatórias pode ser dada por.

$$\sigma^2 = \sum_i c_i^2 \sigma_i^2 \quad (3.11)$$

Transportando a Equação 11 para o nosso mundo amostral, pode-se provar que a variância de cada efeito é dada por

$$V_{(ef)} = \frac{4s^2}{\sum_{i=1}^n r_i} \quad (3.12)$$

em que  $s_c^2$  é dada pela Equação 9 e  $r_i$  é o número de replicatas em cada ensaio  $n$ .

E, finalmente, para calcular o valor do erro de cada efeito, basta extrair a raiz quadrada de  $V_{(ef)}$ .

### 3.2.4 Estimativa dos erros para os efeitos, sem repetições

Uma maneira de estimar erros de efeitos sem a realização de repetições é supor que interações de altas ordens para  $k \geq 3$  não são significativos e, portanto, são erros experimentais nos valores dos efeitos.

Aplicando a Equação 9 sobre estes efeitos de interação e fazendo algumas considerações, obtém-se a variância dos efeitos conforme a 3.13

$$V_{(ef)} = \frac{\sum (ef_{ii})^2}{l} \quad (3.13)$$

em que  $ef_{ii}$  são os efeitos de interação considerados como erros experimentais e  $l$  é o número total de efeitos considerados.

É preciso estar atento ao utilizar este tipo de estimativa do erro. Nem sempre os efeitos de altas ordens são irrelevantes e, se tais efeitos forem incluídos no cálculo, os erros tornam-se altos e, desta maneira, não é possível distinguir com confiabilidade aqueles que são realmente importantes.

O número de graus de liberdade utilizado para avaliação dos efeitos agregados a estes erros é o mesmo número total de efeitos considerados como erros, isto é, o valor  $l$  da Equação 13.

### 3.2.5 Estimativa dos erros para os efeitos e coeficientes a partir das repetições no ponto central

Conforme mencionado anteriormente, uma das grandes vantagens da inclusão de experimentos no centro do planejamento é devida à estimativa do erro com poucas repetições, normalmente entre 3 e 5.

A partir das repetições realizadas é possível obter a média e a variância das replicatas de acordo com as Equações 10 e 11, respectivamente.

Por outro lado, sendo os coeficientes obtidos através da Equação 8, nota-se que a matriz  $(X^t X)^{-1}$  apresenta grande influência na variância dos parâmetros de regressão. O produto desta matriz com o valor da variância obtida através das repetições no ponto central (Equação 11) fornece a matriz  $V(b)$  conhecida como matriz de variância-covariância (Equação 14). A matriz  $V(b)$  é simétrica e os elementos de sua diagonal são as variâncias dos parâmetros de regressão, na mesma ordem em que elas aparecem na equação de regressão. Os elementos fora da diagonal são as covariâncias entre os parâmetros de regressão. A raiz quadrada dos elementos da diagonal principal da matriz  $V(b)$  determina os valores correspondentes dos erros padrão ( $s_{bi}$ ) dos coeficientes calculados (Equação 15).

$$V(b) = (X^t X)^{-1} s^2 \quad (3.14)$$

$$s_{bi} = \pm \sqrt{v(b)_{ij,i}} = 1, 2, \dots, n \quad (3.15)$$

### 3.2.6 Identificação dos efeitos e coeficientes significativos

Na literatura há diferentes métodos para se avaliar efeitos e coeficientes significativos; entre os mais usados destaca-se a análise de variância (ANOVA), o gráfico de probabilidade normal (distribuição normal) e a comparação de efeitos com uma medida independente da variabilidade. Neste trabalho, as avaliações de significância para a decisão estatística, tanto para efeitos como para coeficientes de modelos, foram realizadas empregando o teste  $t$  (distribuição de Student), através do valor  $p$ .

Para realizar a interpretação correta utilizando o valor  $p$  é necessário compreender os testes de hipótese e significância, conforme descrito abaixo.

Em muitos casos, formula-se uma hipótese estatística com o propósito de rejeitá-la ou invalidá-la. Por ex., se o desejo é decidir se um sistema é diferente de outro, formula-se a hipótese de que não há diferença entre os sistemas. Essa hipótese é denominada nula e representada por  $H_0$ . Qualquer hipótese diferente da pré-fixada é denominada hipótese alternativa e é normalmente representada por  $H_1$ .

Se uma hipótese for rejeitada quando deveria ser aceita, diz-se que foi cometido um erro tipo I. Se, por outro lado, uma hipótese for aceita quando deveria ter sido rejeitada, diz-se que foi cometido um erro do tipo II. Em ambos os casos foi tomada uma decisão errada ou foi cometido um erro de julgamento.

Para que qualquer teste de hipótese ou regra de decisão seja adequada, eles devem ser planejados de modo que os erros de decisão sejam reduzidos ao mínimo.

Ao testar uma hipótese estabelecida, a probabilidade máxima, representada frequentemente por  $\alpha$ , com a qual se sujeitaria a correr o risco de um erro do tipo I é denominada nível de significância do teste.

Se, por ex., é escolhido um nível de significância 0,05 ou 5%, há então cerca de 5 chances em 100 da hipótese nula ser rejeitada, quando deveria ser aceita, isto é, há uma confiança de 95% de que se tomou uma decisão correta. Nesses casos, diz-se que a hipótese é rejeitada ao nível de significância 0,05, o que significa que a probabilidade de erro seria de 0,05.

Considerando a hipótese nula de que o valor do efeito se confunde com seu erro, pode-se formular a seguinte regra de decisão:

- a) rejeição da hipótese nula com 5% de significância, quando o valor de  $t_{calc}$  situar-se fora do intervalo entre  $\pm t\alpha$  ( $|t_{calc}| \geq t\alpha$ ). Isso equivale a dizer que o valor estatístico amostral observado é significativo no nível definido e,
- b) aceitação da hipótese, caso contrário.

Uma maneira alternativa de concluir o teste de hipótese é comparar o valor  $p$  do teste estatístico amostral com o nível de significância  $\alpha$ . O valor  $p$  do teste estatístico amostral é o menor nível de significância para que se rejeite  $H_0$ . Neste sentido, compara-se o valor  $p$  com  $\alpha$  e, se o valor  $p \leq \alpha$  rejeita-se  $H_0$ , caso contrário  $H_0$  é aceito. A vantagem de se conhecer o valor  $p$  está na possibilidade de se avaliar todos os níveis de significância para

que o resultado observado possa ser estatisticamente rejeitado. A representação gráfica do valor  $p$  é descrita na Figura 3.

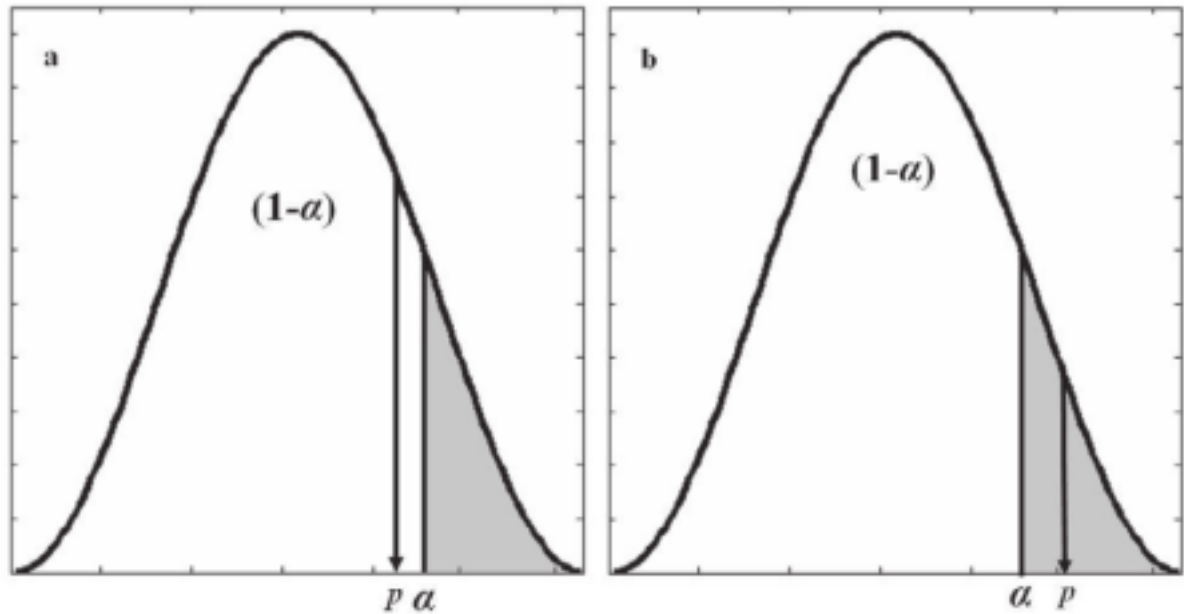


Figure 3.2: Representação gráfica do valor  $p$  para um teste unilateral.

Conforme as Figuras 3.2(a e b), o nível de significância  $\alpha$  é a área hachurada no gráfico de distribuição. Na Figura 3.2a a área correspondente ao valor  $p$  é maior que o nível de significância, portanto, o valor calculado do teste estatístico está fora da região crítica, o que implica em aceitar  $H_0$ . Para a Figura 3b a área do valor  $p$  é menor que o nível de significância e assim, o valor calculado do teste estatístico está dentro da região crítica, o que implica em rejeitar  $H_0$ . É importante ressaltar que esta discussão é válida tanto para testes unilaterais quanto para bilaterais.

Especificamente, o valor  $p$  representa a probabilidade de validade do erro envolvido no resultado observado, isto é, como representativo da população. Por ex., levando em consideração o valor de um efeito, se o valor do teste estatístico calculado (razão entre o efeito e seu erro) apresentar um grande desvio da distribuição de Student, ele provavelmente descreve algo mais que o resíduo experimental. Desta maneira, será significativo dentro de um intervalo de confiança e assim,  $|t_{calc}| \geq t_\alpha$  ou  $p \leq \alpha$ , para o número de graus de liberdade em questão.

### 3.2.7 Avaliação do modelo

O modelo obtido pode não ser exatamente aquele que descreve a região estudada do sistema e, neste caso, não pode ser usado para fazer estimativas para deslocamento e muito menos para extrair conclusões sobre a região ótima. A maneira mais confiável de se avaliar a qualidade do ajuste do modelo é empregando a análise de variância (ANOVA).

Na ANOVA a variação total da resposta é definida como a soma quadrática de dois

componentes: a soma quadrática da regressão ( $SQ_{regr}$ ) e a soma quadrática dos resíduos ( $SQ_{res}$ ). A soma quadrática da variação total, corrigida para a média ( $SQ_{total}$ ) pode, assim, ser escrita como a Equação 16

$$SQ_{total} = SQ_{regr} + SQ_{res} \quad (3.16)$$

em que  $SQ_{regr}$  e  $SQ_{res}$  são apresentadas com mais detalhes nas Equações 17 e 18, respectivamente.

$$SQ_{regr} = \sum_{i=1}^m \sum_{j=1}^{r_i} (\hat{y}_i - \bar{y})^2 \quad (3.17)$$

$$SQ_{res} = \sum_{i=1}^m \sum_{j=1}^{r_i} (y_{ij} - \bar{y})^2 \quad (3.18)$$

Da Equação 24,  $m$  é o número total de níveis do planejamento, isto é, pontos experimentais do planejamento  $\hat{y}_i$  é o valor estimado pelo modelo para o nível  $i$  e  $\bar{y}_i$  é o valor médio das replicatas ( $r$ ). No entanto, como há somente replicatas no ponto central, a média das replicatas nos níveis (+) mais e (-) menos é o próprio valor observado do ensaio naquele nível. Note que o segundo somatório indica que se deve fazer o quadrado das diferenças inclusive com cada repetição.

As replicatas realizadas no ponto central deixarão um resíduo para cada observação  $y_i$ . A soma quadrática destes resíduos fornece a soma quadrática residual somente no nível zero.

A Equação 25 indica que o quadrado da diferença entre o valor de cada observação ( $y_i$ ) e o valor estimado ( $\hat{y}_i$ ) e, inclusive das replicatas ( $y_j$ ) em cada nível ( $m$ ), fornece a soma quadrática residual de todos os níveis.

Quando algum modelo é ajustado aos dados, a soma quadrática do erro puro é sempre uma parte da soma quadrática dos resíduos. Então, cada resíduo pode ser desmembrado em duas partes, isto é

$$(y_{ij} - \hat{y}_i) = (y_{ij} - \bar{y}) - (\hat{y}_i - \bar{y}_i) \quad (3.19)$$

Pela Equação 19, o primeiro termo da direita diz respeito à diferença entre o valor de cada observação individual no nível e a média de todas as observações naquele nível. Já o segundo termo corresponde à diferença entre o valor estimado no nível e a média de todas as observações naquele nível. A subtração entre estes dois termos fornece como resposta o resíduo de cada observação individual. Para evitar os termos negativos na Equação 19, tomam-se as suas diferenças quadráticas e obtém-se Equação 20

$$\sum_{i=1}^m \sum_{j=1}^{r_i} (y_{ij} - \hat{y}_i)^2 = \sum_{i=1}^m \sum_{j=1}^{r_i} (y_{ij} - \bar{y})^2 + \sum_{i=1}^m (\hat{y}_i - \bar{y})^2 \quad (3.20)$$

O primeiro termo da direita é chamado de soma quadrática do erro puro e está relacionado exclusivamente com os erros aleatórios das replicatas. Já o segundo termo da direita é chamado de soma quadrática da falta de ajuste, pois ele é uma medida da discrepância entre a resposta do modelo de previsão ( $\hat{y}_i$ ) e a média das replicatas ( $\bar{y}_i$ ) realizadas no conjunto de condições experimentais.

A Equação 20 pode ser resumida da seguinte maneira

$$SQ_{res} = SQ_{ep} + SQ_{faj} \quad (3.21)$$

Para cada fonte de variação (regressão, resíduos, falta de ajuste, erro puro e total), é necessário obter o número de graus de liberdade. Não introduzindo detalhes, pode-se provar que o número de graus de liberdade para as Equações 17, 18 e 20 são  $p-1$ ,  $n-p$  e  $(n-m) + (m-p)$ , respectivamente, em que  $p$  é o número de parâmetros (coeficientes) do modelo,  $n$  é o número total de observações (ensaios) e  $m$  é o número de níveis do planejamento. O número de graus de liberdade para as outras fontes de variação pode ser obtido por cálculos algébricos simples.

A divisão da soma quadrática de cada fonte de variação pelo seu respectivo número de grau de liberdade fornece a média quadrática ( $MQ$ ). A razão entre a média quadrática da regressão ( $MQ_{reg}$ ) pela média quadrática dos resíduos ( $MQ_{res}$ ), que nada mais é do que a razão entre duas variâncias, pode ser usada para comparar tais fontes de variação através do teste F (distribuição de Fisher), levando em consideração seus respectivos números de graus de liberdade. O mesmo pode ser feito para a razão entre a média quadrática da falta de ajuste ( $MQ_{faj}$ ) pela média quadrática do erro puro ( $MQ_{ep}$ ).

Assim, como foi realizada a avaliação dos efeitos e coeficientes empregando o teste t, através do valor  $p$ , o mesmo será feito para comparar as duas fontes de variação entre si, empregando-se neste caso o teste F (Equação 22)

$$\frac{MQ_1}{MQ_2} = F_{v_1 v_2} \quad (3.22)$$

em que  $MQ_1$  e  $MQ_2$  são as médias quadráticas das fontes de variação 1 e 2, respectivamente, e  $\nu_1, \nu_2$  são seus respectivos números de graus de liberdade.

Exemplificando, a razão entre as médias quadráticas da falta de ajuste e do erro puro é o valor calculado do teste estatístico ( $F_{calc}$ ), que é usado para comparar qual é mais significativo. Pode-se formular uma hipótese nula ( $H_0$ ) considerando que não há diferença entre as fontes de variação comparadas. Esta hipótese se reflete na seguinte regra de decisão:

a) rejeição da hipótese nula com 5% de significância, quando o valor de  $F_{calc}$  se situar fora do intervalo definido por  $F\alpha$  ( $|F_{calc}| \geq F\alpha$ ), ou seja,  $p \leq \alpha$ . Isso equivale a dizer que o valor estatístico amostral observado é significativo no nível definido e as fontes de variação comparadas são diferentes.

b) Aceitação da hipótese nula, caso contrário. Se  $H_0$  for rejeitada para  $MQ_{faj}/MQ_{ep}$ ,

isto é,  $p \leq \alpha$ , então há uma falta de ajuste significativa ao nível de probabilidade e número de graus de liberdade definidos (normalmente,  $\alpha = 0,05$ ) e o modelo não é adequado.

Em termos práticos, um bom modelo necessita ter uma regressão significativa e uma falta de ajuste não significativa. Isto equivale a dizer que a maior parte da variação total das observações em torno da média deve ser descrita pela equação de regressão e o restante certamente, ficará com os resíduos. Da variação que fica com os resíduos é esperado que a maior parte fique com o erro puro, que é o erro experimental, e não com a falta de ajuste, que está diretamente relacionada com o modelo.

Outros parâmetros para observar se toda variação em torno da média foi explicada pela regressão são o valor do coeficiente de variação  $R^2$  (Equação 23) e o gráfico dos resíduos. O valor de  $R^2$  representa a fração da variação que é explicada pela falta de ajuste do modelo (Equação 23). Quanto mais próximo de 1 o valor do coeficiente estiver, melhor estará o ajuste do modelo às respostas observadas.

$$R^2 = \frac{SQ_{res}}{SQ_{total}} \quad (3.23)$$

Note que o erro puro não explica nada do modelo, então o valor máximo possível de  $R^2$  é

$$R^2 = \frac{SQ_{total} - SQ_{ep}}{SQ_{total}} \quad (3.24)$$

A análise de variância (ANOVA), de acordo com as Equações descritas para avaliação do modelo, está resumida na tabela abaixo.

Fonte de variação	SQ	N° de graus de liberdade	MQ	F
Regressão	$SQ_{regr}$	$p - 1$	$MQ_{regr}$	$MQ_{regr}/MQ_{res}$
Resíduos	$SQ_{res}$	$n - p$	$MQ_{res}$	
Falta de ajuste	$SQ_{faj}$	$m - p$	$MQ_{faj}$	$MQ_{faj}/MQ_{ep}$
Erro puro	$SQ_{ep}$	$n - m$	$MQ_{ep}$	
Total	$SQ_{total}$	$n - 1$		

Table 3.1: Análise de variância para regressão múltipla

Uma boa prática é examinar a distribuição dos resíduos, pois ajuda a verificar se não há nada de errado com o modelo. Neste caso, pode-se fazer um gráfico dos valores estimados pelo modelo ( $\hat{y}_i$ ) com os valores da diferença entre os valores observados experimentalmente ( $y_i$ ) e seus respectivos valores estimados, isto é,  $(y_i - \hat{y}_i)$ . Se os resíduos não estiverem aleatoriamente distribuídos, pode-se desconfiar do modelo e investir em outros recursos para sua melhoria.



### 3.3 Diagrama de pacotes – assuntos

Nesta seção, apresentamos o diagrama de pacotes na Figura 3.4. Percebemos a grande interação entre esses componentes, reafirmando o que foi sicutido na anteriormente. Neste diagrama, temos os seguintes componentes, representados em UML 2.0..

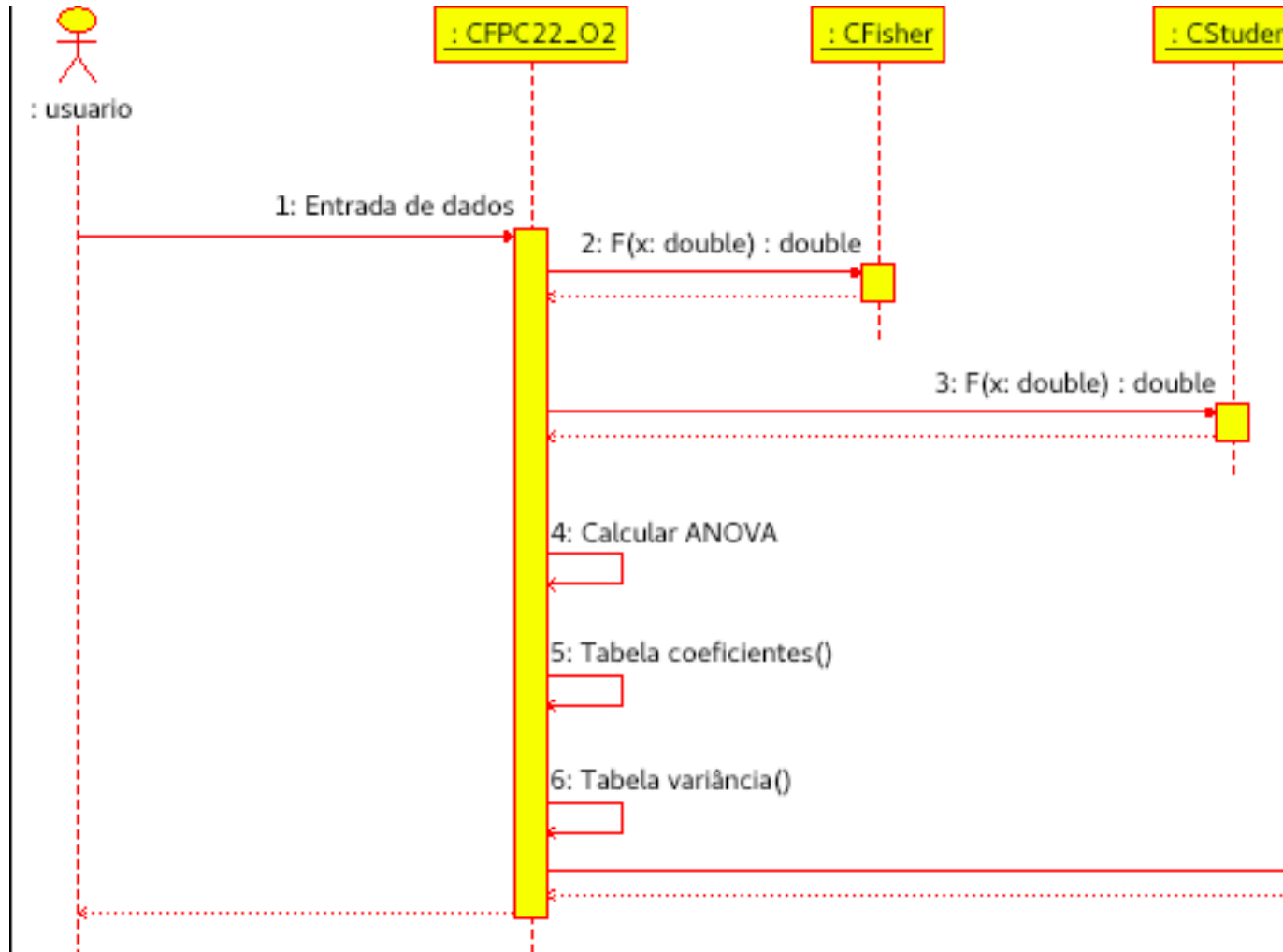


Figure 3.3: Diagrama de Pacotes

# Chapter 4

## AOO – Análise Orientada a Objeto

A terceira etapa do desenvolvimento de um projeto de engenharia, no nosso caso um software aplicado a engenharia de petróleo, é a AOO – Análise Orientada a Objeto. A AOO utiliza algumas regras para identificar os objetos de interesse, as relações entre os pacotes, as classes, os atributos, os métodos, as heranças, as associações, as agregações, as composições e as dependências.

O modelo de análise deve ser conciso, simplificado e deve mostrar o que deve ser feito, não se preocupando como isso será realizado.

O resultado da análise é um conjunto de diagramas que identificam os objetos e seus relacionamentos.

### 4.1 Diagramas de classes

O diagrama de classes é apresentado na Figura 4.1.

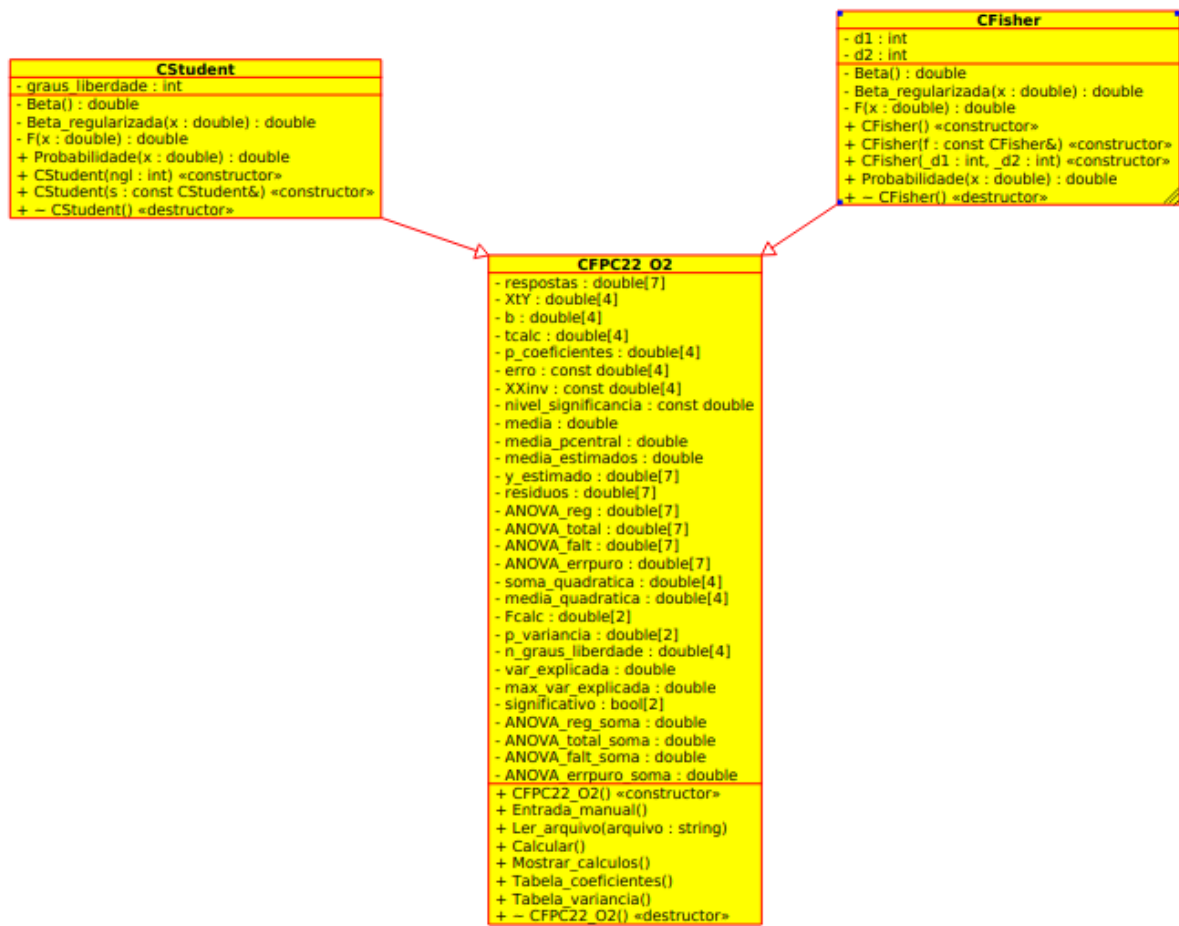


Figure 4.1: Diagrama de classes

### 4.1.1 Dicionário de classes

- Classe CFisher: representa o calculo do teste F fisher para a avaliação do modelo de entre 2 amostra.
- Classe CStudent: representa o calculo do teste t studend para avaliação para multiplas amostras..
- Classe CFPC22\_02: representa os calculos dos efeitos das variaveis em analises gerando a ANOVA.

## 4.2 Diagrama de seqüência – eventos e mensagens

O diagrama de seqüência enfatiza a troca de eventos e mensagens e sua ordem temporal. Contém informações sobre o fluxo de controle do software. Costuma ser montado a partir de um diagrama de caso de uso e estabelece o relacionamento dos atores (usuários e sistemas externos) com alguns objetos do sistema.

### 4.2.1 Diagrama de sequência geral

Veja o diagrama de sequência na Figura 4.2.

- O diagrama de sequência aqui representado, demonstra como o software faz para plotar os gráficos. Primeiramente o usuário insere os dados pedidos pelo simulador que por sua vez chama a classe CFPC22\_02 para calcular as variâncias. Em seguida o software calcula todos os parâmetros necessários para criar ANOVA. Finalizamos as tabelas com a apresentação dos gráficos para serem analisados

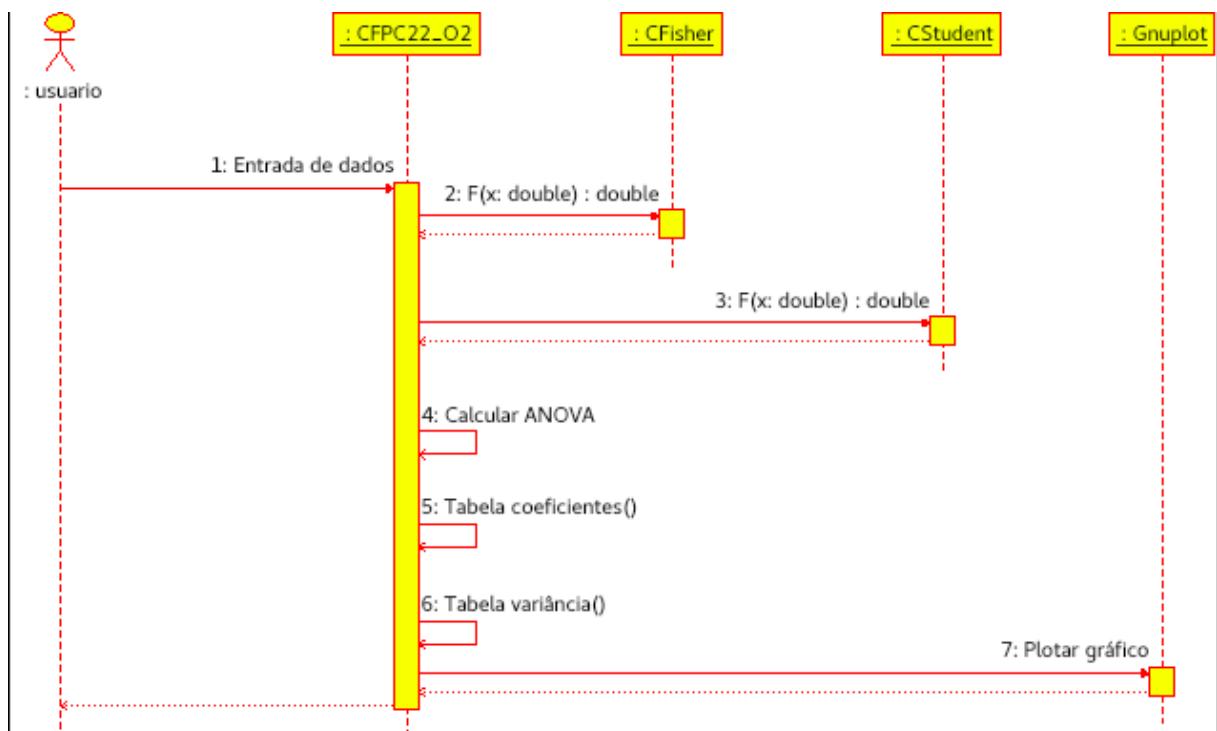


Figure 4.2: Diagrama de sequência

### 4.3 Diagrama de comunicação – colaboração

No diagrama de comunicação o foco é a interação e a troca de mensagens e dados entre os objetos. Veja na Figura 4.3 o diagrama de comunicação mostrando a sequência que um objeto da classe CFPC22\_02 para construir a ANOVA.

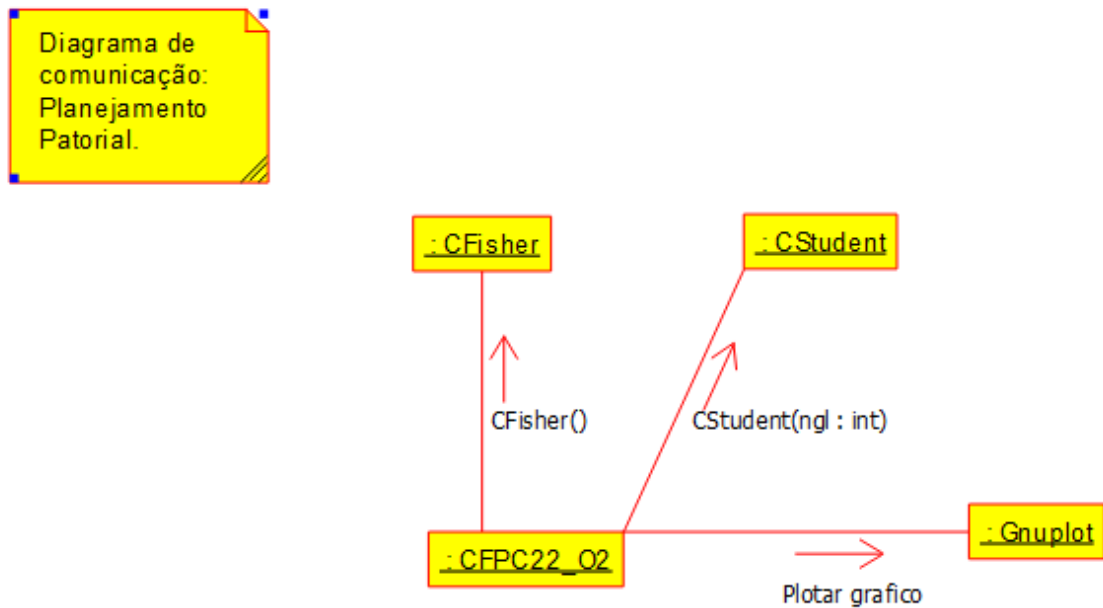


Figure 4.3: Diagrama de comunicação.

## 4.4 Diagrama de máquina de estado

Um diagrama de máquina de estado representa os diversos estados que o objeto assume e os eventos que ocorrem ao longo de sua vida ou mesmo ao longo de um processo (histórico do objeto). É usado para modelar aspectos dinâmicos do objeto.

A Figura 4.5 demonstra os estados que a Classe `CFPC22_02` apresenta durante a execução do software.

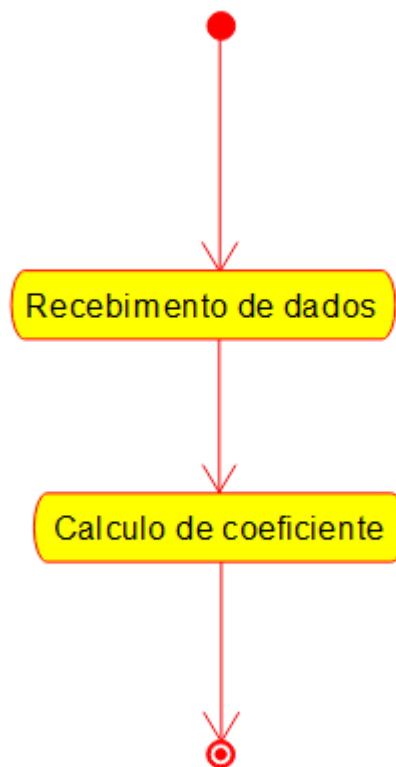


Figure 4.4: Diagrama de máquina de estado para a classe CFPC22\_02.

## 4.5 Diagrama de atividades

Veja na Figura 4.6 o diagrama de atividades corresponde a uma atividade específica da Classe CFPC22\_02, onde calcula-se o erro a partir dos coeficientes.

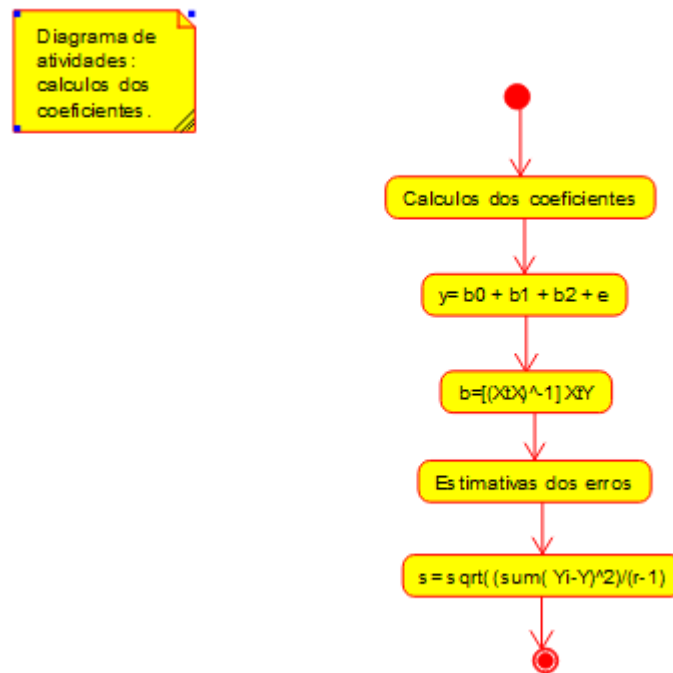


Figure 4.5: Diagrama de atividades do metodo minimos quadrados da classe CFP22\_02.

.

# Chapter 5

## Projeto

Neste capítulo do projeto de engenharia veremos questões associadas ao projeto do sistema, incluindo protocolos, recursos, plataformas suportadas, implicações nos diagramas feitos anteriormente, diagramas de componentes e implantação. Na segunda parte revisamos os diagramas levando em conta as decisões do projeto do sistema.

### 5.1 Projeto do sistema

Depois da análise orientada a objeto desenvolve-se o projeto do sistema, qual envolve etapas como a definição dos protocolos, da interface API, o uso de recursos, a subdivisão do sistema em subsistemas, a alocação dos subsistemas ao hardware e a seleção das estruturas de controle, a seleção das plataformas do sistema, das bibliotecas externas, dos padrões de projeto, além da tomada de decisões conceituais e políticas que formam a infraestrutura do projeto.

Deve-se definir padrões de documentação, padrões para o nome das classes, padrões de retorno e de parâmetros em métodos, características da interface do usuário e características de desempenho.

#### 1. Protocolos

- A única intercomunicação será entre o software desenvolvido e o software Gnuplot, que plotará os gráficos desejados pelo usuário;
- O software receberá dados via teclado;
- A interface utilizada será em modo texto;
- O software terá como saída arquivos de extensão .dat e gráficos em arquivos de extensão .png.

#### 2. Recursos

- O presente programa precisará utilizar o HD, o processador, o teclado, a tela, o mouse, a memória e demais componentes internos do computador.



### 3. Controle

- Não haverá necessidade de grande espaço na memória visto que o programa e seus componentes trabalham com dados relativamente pequenos.
- Neste projeto todos os cálculos necessitam de estruturas de repetição.
- Neste projeto não há necessidade de uso de processos de processamento paralelo, pois os cálculos realizados requerem pouco esforço de processamento

### 4. Plataformas

- O software irá operar nos sistemas operacionais Windows e GNU/Linux, sendo desenvolvido e testado em ambos os sistemas.
- Não haverá necessidade de grandes mudanças para tornar o programa multiplataforma pois a linguagem escolhida, C++, tem suporte em todos estes sistemas operacionais, [Bueno, 2003].
- Para a geração de gráficos será utilizado o software livre Gnuplot.
- Os ambientes de desenvolvimento serão o CodeBlocks (Windows) e Kate (Linux).

## 5.2 Projeto orientado a objeto – POO

O projeto orientado a objeto é a etapa posterior ao projeto do sistema. Baseia-se na análise, mas considera as decisões do projeto do sistema. Acrescenta a análise desenvolvida e as características da plataforma escolhida (hardware, sistema operacional e linguagem de softwareção). Passa pelo maior detalhamento do funcionamento do software, acrescentando atributos e métodos que envolvem a solução de problemas específicos não identificados durante a análise.

Envolve a otimização da estrutura de dados e dos algoritmos, a minimização do tempo de execução, de memória e de custos. Existe um desvio de ênfase para os conceitos da plataforma selecionada.

### Efeitos do projeto no modelo estrutural

- Adicionar nos diagramas de pacotes as bibliotecas e subsistemas selecionados no projeto do sistema (exemplo: a biblioteca gráfica selecionada).
  - Neste projeto foi utilizada como biblioteca gráfico CGnuplot.
- Novas classes e associações oriundas das bibliotecas selecionadas e da linguagem escolhida devem ser acrescentadas ao modelo.
- Estabelecer as dependências e restrições associadas à plataforma escolhida.
  - Neste projeto não se aplica.

**Efeitos do projeto no modelo dinâmico**

- Revisar os diagramas de seqüência e de comunicação considerando a plataforma escolhida.
- Verificar a necessidade de se revisar, ampliar e adicionar novos diagramas de máquinas de estado e de atividades.

**Efeitos do projeto nos atributos**

- Atributos novos podem ser adicionados a uma classe, como, por exemplo, atributos específicos de uma determinada linguagem de softwareção (acesso a disco, ponteiros, constantes e informações correlacionadas).

**Efeitos do projeto nos métodos**

- Em função da plataforma escolhida, verifique as possíveis alterações nos métodos. O projeto do sistema costuma afetar os métodos de acesso aos diversos dispositivos (exemplo: hd, rede).
- De maneira geral os métodos devem ser divididos em dois tipos: i) tomada de decisões, métodos políticos ou de controle; devem ser claros, legíveis, flexíveis e usam polimorfismo. ii) realização de processamentos, podem ser otimizados e em alguns casos o polimorfismo deve ser evitado.

**Efeitos do projeto nas heranças**

- Reorganização das classes e dos métodos (criar métodos genéricos com parâmetros que nem sempre são necessários e englobam métodos existentes).
- Abstração do comportamento comum (duas classes podem ter uma superclasse em comum).
- Utilização de delegação para compartilhar a implementação (quando você cria uma herança irreal para reaproveitar código). Usar com cuidado.
- Revise as heranças no diagrama de classes.

## 5.3 Diagrama de componentes

O diagrama de componentes mostra a forma como os componentes do software se relacionam, suas dependências. Inclui itens como: componentes, subsistemas, executáveis, nós, associações, dependências, generalizações, restrições e notas.

Veja na Figura 5.1 um exemplo de diagrama de componentes. Observe que este inclui dependências, ilustrando as relações entre os arquivos. O software executável a ser gerado depende da biblioteca gerada e do módulo de arquivos

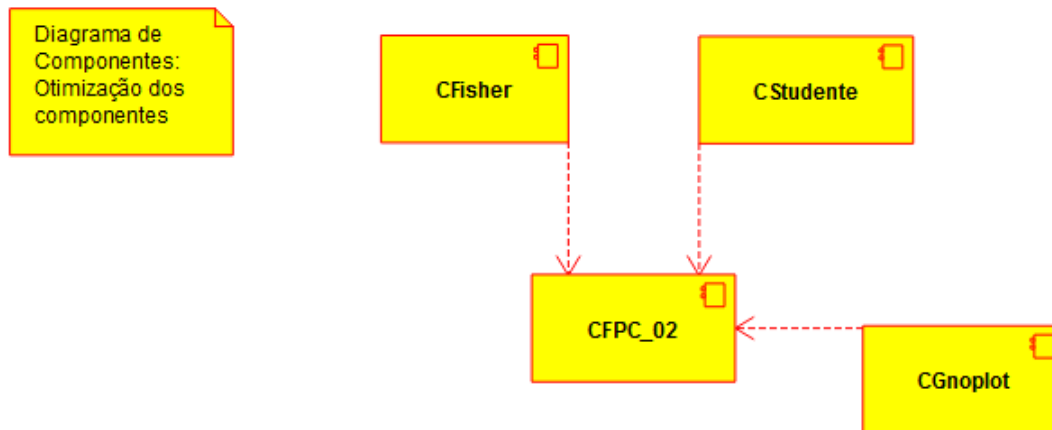


Figure 5.1: Diagrama de componentes.

## 5.4 Diagrama de implantação

O diagrama de implantação é um diagrama de alto nível que inclui relações entre o sistema e o hardware e que se preocupa com os aspectos da arquitetura computacional escolhida. Seu enfoque é o hardware, a configuração dos nós em tempo de execução.

O diagrama de implantação deve incluir os elementos necessários para que o sistema seja colocado em funcionamento: computador, periféricos, processadores, dispositivos, nós, relacionamentos de dependência, associação, componentes, subsistemas, restrições e notas.

Veja na Figura 5.2 um exemplo de diagrama de implantação de uma coluna cromatografica. Observe a filtragem dos componentes SARA. Os resultados são armazenados em um servidor de arquivos.

O procedimento foi realizado no laboratório de Geoquímica no campus LENEP em Macaé.

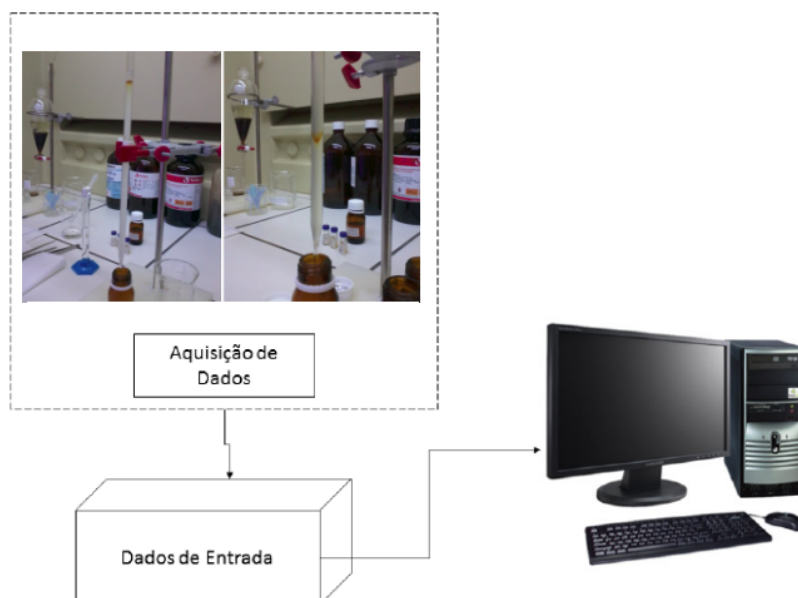


Figure 5.2: Diagrama de implantação

# Chapter 6

## Implementação

Neste capítulo do projeto de engenharia apresentamos os códigos fonte que foram desenvolvidos.

**Nota:** os códigos devem ser documentados usando padrão **javadoc**. Posteriormente usar o programa **doxygen** para gerar a documentação no formato html.

- Veja informações gerais aqui <http://www.doxygen.org/>.
- Veja exemplo aqui <http://www.stack.nl/~dimitri/doxygen/manual/docblocks.html>.

**Nota:** ao longo deste capítulo usamos inclusão direta de arquivos externos usando o pacote *listings* do L<sup>A</sup>T<sub>E</sub>X. Maiores detalhes de como a saída pode ser gerada estão disponíveis nos links abaixo.

- [http://en.wikibooks.org/wiki/LaTeX/Source\\_Code\\_Listings](http://en.wikibooks.org/wiki/LaTeX/Source_Code_Listings).
- <http://mirrors.ctan.org/macros/latex/contrib/listings/listings.pdf>.

### 6.1 Código fonte

Apresenta-se a seguir um conjunto de classes (arquivos .h e .cpp) além do programa `main`.

Apresenta-se na listagem o arquivo com código da classe `CStudent`.

Listing 6.1: Arquivo de cabeçalho da classe `CStudent`.

```
1
2 #ifndef CStudent_hpp
3 #define CStudent_hpp
4
5     /// classe CStudent
6 class CStudent
7 {
8
```

```

9  private:
10     /// tamanho da amostra(n) menos 1, n > 1
11     int graus_liberdade;
12     /// Beta
13     double Beta();
14     /// Beta Beta_regularizada
15     double Beta_regularizada(double x);
16     /// Função gaussiana
17     double F(double x);
18
19  public:
20     /// Calculo da Probabilidade
21     double Probabilidade(double x);
22     ///costrutor default inicializando com ngl.
23     CStudent(int ngl);
24     /// Construtor de cópia .
25     CStudent ( const CStudent & s ): graus_liberdade (s.graus_liberdade
26         ) { }
27     /// Destrutor default.
28     ~CStudent();
29 # endif

```

Apresenta-se na listagem o arquivo de implementação da classe CStudent.

Listing 6.2: Arquivo de implementação da classe CStudent.

```

30 #include "CStudent.hpp"
31 #include <cmath>
32 #include <iostream>
33
34 const double PI = 3.1416;
35 using namespace std;
36 /// Costrutur default.
37 CStudent::CStudent(int ngl)
38 {
39     graus_liberdade = ngl;
40 }
41 /// Metodo calcular Probabilidade
42 double CStudent::Probabilidade(double x)
43 {
44     double argumento = (graus_liberdade / (pow(x, 2) + graus_liberdade))
45         ;
46     return Beta_regularizada(argumento);
47 }
48 /// Metodo calcular Beta
49 double CStudent::Beta()
50 {
51     int n = 1000;
52     double largura = 1.0 / n;

```

```

52     double integral = 0.0;
53     double centro = largura / 2.0;
54     for (int i = 1; i < n; i++)
55     {
56         integral += largura * F(centro);
57         centro += largura;
58     }
59     return integral;
60 }
61     /// Metodo calcular Beta_regularizada.
62 double CStudent::Beta_regularizada(double x)
63 {
64     int n = 1000;
65     double largura = x / n;
66     double integral = 0.0;
67     double centro = largura / 2.0;
68     for (int i = 1; i < n; i++)
69     {
70         integral += largura * F(centro);
71         centro += largura;
72     }
73     return integral / Beta();
74 }
75     /// Metodo calcular F
76 double CStudent::F(double x)
77 {
78     double a = graus_liberdade / 2.0;
79     double b = 0.5;
80     return (pow(x, a - 1.0) * pow(1 - x, b - 1.0));
81 }
82     /// Destrutor
83 CStudent::~CStudent()
84 {
85 }

```

Apresenta-se na listagem o arquivo com código da classe CFisher.

Listing 6.3: Arquivo de cabeçalho da classe CFisher.

```

86
87 #ifndef CFisher_hpp
88 #define CFisher_hpp
89
90 class CFisher
91 {
92     private:
93
94     int d1; /// variavel 1
95     int d2; /// variavel 2
96     /// Calculo de beta

```

```

97     double Beta();
98     /// calculo de beta regularizado
99     double Beta_regularizada(double x);
100    /// Calculo da gaussiana
101    double F(double x);
102
103    public:
104    /// Construtor default .
105    CFisher( ) { }
106    /// Construtor de cópia .
107    CFisher ( const CFisher & f ): d1 (f. d1 ), d2 (f. d2 ) { }
108    /// Costrutor sobrecarregado .
109    CFisher(int _d1, int _d2);
110    /// Calculo da probabilidade
111    double Probabilidade(double x);
112    /// Destrutor deufault .
113    ~CFisher();
114};
115 # endif

```

Apresenta-se na listagem o arquivo de implementação da classe CFisher.

Listing 6.4: Arquivo de implementação da classe CFisher.

```

116#include "CFisher.hpp"
117#include <math.h>
118#include <iostream>
119
120const double PI = 3.1416;
121using namespace std;
122
123    /// Costrutor sobrecarregado .
124CFisher::CFisher(int _d1, int _d2)
125{
126    d1 = _d1;
127    d2 = _d2;
128}
129    /// Metodo get .
130double CFisher::Probabilidade(double x)
131{
132    double argumento = (d1 * x / (d1 * x + d2));
133    return 1 - Beta_regularizada(argumento);
134}
135    /// Metodo calcular Beta
136double CFisher::Beta()
137{
138    int n = 1000;
139    double largura = 1.0 / n;
140    double integral = 0.0;
141    double centro = largura / 2.0;

```



```

142     for (int i = 1; i < n; i++)
143     {
144         integral += largura * F(centro);
145         centro += largura;
146     }
147     return integral;
148 }
149     /// Metodo calcular Beta_regulizada.
150 double CFisher::Beta_regularizada(double x)
151 {
152     int n = 1000;
153     double largura = x / n;
154     double integral = 0.0;
155     double centro = largura / 2.0;
156     for (int i = 1; i < n; i++)
157     {
158         integral += largura * F(centro);
159         centro += largura;
160     }
161     return integral / Beta();
162 }
163     /// Metodo calcular F
164 double CFisher::F(double x)
165 {
166     double a = d1 / 2.0;
167     double b = d2 / 2.0;
168     return (pow(x, a - 1.0) * pow(1 - x, b - 1.0));
169 }
170     ///Destrutor
171 CFisher::~CFisher()
172 {
173 }

```

Apresenta-se na listagem o arquivo com código da classe CFPC22\_02.

Listing 6.5: Arquivo de cabeçalho da classe CFPC22\_02.

```

174
175 #ifndef CFPC22_02_hpp
176 #define CFPC22_02_hpp
177
178 #include <string>
179
180 using namespace std;
181     /// Classe CFPC22_02 para calcular planejamento Fatorial de 2^2
182 class CFPC22_02
183 {
184 private:
185     double respostas[7]; ///< Vetor de respostas obida laboratório

```

```

186  double XtY[4];  ///< Matriz do planejamento transposta vezes o vetor
      resposta
187  double b[4];  ///< vetor de regressã
188  double tcalc[4];  ///< distribuição de Student
189  double p_coeficientes[4];  ///< valor estatitico amostral de stundent
190  const double erro[4] = {0.378, 0.5, 0.5, 0.5};  ///< Matriz erro
      aleatório associado ao modelo
191  const double XXinv[4] = {0.143, 0.25, 0.25, 0.25};  ///< Matriz do
      planejamento vezes sua inversa
192  const double nivel_significancia = 0.05;  ///< significa que a
      probabilidade de erro e de 0,05
193  double media = 0;  ///< mediada das resposta encontrada no laboratório
194  double media_pcentral = 0;  ///< Media do ponto central para calculos
      dos erros
195  double media_estimados = 0;  ///< media estimada do Y estimado pelo
      modelo
196  double y_estimado[7];  ///< Vetor de respostas estimado
197  double residuos[7];  ///< residuos gerado pelo modelo ex; (Yi lab) -( Yi
      estimado)
198  double ANOVA_reg[7];  ///< quadratica da regressao
199  double ANOVA_total[7];  ///< quadrática da variação total
200  double ANOVA_falt[7];  ///< quadrática da falta de ajusta (discrepancia)
201  double ANOVA_errrpuro[7];  ///< erro puro no laboratorio
202  double soma_quadratica[4];  ///< soma quadratica
203  double media_quadratica[4];  ///< soma quadratica da media
204  double Fcalc[2];  ///< calcular F distribuição de Fisher.
205  double p_variancia[2];  ///< valor estatitico amostral
206  double n_graus_liberdade[4] = {3, 3, 1, 2};  ///< graus de liberdade
207  double var_explicada;  ///< Analise de variancia
208  double max_var_explicada;  ///< Analise de variancia maxima
209  bool significativo[2];  ///< mostra se foi ou nao significativo
210  double ANOVA_reg_soma = 0;  ///< soma quadratica da regressao
211  double ANOVA_total_soma = 0;  ///< soma quadrática da variação tota
212  double ANOVA_falt_soma = 0;  ///< soma quadrática da falta de ajusta (
      discrepancia)
213  double ANOVA_errrpuro_soma = 0;  ///< somatorio do erro puro no
      laboratorio
214
215 public:
216     ///< construtor default
217     CFPC22_02();
218     ///< o usuario estrara com os valores manualmente obitidos em
      laboratorio .
219     void Entrada_manual();
220     ///< o usuario informara o arquivo com os dados obitidos em
      laboratorio.
221     void Ler_arquivo(string arquivo);
222     ///< ler dados para plot

```

```

223 void get_resposta(double _resp[7]);
224 /// ler dados para plot
225 void get_yestimado(double _yresp[7]);
226 /// o programa ira realizar uma sequencia de calculos estatiticos
para ANOVA.
227 void Calcular();
228 /// Mostrar na tela os valores caulculados da ANOVA.
229 void Mostrar_calculos();
230 /// mostrar os coeficientes referente a sua equação.
231 void Tabela_coeficientes();
232 /// Salvar os arquivos em .txt
233 void Arquivo_saida();
234 /// mostrar a tabela da ANOVA
235 void Tabela_variancia();
236 /// Destrutor default.
237 ~CFPC22_02();
238 };
239 #endif

```

Apresenta-se na listagem o arquivo de implementação da classe CFPC22\_02.

Listing 6.6: Arquivo de implementação da classe CFPC22\_02.

```

240 #include <string>
241 #include <fstream>
242 #include <iostream>
243 #include <cmath>
244
245 #include "CFPC22_02.hpp"
246 #include "CStudent.hpp"
247 #include "CFisher.hpp"
248
249 using namespace std;
250
251 CFPC22_02::CFPC22_02()
252 {
253 }
254 /// o usuario informara o arquivo com os dados obitidos em
laboratorio.
255 void CFPC22_02::Ler_arquivo(string arquivo)
256 {
257     ifstream entrada("./" + arquivo + ".txt"); /// ler arquivo em formato
.txt
258
259     for (int i = 0; i < 7; i++) /// percorre os valores de i=0 ate i=6
260     {
261         entrada >> respostas[i];
262     }
263     this->Calcular(); /// ponteiro para ter acesso a Calcular
264 }

```

```

265      /// o usuario estrara com os valores manualmente obitidos em
          laboratorio .
266 void CFPC22_02::Entrada_manual()
267 {
268     cout << "\nDigite o valor de y para x1=-1, x2=-1:"; /// o usuario
          entarrar com o primeiro valor do lab.
269     cin >> respostas[0]; /// amarzenar respostas 0
270     cout << "Digite o valor de y para x1=1, x2=-1:"; /// o usuario
          entarrar com o segundo valor do lab.
271     cin >> respostas[1]; /// amarzenar respostas 0
272     cout << "Digite o valor de y para x1=-1, x2=1:"; /// o usuario
          entarrar com o terceiro valor do lab.
273     cin >> respostas[2]; /// amarzenar respostas 0
274     cout << "Digite o valor de y para x1=1, x2=1:"; /// o usuario
          entarrar com o quarto valor do lab.
275     cin >> respostas[3]; /// amarzenar respostas 0
276     for (int i = 4; i < 7; i++)
277     {
278         cout << "Digite o valor de y do ponto central (" << i - 3 << "
          /3):"; /// valores do ponto central
279         cin >> respostas[i]; /// armazena os valores do ponto central
280     }
281     this->Calcular();
282 }
283 /// o programa ira realizar uma sequencia de calculos estatiticos
para ANOVA.
284 void CFPC22_02::Calcular()
285 {
286     double soma = 0;
287     for (int i = 0; i < 7; i++)
288     {
289         soma += respostas[i];
290     }
291
292     media = (soma / 7); ///soma das respostas dividido pela quantidade
          de amostras
293     media_pcentral = ((respostas[4] + respostas[5] + respostas[6]) / 3);
          /// media do ponto central
294
295     XtY[0] = soma;
296     XtY[1] = (respostas[1] + respostas[3] - respostas[0] - respostas[2])
          ;
297     XtY[2] = (respostas[2] + respostas[3] - respostas[0] - respostas[1])
          ;
298     XtY[3] = (respostas[0] + respostas[3] - respostas[1] - respostas[2])
          ;
299
300     CStudent student(2);

```

```
301     for (int i = 0; i < 4; i++)
302     {
303         b[i] = XXinv[i] * XtY[i];
304         tcalc[i] = b[i] / erro[i];
305         p_coeficientes[i] = student.Probabilidade(tcalc[i]);
306     }
307
308     y_estimado[0] = b[0] - b[1] - b[2] + b[3];
309     y_estimado[1] = b[0] + b[1] - b[2] - b[3];
310     y_estimado[2] = b[0] - b[1] + b[2] - b[3];
311     y_estimado[3] = b[0] + b[1] + b[2] + b[3];
312     y_estimado[4] = b[0];
313     y_estimado[5] = b[0];
314     y_estimado[6] = b[0];
315     for (int i = 0; i < 7; i++)
316     {
317         media_estimados += y_estimado[i];
318     }
319     media_estimados /= 7;
320
321     for (int i = 0; i < 7; i++)
322     {
323         residuos[i] = respostas[i] - y_estimado[i];
324     }
325
326     for (int i = 0; i < 7; i++)
327     {
328         ANOVA_reg[i] = pow(y_estimado[i] - media, 2);
329         ANOVA_reg_soma += ANOVA_reg[i];
330     }
331
332     for (int i = 0; i < 7; i++)
333     {
334         ANOVA_total[i] = pow(respostas[i] - media, 2);
335         ANOVA_total_soma += ANOVA_total[i];
336     }
337
338     for (int i = 0; i < 4; i++)
339     {
340         ANOVA_falt[i] = pow(y_estimado[i] - respostas[i], 2);
341         ANOVA_falt_soma += ANOVA_falt[i];
342     }
343     for (int i = 4; i < 7; i++)
344     {
345         ANOVA_falt[i] = pow(y_estimado[i] - media_pcentral, 2);
346         ANOVA_falt_soma += ANOVA_falt[i];
347     }
348     for (int i = 0; i < 4; i++)
```

```

349     {
350         ANOVA_errpuro[i] = pow(respostas[i] - respostas[i], 2);
351         ANOVA_errpuro_soma += ANOVA_errpuro[i];
352     }
353     for (int i = 4; i < 7; i++)
354     {
355         ANOVA_errpuro[i] = pow(respostas[i] - media_pcentral, 2);
356         ANOVA_errpuro_soma += ANOVA_errpuro[i];
357     }
358
359     soma_quadratica[0] = ANOVA_reg_soma;
360     soma_quadratica[1] = ANOVA_falt_soma + ANOVA_errpuro_soma;
361     soma_quadratica[2] = ANOVA_falt_soma;
362     soma_quadratica[3] = ANOVA_errpuro_soma;
363
364     for (int i = 0; i < 4; i++)
365     {
366         media_quadratica[i] = soma_quadratica[i] / n_graus_liberdade[i];
367     }
368     Fcalc[0] = media_quadratica[0] / media_quadratica[1];
369     Fcalc[1] = media_quadratica[2] / media_quadratica[3];
370
371     CFisher fisher1(n_graus_liberdade[0], n_graus_liberdade[1]);
372     p_variancia[0] = fisher1.Probabilidade(Fcalc[0]);
373
374     CFisher fisher2(n_graus_liberdade[2], n_graus_liberdade[3]);
375     p_variancia[1] = fisher2.Probabilidade(Fcalc[1]);
376     for (int i = 0; i < 2; i++)
377     {
378         if (p_variancia[i] < nivel_significancia) {
379             significativo[i] = true;
380         }
381         else {
382             significativo[i] = false;
383         }
384     }
385
386     var_explicada = 100 * (soma_quadratica[0] / ANOVA_total_soma);
387     max_var_explicada = 100 * ((ANOVA_total_soma - soma_quadratica[3]) /
388                                ANOVA_total_soma);
389
390     /// Mostrar na tela os valores calculados da ANOVA.
391     void CFPC22_02::Mostrar_calculos()
392     {
393         cout << "media:" << media << endl;
394         cout << "media_pc:" << media_pcentral << endl;
395         cout << "Xy_" << XtY[0] << "_" << XtY[1] << "_" << XtY[2] << "
396             _" << XtY[3] << "_\n";

```

```

395     for (int i = 0; i < 4; i++)
396     {
397         cout << "b_-" << b[i] << "\nt_-" << tcalc[i] << endl;
398     }
399     for (int i = 0; i < 7; i++)
400     {
401         cout << endl
402             << "y_estimado_-" << y_estimado[i];
403     }
404     for (int i = 0; i < 7; i++)
405     {
406         cout << endl
407             << "residuo_-" << residuos[i];
408     }
409     cout << "Media_estimados_-" << media_estimados << endl;
410     for (int i = 0; i < 7; i++)
411     {
412         cout << endl
413             << "ANOVA_reg_-" << ANOVA_reg[i];
414     }
415     for (int i = 0; i < 7; i++)
416     {
417         cout << endl
418             << "ANOVA_total_-" << ANOVA_total[i];
419     }
420     for (int i = 0; i < 7; i++)
421     {
422         cout << endl
423             << "ANOVA_falt_-" << ANOVA_falt[i];
424     }
425     for (int i = 0; i < 7; i++)
426     {
427         cout << endl
428             << "ANOVA_errpuro_-" << ANOVA_errpuro[i];
429     }
430     cout << endl
431         << "ANOVA_reg_soma_-" << ANOVA_reg_soma;
432     cout << endl
433         << "ANOVA_total_soma_-" << ANOVA_total_soma;
434     cout << endl
435         << "ANOVA_falt_soma_-" << ANOVA_falt_soma;
436     cout << endl
437         << "ANOVA_errpuro_soma_-" << ANOVA_errpuro_soma;
438     for (int i = 0; i < 4; i++)
439     {
440         cout << "\nmedia_-" << i << ":-" << media_quadratica[i];
441     }
442     cout << "\nFcalc1_-" << Fcalc[0] << "\np_-" << p_variancia[0];

```

```

443         cout << "\nFcalc2_-" << Fcalc[1] << "\np_-" << p_variancia[1];
444     cout << "\nVariacao_explicada_-" << var_explicada;
445     cout << "\nmax_variacao_explicada_-" << max_var_explicada;
446 }
447     /// mostrar os coeficientes referente a sua equação.
448 void CFPC22_02::Tabela_coeficientes()
449 {
450     cout << endl
451         << "Tabela_de_coeficientes";
452     cout << endl
453         << "-----";
454     cout << endl
455         << '\t' << "Efeitos\tErro\tt(2)\tp" << endl;
456     cout << "Media\t" << b[0] << '\t' << erro[0] << '\t' << tcalc[0] <<
        '\t' << p_coeficientes[0] << endl;
457     cout << "X1\t" << 2 * b[1] << '\t' << 2 * erro[1] << '\t' << tcalc
        [1] << '\t' << p_coeficientes[1] << endl;
458     cout << "X2\t" << 2 * b[2] << '\t' << 2 * erro[2] << '\t' << tcalc
        [2] << '\t' << p_coeficientes[2] << endl;
459     cout << "X12\t" << 2 * b[3] << '\t' << 2 * erro[3] << '\t' << tcalc
        [3] << '\t' << p_coeficientes[3] << endl;
460     cout << "-----" << endl;
461 }
462     /// mostrar a tabela da ANOVA
463 void CFPC22_02::Tabela_variancia()
464 {
465     cout << endl
466         << "Analise_de_variancia";
467     cout << endl
468         << "
        -----
        ";
469     cout << endl
470         << "FV\t\t" << "SQ\t" << "nGL\t" << "MQ\t" << "Fcalc.\t" << "p"
        << endl;
471     cout << "Regressão\t" << soma_quadrativa[0] << '\t' <<
        n_graus_liberdade[0] << '\t' << media_quadrativa[0] << '\t' <<
        Fcalc[0] << '\t' << p_variancia[0] << endl;
472     cout << "Resíduos\t" << soma_quadrativa[1] << '\t' <<
        n_graus_liberdade[1] << '\t' << media_quadrativa[1] << endl;
473     cout << "F.Ajuste\t" << soma_quadrativa[2] << '\t' <<
        n_graus_liberdade[2] << '\t' << media_quadrativa[2] << '\t' <<
        Fcalc[1] << '\t' << p_variancia[1] << endl;
474     cout << "Erro_Puro\t" << soma_quadrativa[3] << "\t" <<
        n_graus_liberdade[3] << '\t' << media_quadrativa[3] << endl;
475     cout << "Total\t\t" << ANOVA_total_soma << '\t' << 6 << endl;
476     cout << "
        -----
    
```



```

        " << endl;
477     cout << "-_RegressÃ£o_";
478     if (!significativo[0]) {
479         cout << "nÃ£o";
480     }
481     cout << "_Significativa" << endl;
482     cout << "-_F._Ajuste_";
483     if (!significativo[1]) {
484         cout << "nÃ£o";
485     }
486     cout << "_Significativa" << endl;
487     cout << "variaÃ§Ã£o explicada\t\t" << var_explicada << '%' << endl;
488     cout << "mÃ;x.de variaÃ§Ã£o explicÃ;vel\t\t" << max_var_explicada <<
        '%', << endl;
489 }
490     /// salvar arquivo em .txt
491 void CFPC22_02::Arquivo_saida(){
492     FILE *saida;
493     saida=fopen("dadosaida.txt", "wb");
494     fstream fout;
495     fout.open("dadosaida.txt");
496
497     fout << endl;
498     fout << "Tabela_de_coeficientes";
499     fout << endl;
500     fout << "-----";
501     fout << endl;
502     fout << '\t' << "Efeitos\tErro\tt(2)\tp" << endl;
503     fout << "Media\t" << b[0] << '\t' << erro[0] << '\t' << tcalc[0] <<
        '\t' << p_coeficientes[0] << endl;
504     fout << "X1\t" << 2 * b[1] << '\t' << 2 * erro[1] << '\t' << tcalc
        [1] << '\t' << p_coeficientes[1] << endl;
505     fout << "X2\t" << 2 * b[2] << '\t' << 2 * erro[2] << '\t' << tcalc
        [2] << '\t' << p_coeficientes[2] << endl;
506     fout << "X12\t" << 2 * b[3] << '\t' << 2 * erro[3] << '\t' << tcalc
        [3] << '\t' << p_coeficientes[3] << endl;
507     fout << "-----" << endl;
508
509     fout << "_Analises_da_variancia_e_os_coeficientes_significativos"<<
        endl;
510     fout << "FV\t\t" << "SQ\t" << "nGL\t" << "MQ\t" << "Fcalc.\t" << "p"
        << endl;
511     fout << "RegressÃ£o\t" << soma_quadrativa[0] << '\t' <<
        n_graus_liberdade[0] << '\t' << media_quadrativa[0] << '\t' <<
        Fcalc[0] << '\t' << p_variancia[0] << endl;
512     fout << "Residuos\t" << soma_quadrativa[1] << '\t' <<
        n_graus_liberdade[1] << '\t' << media_quadrativa[1] << endl;
513     fout << "F._Ajuste\t" << soma_quadrativa[2] << '\t' <<

```

```

        n_graus_liberdade[2] << '\t' << media_quadratica[2] << '\t' <<
        Fcalc[1] << '\t' << p_variancia[1] << endl;
514 fout << "Erro_Puro\t" << soma_quadratica[3] << "\t" <<
        n_graus_liberdade[3] << '\t' << media_quadratica[3] << endl;
515 fout << "Total\t\t" << ANOVA_total_soma << '\t' << 6 << endl;
516 fout << "
        -----
        " << endl;
517 fout << "-_Regressão";
518 if (!significativo[0]) {
519     cout << "Não";
520 }
521 fout << "_Significativa" << endl;
522 fout << "-_F._Ajuste_";
523 if (!significativo[1]) {
524     cout << "Não";
525 }
526 fout << "_Significativa" << endl;
527 fout << "variação explicada\t\t" << var_explicada << '%' << endl;
528 fout << "máx.de variação explicável\t\t" << max_var_explicada <<
        '%' << endl;
529
530 fout.close();
531 }
532     /// redefinindo vetor respostas
533 void CFPC22_02::get_resposta(double _resp[7])
534 {
535     for (int i=0;i<7;i++)
536     {
537         _resp[i]=respostas[i];
538     }
539 };
540     /// redefinindo y_estimado
541 void CFPC22_02::get_yestimado(double _yresp[7])
542 {
543     for (int i=0;i<7;i++)
544     {
545         _yresp[i]=y_estimado[i];
546     }
547 };
548 /// Destrutor default.
549 CFPC22_02::~~CFPC22_02()
550 {
551 }

```

Apresenta-se na listagem a seguir a implementação do Main.

Listing 6.7: Arquivo de implementação da função main().

```
553 #include <iostream>
```

```

554#include "CFPC22_02.hpp"
555#include <string>
556#include "CFisher.hpp"
557#include "CStudent.hpp"
558#include "CGnuplot.h"
559#include "vector"
560#include <fstream>
561
562using namespace std;
563
564
565int main()
566{
567    double tmp1[7],tmp2[7]; //vetor no estilo de C
568    vector<double> tmp11, tmp21; //vetor no estilo de C++
569
570    int entrada = 0;
571    CFPC22_02 planejamento; // objeto planejamento
572
573    cout << "\n\n#####_PLANEJAMENTO_FATORIAL_v1.0_#####\nAutor:_
        Jhonny_Carvalho;_LENEP-UENF_(2019)\n
        -----\n"; // Título em
        tela
574
575    while (entrada != 1 && entrada != 2)
576    {
577        cout << "Selecione_o_modode_entrada_de_dados:\n1_-_
            Utilizar_um_arquivo_de_entrada\n2_-_Entrada_manual\n"
            ;
578        cin >> entrada; cin.get();
579    }
580    if (entrada == 1)
581    {
582        string arquivo;
583        cout << "Lista_de_arquivos.txt):_";
584        system("ls*.txt"); // lista dos arquivos txt
585        cout << "Digite_o_nome_do_arquivo(sem.txt):_";
586        cin >> arquivo; cin.get();
587        planejamento.Ler_arquivo(arquivo); // le dados do
            arquivo
588    }
589    else
590    {
591        planejamento.Entrada_manual(); // entrada manual
592    }
593
594    planejamento.Tabela_coeficientes(); //calcula coef
595    planejamento.Tabela_variancia(); //calcula variancia

```

```

596     planejamento.Arquivo_saida();/// salvar arquivos
597     planejamento.get_resposta(tmp1);
598     planejamento.get_yestimado(tmp2);
599
600     for (int i=0;i<7;i++)
601     {
602         tmp11.push_back(tmp1[i]);
603         tmp21.push_back(tmp2[i]);
604         //cout << tmp1[i] << "," << tmp2[i] << endl; //vetor c
605         cout << tmp11[i] << "," << tmp21[i] << endl; //vetor c++
606     }
607
608     cout << "size=" << tmp11.size() << ",size=" << tmp21.size() << endl; //
        vetor c++
609
610 // Plot do grafico utilizando CGnuplot:
611
612     //Gnuplot::set_GNUPlotPath("C:/program files/gnuplot/bin"); //
        setando Gnuplot no Windows
613     //Gnuplot::Terminal("win"); //Utilizando terminal do Windows
614
615     CGnuplot gp("points"); //Define o estilo do gráfico
616     gp.Grid(); // Habilitar grid do gráfico plotada
617     gp.set_pointsize (2.0); //Setando tamanho do ponto
618     gp.set_title ("Plot: Observados VS Estimados"); // Definindo título
        para o gráfico
619     gp.set_ylabel("Estimados"); //Definindo nome do Eixo Y
620     gp.set_xlabel("Observados"); //Definindo nome do Eixo X
621     gp.set_yautoscale (); // Seta automaticamente os limites para o eixo
        Y
622     gp.set_xautoscale (); // Seta automaticamente os limites para o eixo
        X
623     gp.PlotVector(tmp11,tmp21,"Dados"); // Plotando vetor de dados (
        observados vs estimados)
624
625     cout << "Digite 'enter' para encerrar.\n";
626
627     /// ofstream data ("resultados.txt", ofstream::out | ios::app ); //
        Construindo a saída de dados p/ arquivo
628     /// data << tmp1;
629     /// data.close();
630     gp.SaveTo("grafico_Obs-VS-Estm","png"); // Salvando sessão do
        gnuplot para um arquivo
631     cin.get();
632     cin.get();
633
634     return 0;
635 }

```

# Chapter 7

## Teste

Todo projeto de engenharia passa por uma etapa de testes. Neste capítulo apresentamos alguns testes do software desenvolvido. Estes testes devem dar resposta aos diagramas de caso de uso inicialmente apresentados (diagramas de caso de uso geral e específicos).

### 7.1 Teste 1: Descrição

O presente trabalho apresenta interface em modo texto. Veja na Figura 7.1 a tela inicial do software, o usuario tem a opção de entrar com os dados de maneira manual ou com arquivo .txt, a interface ca da seguinte maneira,

### 7.2 Teste 2: Resultados

Neste teste foi introduzido os dados de entrada por arquivo .txte para o planejamento. O software realiza os cálculos e exibe os seguintes resultados na figura abaixo. Além dos resultados exibidos no console, o software salva em disco arquivos .txt contendo o resultados dos calculados, que são apresentados na Figura 7.3 .

```
##### PLANEJAMENTO FATORIAL v1.0 #####
Autor: Jhonny Carvalho; LENEP-UENF (2019)
-----
Selecione o modo de entrada de dados:
1 - Utilizar um arquivo de entrada
2 - Entrada manual
█
```

Figure 7.1: Tela do programa mostrando xxx

```
1 - Utilizar um arquivo de entrada
2 - Entrada manual
1
dados-lab.txt resultados.txt
Lista de arquivos.txt): Digite o nome do arquivo (sem .txt): dados-lab

Tabela de coeficientes
-----
Media      Efeitos Erro      t(2)      p
X1          9.5      1      9.5      0.0112476
X2         10.5      1     10.5     0.00923455
X12         0.5      1      0.5     0.687277
-----

Analise de variancia
-----
FV      SQ      nGL      MQ      Fcalc.  p
Regressão      200.771 3      66.9237 94.3381 0.00202821
Resíduos      2.12821 3      0.709403
F. Ajuste      0.128208      1      0.128208      0.128208      0.754557
Erro Puro      2      2      1
Total      202.857 6
-----

- Regressão Significativa
- F. Ajuste não Significativa
variação explicada      98.9717%
máx. de variação explicável      99.0141%
45,45.162
54,54.162
55,55.162
65,65.162
56,54.912
54,54.912
55,54.912
size=7, size=7
Digite 'enter' para encerrar.
█
```

Figure 7.2: Tela do programa mostrando resultados

# Chapter 8

## Documentação

Todo projeto de engenharia precisa ser bem documentado. Neste sentido, apresenta-se neste capítulo a documentação de uso do "software Planejamento Fatorial 2<sup>2</sup>". Esta documentação tem o formato de uma apostila que explica passo a passo como usar o software.

### 8.1 Documentação do usuário

Descreve-se aqui o manual do usuário, um guia que explica, passo a passo a forma de instalação e uso do software desenvolvido.

#### 8.1.1 Como instalar o software

Abra o terminal, vá para o diretório onde está o projeto, compile o programa e, depois o execute. Logo após, siga os seguintes passos:

Entre com os valores dos 7 resultados obtidos em laboratórios de acordo com o Planejamento Fatorial, sendo os 3 últimos o ponto central usados para cálculo dos erros.

- Primeiro valor de entrada (-)  $X_1$  e (+)  $X_2$ ;
- Segundo valor de entrada (-)  $X_1$  e (-)  $X_2$ ;
- Terceiro valor de entrada (+)  $X_1$  e (+)  $X_2$ ;
- Quarto valor de entrada (+)  $X_1$  e (-)  $X_2$ ;
- E os três últimos valores de entrada (0)  $X_1$  e (0)  $X_2$ .

Após isso serão mostrados no console os resultados, e serão salvos “.txt” com o resultados e os gráficos em “.png”.

## 8.2 Documentação para desenvolvedor

Apresenta-se nesta seção a documentação para o desenvolvedor, isto é, informações para usuários que queiram modificar, aperfeiçoar ou ampliar este software.

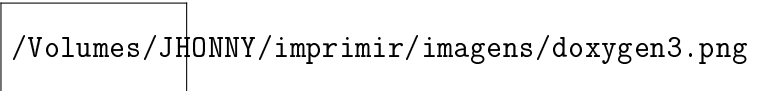
### 8.2.1 Dependências

Para compilar o software é necessário atender as seguintes dependências:

- Instalar o compilador `g++` da GNU disponível em <http://gcc.gnu.org>. Para instalar no GNU/Linux use o comando `yum install gcc`.
- Biblioteca `CGnuplot`; os arquivos para acesso a biblioteca `CGnuplot` devem estar no diretório com os códigos do software;
- O software `gnuplot`, disponível no endereço <http://www.gnuplot.info/>, deve estar instalado. É possível que haja necessidade de setar o caminho para execução do `gnuplot`.
- No sistema operacional Windows: Instalar o compilador apropriado; Recomenda-se o CodeBlocks 16.01.

### 8.2.2 Como gerar a documentação usando doxygen

A documentação do código do software deve ser feita usando o padrão JAVADOC, conforme apresentada no Capítulo - Documentação, do livro texto da disciplina. Depois de documentar o código, use o software `doxygen` para gerar a documentação do desenvolvedor no formato html. O software `doxygen` lê os arquivos com os códigos (\*.h e \*.cpp) e gera uma documentação muito útil e de fácil navegação no formato html.



/Volumes/JHONNY/imprimir/imagens/doxygen3.png

Figure 8.1: Documentacao do doxygen



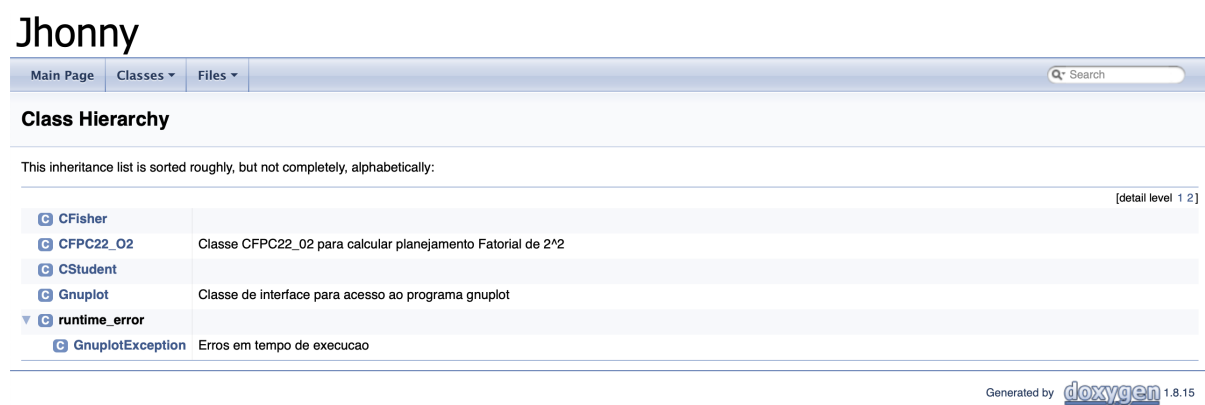


Figure 8.2: Documentacao doxyen

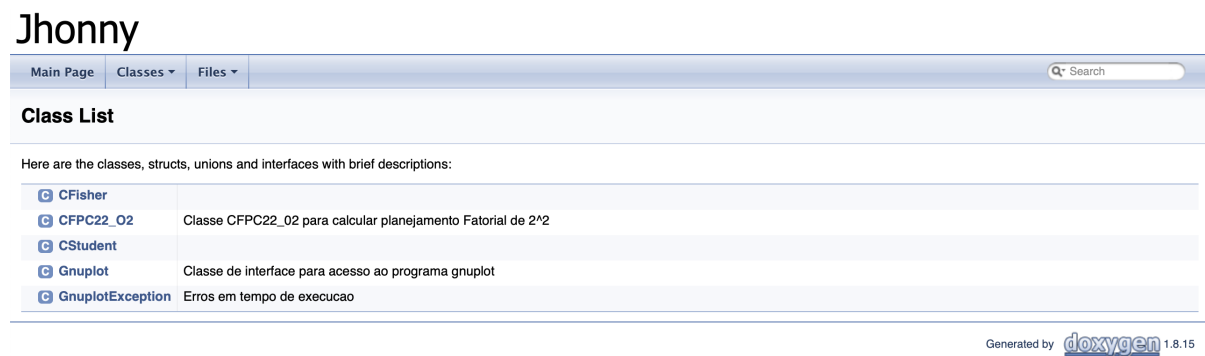


Figure 8.3: Documentacao do doc.gen



# Bibliography

- [Ashoori and Vanini, 2016] Ashoori, A. and Vanini, S. S. (2016). Thermal buckling of annular microstructure-dependent functionally graded material plates resting on an elastic medium. *Composites Part B: Engineering*, 87:245–255. 2
- [Bissada et al., 2016] Bissada, K. A., Tan, J., Szymczyk, E., Darnell, M., and Mei, M. (2016). Group-type characterization of crude oil and bitumen. part i: Enhanced separation and quantification of saturates, aromatics, resins and asphaltenes (sara). *Organic Geochemistry*, 95:21–28. 1, 2
- [Brasil and Araújo, 2014] Brasil, N. I. and Araújo, M. A. S. (2014). Processamento de petróleo e gás: petróleo e seus derivados, processamento primário, processos de refino, petroquímica, meio ambiente. 2
- [Breitkreitz et al., 2014] Breitkreitz, M. C., Souza, A. M. d., Poppi, R. J., et al. (2014). Experimento didático de quimiometria para planejamento de experimentos: avaliação das condições experimentais na determinação espectrofotométrica de ferro ii com o-fenantrolina. um tutorial, parte iii. *Quim. Nova*, 37(3):564–573. 9
- [Collins et al., 2006] Collins, C. H., Braga, G. L., and Bonato, P. S. (2006). Fundamentos de cromatografia. In *Fundamentos de cromatografia*. Unicamp. 2
- [Cunico et al., 2008] Cunico, M., Cunico, M., Miguel, O., Zawadzki, S., Peralta-Zamora, P., and Volpato, N. (2008). Planejamento fatorial: Uma ferramenta estatística valiosa para a definição de parâmetros experimentais empregados na pesquisa científica. *Visão Acadêmica Acadêmica*, 9(1). 9
- [Fan and Buckley, 2002] Fan, T. and Buckley, J. S. (2002). Rapid and accurate sara analysis of medium gravity crude oils. *Energy & Fuels*, 16(6):1571–1575. 2
- [Jiang et al., 2008] Jiang, C., Larter, S. R., Noke, K. J., and Snowdon, L. R. (2008). Tlc-fid (iatroscan) analysis of heavy oil and tar sand samples. *Organic Geochemistry*, 39(8):1210–1214. 2
- [Karlsen and Larter, 1991] Karlsen, D. A. and Larter, S. R. (1991). Analysis of petroleum fractions by tlc-fid: applications to petroleum reservoir description. *Organic Geochemistry*, 17(5):603–617. 2

- [Killops and Killops, 2013] Killops, S. D. and Killops, V. J. (2013). *Introduction to organic geochemistry*. John Wiley & Sons. 1
- [Teófilo et al., 2006] Teófilo, R. F., Ferreira, M. M., et al. (2006). Quimiometria ii: planilhas eletrônicas para cálculos de planejamentos experimentais, um tutorial. *Química Nova*, 29(2):338. 9

# Index

Análise orientada a objeto, 21  
AOO, 21  
atributos, 29  
  
Casos de uso, 6  
colaboração, 23  
comunicação, 23  
Concepção, 4  
Controle, 28  
  
Diagrama de colaboração, 23  
Diagrama de componentes, 29  
Diagrama de execução, 30  
Diagrama de máquina de estado, 24  
Diagrama de sequência, 22  
  
Efeitos do projeto nas heranças, 29  
Efeitos do projeto nos métodos, 29  
Elaboração, 8  
especificação, 4  
estado, 24  
Eventos, 22  
  
Heranças, 29  
heranças, 29  
  
Implementação, 32  
  
métodos, 29  
Mensagens, 22  
modelo, 28, 29  
  
Plataformas, 28  
POO, 28  
Projeto do sistema, 27  
Projeto orientado a objeto, 28  
Protocolos, 27  
  
Recursos, 27