

UNIVERSIDADE FEDERAL DE SÃO CARLOS  
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA  
DEPARTAMENTO DE QUÍMICA

ANDRÉ LUIZ MACHADO SIMÃO DE SOUZA

**PYTHON APLICADO EM PLANEJAMENTO  
FATORIAL EM QUÍMICA**

São Carlos-SP

2022

ANDRÉ LUIZ MACHADO SIMÃO DE SOUZA

# **PYTHON APLICADO EM PLANEJAMENTO FATORIAL EM QUÍMICA**

Trabalho de Conclusão de Curso, apresentado a  
Universidade Federal de São Carlos – Campus  
São Carlos, como parte das exigências para a  
obtenção do título de Bacharelado em Química.

Orientador: Prof. Dr. Edenir Rodrigues Pereira Filho

São Carlos-SP

2022



**FUNDAÇÃO UNIVERSIDADE FEDERAL DE SÃO CARLOS**

**DEPARTAMENTO DE QUÍMICA - DQ/CCET**

Rod. Washington Luís km 235 - SP-310, s/n - Bairro Monjolinho, São Carlos/SP, CEP  
13565-905

Telefone: (16) 33518206 - <http://www.ufscar.br>

DP-TCC-FA nº 18/2022/DQ/CCET

**Graduação: Defesa Pública de Trabalho de Conclusão de Curso**

**Folha Aprovação (GDP-TCC-FA)**

**FOLHA DE APROVAÇÃO**

**ANDRÉ LUIZ MACHADO SIMÃO DE SOUZA**

**PYTHON APLICADO EM PLANEJAMENTO FATORIAL EM QUÍMICA**

**Trabalho de Conclusão de Curso**

**Universidade Federal de São Carlos - Campus São Carlos**

São Carlos, 08 de setembro de 2022

**ASSINATURAS E CIÊNCIAS**

<b>Cargo/Função</b>	<b>Nome Completo</b>
Orientador	Prof. Dr. Edenir Rodrigues Pereira Filho
Membro da Banca 1	Dr. Vinícius Câmara Costa
Membro da Banca 2	Prof. Dr. José Mário de Aquino



Documento assinado eletronicamente por **Caio Marcio Paranhos da Silva, Professor(a)**, em 27/09/2022, às 08:33, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site <https://sei.ufscar.br/autenticacao>, informando o código verificador **0828789** e o código CRC **FFB507A5**.

**Referência:** Caso responda a este documento, indicar expressamente o Processo nº 23112.035411/2022-78

SEI nº 0828789

Modelo de Documento: Grad: Defesa TCC: Folha Aprovação, versão de 02/Agosto/2019

# Agradecimentos

Sou grato à Deus por ter saúde e determinação nas metas traçadas juntos com meus familiares, especialmente minha mãe Cristiane, a mulher que me apoiou e acompanhou a minha trajetória e que iluminou o meu caminho desde sempre.

Agradeço aos meus colegas do curso de Bacharelado em Química, em especial ao Pedro Ferreira, Neri Godoy e Gabriel Uechi, por terem sido parceiros e por ter compartilhado todos os momentos bons (ou não tão bons), o importante é que conhecimento foi trocado em todo este percurso e tenho certeza que sem eles não seria tão incrível os momentos vividos neste período acadêmico.

Sou grato a todos os professores que fizeram parte na jornada rumo à Universidade Federal de São Carlos, especialmente para a Escola Estadual Paschoal Carlos Magno, onde tive contato com excelentes professores que me incentivaram a estudar para os vestibulares, alimentando o meu sonho de um dia estar em uma universidade pública.

Por fim, mas não o menos importante, agradeço todos os professores, do curso de Bacharelado em Química que estiveram presentes comigo e com minhas dificuldades nessa caminhada neste período de aprendizagem e formação, os senhores com certeza são parte dessa vitória.

Agradeço em especial ao professor e orientador Dr. Edenir Rodrigues Pereira Filho que apoiou toda a construção do Projeto Planejamento Fatorial em Python.

Obrigado a todos!

## RESUMO

O escopo deste trabalho de conclusão de curso envolve aplicação de uma nova ferramenta para planejamento fatorial para a otimização de experimentos com o uso da linguagem de programação Python em que o autor adaptou e compilou três rotinas do programa livre Octave em uma biblioteca nomeada de “pde.py”. Para tal ilustração do uso da ferramenta, foram reproduzidos os quatros exemplos abordados no artigo publicado pela revista *Química Nova* em que é abordado e discutido a aplicação da rotina programacional em planejamento fatorial completo (1), fracionário (2), tipo *Doehlert* (3) e *Box-Behnken* (4). O intuito do trabalho é apresentar em etapas os procedimentos realizados em cada exemplo com esta nova ferramenta, deste modo foram preparados 63 vídeos que auxiliam o leitor em todo o desenvolvimento no conteúdo abordado.

**Palavras-chave:** Python. Planejamento Fatorial. Análise de Variância. Modelo de Regressão. Gráficos de Superfície de Resposta e Contorno. Cálculo de Efeitos.

## Abstract

### Python applied to factorial design in chemistry

The scope of this course conclusion work involves the application of a new tool for factorial planning for the optimization of experiments using the Python programming language, where the author adapted and compiled three routines of the free program Octave in a library named "pde.py". To illustrate the use of the tool, the four examples discussed in the article published by the journal *Química Nova* were reproduced, in which the application of the programming routine in complete (1), fractional (2), Doehlert type (3) factorial planning is approached and discussed. and Box-Behnken (4). The purpose of the work is to present in stages the procedures performed in each example with this new tool, in this way 63 videos were prepared that help the reader throughout the development of the content addressed.

**Keywords:** Python. Factor Planning. Analysis of Variance. Regression Model. Surface and Corner Graphics. Effects Calculation.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Uso de ferramentas aplicadas para ciência de dados desde 2015. . . . .	10
Figura 2 – Procedimento geral para realização e interpretação correta de um planejamento fatorial. . . . .	13
Figura 3 – Gráfico de porcentagem relativa aos efeitos calculados (a). Representação dos efeitos calculados em um histograma (b). Gráfico de efeitos em relação à distribuição gaussiana (c). . . . .	15
Figura 4 – Imagens geradas pela rotina programacional no Python. Gráfico de porcentagem relativa aos efeitos calculados ao exemplo 1 (a). Gráfico de probabilidade aplicado ao exemplo 1(b). . . . .	21
Figura 5 – Imagens geradas pela rotina programacional no Python. Gráfico de porcentagem relativa aos contrastes calculados ao exemplo 2 para resposta 1 (a) e 2 (c). Gráfico de probabilidade aplicado ao exemplo 2 para resposta 1 (b) e 2 (d). . . . .	23
Figura 6 – Imagens geradas pela rotina programacional no Python. Gráficos das médias quadráticas (a), teste F1 para MQR e MQr (b) teste F2 para MQfaj e MQep (c) e máxima variação explicada $R^2$ e explicável $R_m^2$ máximo aplicado ao exemplo 3 (d). . . . .	26
Figura 7 – Imagens geradas pela rotina programacional no Python. Gráficos valores experimentais versus previstos (a), previsto versus resíduos e histograma de resíduos (b) teste F2 para MQfaj e MQep (c) e coeficientes de regressão. . . . .	28
Figura 8 – Imagens geradas pela rotina programacional no Python. Gráficos de superfície de resposta e contorno para o modelo gerado pelo exemplo 3. . . . .	29

## LISTA DE TABELAS

Tabela 1 – Modelo de tabela ANOVA gerada pela rotina programacional "pde.py". . . . .	18
Tabela 2 – Planejamento fatorial completo $2^3$ para verificar a influência de 3 variáveis (v1 a v3) nos sinais de fluorescência de Sb. . . . .	20
Tabela 3 – Planejamento fatorial fracionário $2^{4-1}$ para analisar a influência de 4 variáveis(v1 a v4) em duas respostas . . . . .	22
Tabela 4 – Planejamento fatorial Doehlert para analisar a influência de 2 variáveis no rendimento na intensidade de fluorescência de antimônio. . . . .	24
Tabela 5 – Planejamento fatorial Box-Behnken para analisar a influência de 4 variáveis no rendimento na produção do benzaldeído. . . . .	30
Tabela 6 – Lista de vídeos preparados para melhor entendimento do uso do Python em planejamento fatorial em química. . . . .	34



## LISTA DE ACRÔNIMOS

HCl	Ácido clorídrico
Sb	Antimônio
NaBH <sub>4</sub>	Borohidreto de sódio
H <sub>2</sub> O <sub>2</sub>	Peróxido de hidrogênio
ANOVA	<i>Analysis of variance</i>
GL	Graus de liberdade
HCA	<i>Hierarchical Cluster Analysis</i>
PCA	<i>Principal Component Analysis</i>
PCR	<i>Principal Component Regression</i>
PEG	Polietileno glicol
SQep	Soma quadrática do erro puro
SQfaj	Soma quadrática de falta de ajuste
MQfaj	Média quadrática de falta de ajuste
SQR	Soma quadrática de regressão
SQres	Soma quadrática de resíduos
SQT	Soma Quadrática total
MQres	Média quadrática dos resíduos
MSLof	<i>Mean of square of lack of fit</i>
MSPE	<i>Mean of square of pure error</i>
MSreg	<i>Mean of square of regression</i>
MSres	<i>Mean of square of residues</i>

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>9</b>
<b>2</b>	<b>OBJETIVOS</b>	<b>11</b>
2.1	Objetivo Geral	11
2.2	Objetivos Específicos	11
<b>3</b>	<b>REVISÃO TEÓRICA</b>	<b>12</b>
3.1	Etapas envolvidas no planejamento fatorial	12
3.2	Planejamento fatorial e triagem de variáveis	12
3.2.1	Cálculo de efeitos	13
3.2.2	Identificação de efeitos importantes	14
3.2.3	Réplicas do ponto central	15
3.2.4	Padrões de confundimentos	16
3.3	Modelo de regressão	17
3.3.1	Análise de variância (ANOVA)	17
<b>4</b>	<b>ROTINA PROGRAMACIONAL EM PYTHON</b>	<b>18</b>
<b>5</b>	<b>EXEMPLO 1: PLANEJAMENTO FATORIAL COMPLETO</b>	<b>19</b>
5.1	Descrição do método: Exemplo 1	19
5.2	Resultados e discussão: Exemplo 1	20
<b>6</b>	<b>EXEMPLO 2: PLANEJAMENTO FATORIAL FRACIONÁRIO</b>	<b>21</b>
6.1	Descrição do método: Exemplo 2	21
6.2	Resultados e discussão: Exemplo 2	22
<b>7</b>	<b>EXEMPLO 3: PLANEJAMENTO FATORIAL DOEHLERT</b>	<b>23</b>
7.1	Descrição do Método: Exemplo 3	23
7.2	Resultados e discussão: Exemplo 3	25
7.3	Resultados e discussão: Exemplo 4	30
<b>8</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>31</b>
	<b>REFERÊNCIAS</b>	<b>32</b>
	<b>APPENDIX A – PLAYLIST 24: PYTHON</b>	<b>34</b>

# 1 INTRODUÇÃO

Após a metade do século XX, a química passou por mudanças drásticas com a química teórica aliada aos computadores. Além disso, a experimentação também evoluiu com o uso dos instrumentos ópticos e eletrônicos, fazendo que a geração de dados fosse aumentada substancialmente. Assim, com os espectrômetros e cromatógrafos inseridos em laboratórios analíticos, gerou-se um gigantesco crescimento na variedade de dados experimentais produzidos com o avanço da tecnologia na área instrumental. Dessa maneira, o desenvolvimento da Quimiometria surgiu com a necessidade de converter extenso conjunto de dados em informação útil ao experimentador, utilizando matemática e estatística como recurso (FERREIRA, 2015).

Dentre diversas definições encontradas na literatura para a “Quimiometria”, a definição original escolhida pela Sociedade de Quimiometria é: “A quimiometria é a aplicação de ferramentas matemáticas e estatísticas à química” (KOWALSKI, 1975). De modo que a Quimiometria é dividida em quatro âmbitos: Análise exploratória de dados químicos com o uso de PCA (*Principal Component Analysis*) e HCA (*Hierarchical Cluster Analysis*); Calibração multivariada com a utilização da PCR (*Principal Component Regression*) e PLS (*Partial Least Squares*) (BRO; SMILDE, 2014); Proposição de modelos de classificação; Emprego de planejamento fatorial para o estudo de condições experimentais ideais (LEARDI, 2009).

Deste modo, o escopo deste trabalho será voltado ao planejamento fatorial para a otimização de experimentos em uma aplicação da linguagem de programação Python no qual o autor adaptou e compilou três rotinas do programa livre Octave de Pereira e Pereira-Filho (2018) em uma biblioteca. Para tal ilustração do uso da ferramenta, foram reproduzidos os quatro exemplos abordados no artigo publicado pela revista *Química Nova* “Aplicação de programa computacional livre em planejamento de experimentos: Um tutorial.” De Pereira e Pereira-Filho (2018).

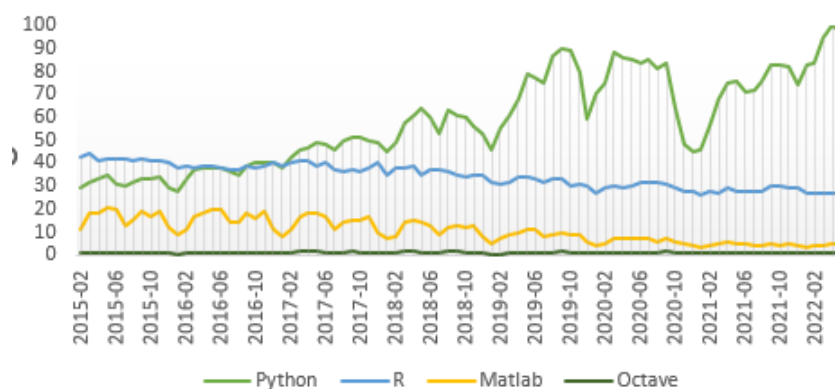
Atualmente, o tema sobre aplicações do emprego de planejamento fatorial para otimização de experimentos está bem difundido pela literatura. No Brasil, houve grande divulgação do tema em materiais na língua portuguesa em 2015 que colaborou muito com a comunidade científica, sobretudo o Barros Neto *et al* (2010). Todavia, as publicações que envolvem o tema não abrangeram o uso de programas ou linguagens de programação que são de fácil acesso e que são popularizadas pelos seus usuários (PEREIRA; PEREIRA-FILHO, 2018). Dessa maneira, o uso da linguagem Python para a aplicação em planejamento fatorial em química se deve as seguintes razões:

- O Python é uma linguagem de programação livre e de fácil acesso;
- É uma linguagem de alto nível, ou seja, é relativamente intuitiva e de fácil aprendizagem;

- O uso da linguagem Python vem crescendo vertiginosamente desde 2008, sendo uma linguagem com grande volume de documentação;
- Comunidade ativa, fóruns como *GitHub* e *Stack Overflow* contribuem em projetos tipo *Open-Source* que facilitam a construção de novos projetos;
- Possibilidade de automatizar processos envolvendo Quimiometria.

A Figura 1 mostra a frequência do crescimento desta ferramenta em relação ao R, Matlab e Octave durante os últimos 14 anos. Obviamente, a principal ferramenta utilizada é o Microsoft Excel devido a sua popularidade, porém a pesquisa realizada pelo *Google Trends* mostra o crescimento do Python diante das outras ferramentas que são utilizadas para a área da ciência de dados. Vale salientar que na pesquisa não estão inclusas todas as linguagens ou programas já registrados para o emprego de planejamento fatorial e também não significa que esta linguagem seja a mais apropriada como ferramenta quimiométrica em relação as demais.

Figura 1 – Uso de ferramentas aplicadas para ciência de dados desde 2015.



Fonte: Adaptado, *Google Trends*, 2022.

Apesar da contextualização do crescimento e popularização do Python, há poucos registros de uma biblioteca que entregam funções para o planejamento fatorial e além do uso desta ferramenta pode ser dificultoso para iniciantes de programação.

Devido a isso, preparou-se uma Playlist contendo 63 vídeos que auxiliam o leitor na utilização desta linguagem de programação (vide Apêndice A). Os vídeos incluem a instalação de editores de código e preparo do ambiente de trabalho, além de incluir vídeos que abordam sobre Python básico tal como: tipos de variáveis; estruturas condicionais; estruturas de repetição; funções e módulos e uso de pacotes externos utilizados, esta parte introdutória à linguagem Python é contemplada até o vídeo 47 play24. Os vídeos foram divididos em 6 seções em que a última se trata justamente dos exemplos abordados neste trabalho de conclusão de curso.

## 2 OBJETIVOS

### 2.1 Objetivo Geral

Realizar uma revisão teórica sobre a técnica de planejamento fatorial para otimização de experimentos e reproduzir os resultados dos 4 exemplos apresentados no artigo publicado na revista *Química Nova* de Pereira e Pereira-Filho (2018), utilizando a biblioteca “pde.py”, rotina programacional que foi elaborada pelo autor para tratamento de dados em Python. Juntamente a isso, elaborar uma playlist para auxiliar o leitor com o uso da ferramenta, que contém: instalação, conceitos básicos e aplicação em planejamento fatorial.

### 2.2 Objetivos Específicos

Através da resolução dos exemplos propostos pelo artigo mencionado, os objetivos específicos são:

- Determinar a influência de 3 variáveis na intensidade de fluorescência visando a determinação de espécies de antimônio em uma configuração de planejamento fatorial completo.
- Determinar a influência de 4 variáveis em relação às respostas de diâmetro e distribuição de partículas de nano-carbono em uma configuração de planejamento fatorial fracionário.
- Gerar e avaliar um modelo de regressão envolvendo duas variáveis em um planejamento fatorial tipo Doehlert. Na sequência, determinar as condições ideais de experimentação através da construção de gráficos de superfície de resposta e contorno.
- Determinar as condições ideais de experimentação em uma aplicação envolvendo 4 variáveis no rendimento de benzaldeído empregando um planejamento fatorial tipo Box-Behnken.

## 3 REVISÃO TEÓRICA

### 3.1 Etapas envolvidas no planejamento fatorial

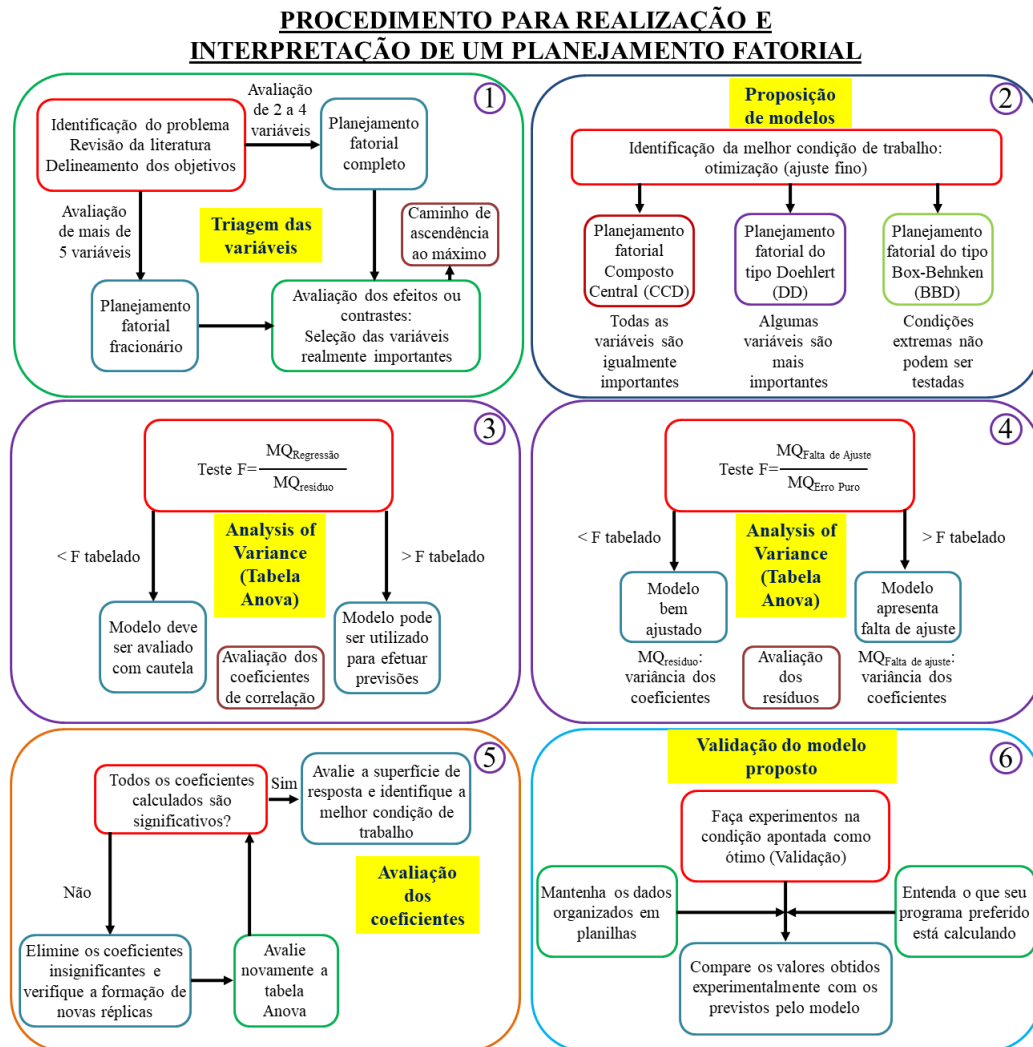
A construção do algoritmo da biblioteca “Planejamento de Experimentos” pela linguagem de programação livre Python é organizado de acordo com o procedimento indicado pela Figura 2. Este é descrito por Pereira e Pereira-Filho (2018) no artigo publicado na revista *Química Nova* “Aplicação de programa computacional livre em planejamento de experimentos: um tutorial”. Assim, o procedimento do planejamento fatorial segue uma organização de modo lógico representada em 6 etapas: triagem de variáveis, onde é determinado os fatores que serão incluídos no planejamento e a seleção dos efeitos significativos ao experimento (1); proposição do modelo para identificar as melhores condições experimentais para a otimização do experimento (2); análise de variância (ANOVA) para avaliar os coeficientes de determinação (3) e avaliação dos resíduos (4); avaliação dos coeficientes gerados pela ANOVA para determinar se o modelo construído possui ou não falta de ajuste (5) e, por fim, a validação do modelo proposto para realizar o experimento com as condições indicadas como ótimas (6).

### 3.2 Planejamento fatorial e triagem de variáveis

O planejamento fatorial completo é uma etapa primordial para determinar as variáveis que são importantes ou que possuem efeito sobre a resposta do problema. (GUARDIA, 1999). Para isso, o experimentador precisa identificar os fatores que envolvem o processo, por exemplo: temperatura, pH, pressão, força iônica, concentração dos reagentes, entre outros. Então, deve-se realizar uma triagem com o auxílio de um levantamento bibliográfico para definir as variáveis e o número de níveis de modo que seja viável em termos pragmáticos. Dentre as vantagens de realizar planejamento fatorial, destacam-se: otimizar procedimentos de experimentações e aferições; extrações de informações de dados químicos; tratamento adequado de informação multivariada; tomada de decisão objetiva; análise da qualidade de determinação analítica (ORTIZ, 2007).

Como mencionado anteriormente, na construção de um planejamento fatorial é preciso determinar quais fatores serão experimentados e a quantos níveis serão considerados. Os fatores podem ser quantitativos ou qualitativos, sendo que estes são características importantes, segundo Box *et al.* (1978): precisa-se de relativamente poucos experimentos por fator estudados; a informação produzida pelo planejamento fatorial é utilizada continuamente por senso comum, aritmética elementar e computação gráfica; o procedimento é escalado adequadamente quando for necessário. Assim, a etapa de triagem determina se o planejamento fatorial é completo ou fracionário, isto é, um planejamento fatorial completo é um sistema que é composto por

Figura 2 – Procedimento geral para realização e interpretação correta de um planejamento fatorial.



Fonte: Adaptado, Pereira e Pereira-Filho, 2018.

um número entre 2 e 4 variáveis, no caso de quantidades superiores será planejamento fatorial fracionário. Além disso, o principal objetivo do processo de triagem é identificar quais variáveis são mais importantes a partir do cálculo de efeitos das variáveis e de suas interações, assunto que é descrito no próximo tópico (PEREIRA-FILHO, 2015).

### 3.2.1 Cálculo de efeitos

Uma vez realizado os experimentos, com fatores devidamente selecionados através de uma revisão bibliográfica de um procedimento em questão, será necessário realizar os cálculos dos efeitos para determinar quais variáveis são significativas em relação à resposta estudada. O cálculo de efeito é descrito pela Equação 3.1, onde o efeito de uma variável é a diferença

das médias dos valores no nível alto ( $\bar{y}_+$ ) e baixo ( $\bar{y}_-$ ).

$$Efeitos = \bar{y}_+ - \bar{y}_- \quad (3.1)$$

Os efeitos também possuem uma classificação de ordem de acordo com a interação dos efeitos, por exemplo em um planejamento fatorial envolvendo quatro variáveis ( $2^4$ ) (PEREIRA-FILHO, 2015):

- O efeito de primeira ordem é o efeito para as variáveis de forma isoladas (1,2,3,4);
- Os efeitos de segunda ordem são calculadas para interações entre duas variáveis (12, 13, 14, 23, 24, 34)
- Os efeitos de terceira ordem são calculadas para interações entre três variáveis (123,124,134, 234);
- E por fim, o efeito de quarta ordem é calculada para interações entre quatro variáveis (1234).

Com isso, os efeitos de ordem superiores a 3 demonstram menor relevância em comparação com efeitos de primeira ou segunda ordem (PEREIRA-FILHO; PEREIRA; LEMOS, 2009).

### 3.2.2 Identificação de efeitos importantes

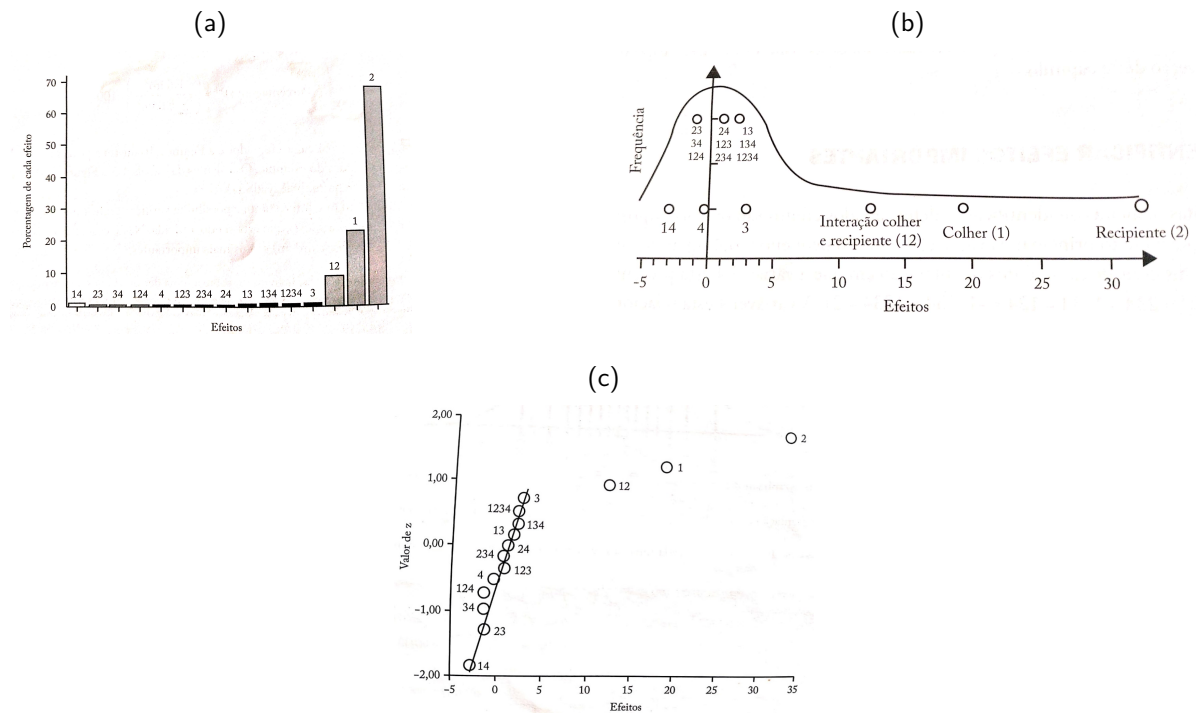
A identificação de efeitos importantes pode ser realizada de diferentes formas, no entanto a forma proposta na resolução dos exemplos encontrados na discussão deste documento foi realizada através da porcentagem dos quadrados de cada efeito, descrito pela Equação 3.2. Isto é, a relação entre o efeito de cada interação sobre a soma entre os quadrados dos efeitos (ZAVISKA et al., 2011). Com isso, os efeitos que possuírem maiores porcentagens terão efeitos significativos, sendo estes positivos ou negativos em relação a resposta.

$$Porcentagem\ de\ um\ efeito = \frac{Efeito_i}{\sum (Efeitos^2)} \quad (3.2)$$

A influência de cada efeito pode ser representada graficamente. Como utilizado no exemplo envolvendo o planejamento da influência de quatro variáveis para obtenção de chocolates (M&M's<sup>TM</sup>) que foi publicado na revista *The Chemical Educator* por Pereira-Filho et al. (2009). Dessa forma, a Figura 3 mostra os efeitos significativos em formatos de histograma, barras e dispersão .



Figura 3 – Gráfico de porcentagem relativa aos efeitos calculados (a). Representação dos efeitos calculados em um histograma (b). Gráfico de efeitos em relação à distribuição gaussiana (c).



Fonte: Adaptada, Pereira-Filho, 2015.

### 3.2.3 Réplicas do ponto central

As réplicas de ponto central são experimentos que estão em uma condição de valor codificado igual a zero. Estes valores possuem oscilações devido aos erros de precisão do método empregado nos ensaios e tem como função calcular as medidas de dispersão, sobretudo a variância experimental, indicada pela Equação 3.3. Deste modo, com o valor de variância, calcula-se o erro experimental (Equação 3.4) que se trata do desvio padrão do erro experimental; posteriormente, o valor do erro de um efeito é obtido com o número de réplicas realizadas ( $n$ ) e número de variáveis envolvidas ( $k$ ), indicada pela Equação 3.5. Com estes valores, o intervalo de confiança (I.C.) é definido com o uso do valor de  $t$  onde é empregado 95% de confiança por convenção, indicado pela Equação 3.6, de modo que pode ser visualizado através do gráfico de probabilidade indicado pela Figura 3c, o que facilita a observação dos efeitos significativos ao planejamento para valores que não estão dentro o intervalo de confiança.

$$\text{Variância Experimental} = \frac{\sum (x_i - \bar{x})^2}{n - 1} \quad (3.3)$$

$$\text{Erro Experimental} = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n - 1}} \quad (3.4)$$

$$\text{Erro de um Efeito} = \frac{2 \times \text{Erro Experimental}}{\sqrt{n \times 2^k}} \quad (3.5)$$

$$I.C. = \text{erro de um efeito} \times t \quad (3.6)$$

No tópico sobre “Modelos de Regressão”, será discutido a criação de modelos estatísticos onde será avaliado se o mesmo possui ou não falta de ajuste através da análise de variância (ANOVA), conforme a etapa 3 e 4, indicado pela Figura 2 anteriormente. Assim sendo, as réplicas do ponto central são determinantes, pois através da soma da variância experimental é possível calcular a soma quadrática do erro puro (SQep) (Equação 3.7) e os graus de liberdade (Equação 3.8).

$$SQep = \sum_{i=1}^m \sum_{j=1}^n (y_{ij} - \bar{y}_i)^2 \quad (3.7)$$

$$g.l. = n^{\circ} \text{ Exp. réplica central} - 1 \quad (3.8)$$

Vale salientar que o uso destas equações será discutido com mais detalhes juntamente com a construção da tabela ANOVA e interpretação dos dados.

### 3.2.4 Padrões de confundimentos

Em planejamentos fatoriais que envolvem a presença de cinco ou mais variáveis, torna-se inviável o planejamento fatorial completo pois o número de experimentos seria em grande quantidade, fazendo que o processo seja inviável e custoso. Assim, o planejamento fatorial fracionário é uma boa opção quando há a possibilidade de considerar uma variável que interaja com todas as outras, sendo esta nomeada de geratriz, que se trata da multiplicação de todas condições das variáveis. Dessa maneira, houve a convenção de nomear os efeitos em contrastes quando existem efeitos confundidos em um sistema (NETO; SCARMINIO; BRUNS, 1996).

Para esclarecer com mais detalhes sobre padrão de confundimento, será utilizado o exemplo que aborda o uso do planejamento fatorial fracionário no artigo da revista *Química Nova*: “Aplicação de programa computacional livre em planejamento de experimentos: um tutorial”, de Pereira e Pereira-Filho (2018). Trata-se do uso do planejamento fatorial fracionário em um experimento onde os autores tem como objetivo minimizar o diâmetro (resposta 1) e maximizar a distribuição (resposta 2) de partículas em nanoestruturas de carbono na matriz polimérica de poli (estireno-co-butil acrilato) (MORAIS et al., 2017). Com isso, o planejamento envolve 4 variáveis, estas são: massa de polietilenoglicol, volume da fase aquosa de ácido nítrico, porcentagem de álcool polivinílico e volume da fase aquosa. Isto é, refere-se a um planejamento fatorial fracionário  $2^{4-1}$ , pois o fator geratriz 1234 é interação do volume com todas as outras variáveis, logo, as variáveis estão confundidas com a quarta variável e também as variáveis isoladas estão confundidas com as variáveis terciárias, assim como está confundindo entre si as variáveis secundárias (NETO; SCARMINIO; BRUNS, 2010).

### 3.3 Modelo de regressão

O segundo passo para realização do planejamento fatorial é o processo de ajuste fino das variáveis importantes para construir um modelo empírico que relaciona as variáveis analisadas em relação às respostas. Geralmente, nesta etapa o número de variáveis é pequena, sendo um número de 2 ou 3 (PEREIRA-FILHO, 2015). A classe “Regression2” da biblioteca “pde.py” gera o modelo quadrático a partir dos dados fornecidos pela matriz  $\mathbf{X}$ , vetor  $y$ , SQep e graus de liberdade das réplicas do ponto central. Em outras palavras, a rotina programacional gera uma equação que descreve um polinômio de ordem de acordo com número de variáveis envolvidas no planejamento fatorial. De modo geral, a Equação 3.9 descreve o modelo que é criado com uso do método “regression2” no Python:

$$\hat{y} = \beta_0 + \sum_{i=1}^k \beta_i x_i + \sum_{i=1}^k \beta_{ii} x_i^2 + \sum_{i=1}^k \beta_{ij} x_i x_j + \varepsilon \quad (3.9)$$

Observe que  $k$  é o número de variáveis envolvidas, os valores  $\beta_i$ ,  $\beta_{ii}$  e  $\beta_{ij}$  tratam-se dos coeficientes de regressão para cada efeito e, por último,  $\varepsilon$  são os resíduos. Uma vez calculado o modelo é preciso verificar se o mesmo possui falta de ajuste por meio da análise de variância (ANOVA), conteúdo que será abordado no próximo tópico (NOVAES et al., 2017).

#### 3.3.1 Análise de variância (ANOVA)

Esta etapa do planejamento fatorial é importante para verificar se o modelo matemático ajustado segue a descrição dos valores experimentais. O método ANOVA compara a variação das variáveis provocada pela combinação dos níveis das mesmas em relação aos erros aleatórios pelas respostas obtidas (BRUNS; SCARMINIO; NETO, 2006). Com isso, a análise de variância é dividida, basicamente, em somas quadráticas, graus de liberdades, média quadrática e teste F, geralmente sendo representada em formato de uma tabela, tal como indicado pela Tabela 1 produzida pela classe “Regression2” no Python.

Em primeiro lugar, a Tabela 1 é composta por 5 somas quadráticas sendo que cada valor existe o seu respectivo grau de liberdade, sendo estas: resíduos (SQres), somatório dos quadrados da diferença entre valores experimentais e previstos pelo modelo e grau de liberdade sendo o  $n^o$  experimentos -  $n^o$  coeficiente (n-p); regressão (SQR), somatório dos quadrados da diferença entre valores experimentais e sua respectiva média com grau de liberdade igual ao  $n^o$  coeficiente - 1 (p-1); total (SQT), trata-se da soma entre SQR e SQres e grau de liberdade igual ao  $n^o$  experimentos - 1 (n-1); erro puro (SQep), somatório dos quadrados da diferença entre valores experimentais das réplicas do ponto central e sua respectiva média com grau de liberdade sendo a diferença entre o número de experimentos totais e experimentos independentes (n-m); e por último, falta de ajuste (SQfaj), a diferença entre SQres e SQep e grau de liberdade equivalente a diferença entre o número de experimentos e coeficientes do modelo (m-p).

Com os valores de média quadrática é possível calcular a médias quadráticas dos resíduos (MQres), dado por SQ regressão (MQR), falta de ajuste (MQfaj) e erro puro (MQep). O cálculo das respectivas médias quadráticas é dado pela divisão da soma quadrática e graus de liberdade. Os testes F1 e F2 são calculados a partir das médias quadráticas. Assim, o teste F1 trata-se da razão  $\frac{MQR}{MQres}$  e com este valor é avaliado o modelo em relação ao valor de F tabelado para 95% de confiança, consequentemente, o modelo não terá falta de ajuste se a resultante desta relação for superior à 10. Do mesmo modo, a relação é aplicada pelo F tabelado para 95% de confiança para o teste F2 ( $\frac{MQfaj}{MQep}$ ), de forma que será assumido não falta de ajuste para valores inferiores à 1 (BOX et al., 1978).

Tabela 1 – Modelo de tabela ANOVA gerada pela rotina progracional "pde.py".

Parâmetro	Soma Quadrática	Graus de liberdade	Média Quadrática	
Regressão:	$\sum_{i=1}^n (\hat{y}_i - \bar{y})^2$	p-1	$\frac{SQR}{p-1}$	Teste F1
Resíduo:	$\sum_{i=1}^n (y_i - \hat{y}_i)^2$	n-p	$\frac{SQRes}{n-p}$	$\frac{MQR}{MQRes}$
Total:	$\sum_{i=1}^n (y_i - \bar{y})^2$	n-1	$\frac{SQT}{n-1}$	Teste F2
Erro Puro:	$\sum_{i=1}^m \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2$	n-m	$\frac{SQep}{n-m}$	$\frac{MQfaj}{MQep}$
Falta de Ajuste:	$\sum_{i=1}^m \sum_{j=1}^{n_i} (\hat{y}_i - \bar{y}_i)^2$	m-p	$\frac{SQfaj}{m-p}$	F Tabelado (95%)
R <sup>2</sup> :	$\frac{SQR}{SQT}$	R:	$\sqrt{\frac{SQR}{SQT}}$	F1:
R <sup>2</sup> máximo:	$\frac{SQT-SQep}{SQT}$	R máximo:	$\sqrt{\frac{SQT-SQep}{SQT}}$	F2:

Fonte: Adaptado, autoria própria, 2022

E por último a Tabela 1 gerada pela rotina programacional do Python fornece os valores de variação explicada e variação explicada máxima. Dado por  $\frac{SQR}{SQT}$ , o coeficiente de variação explicada descreve a quantidade de informação que a SQR explica no que diz respeito à SQT. Enquanto, o coeficiente de variação explicada máxima, é dado por  $(\frac{SQT-SQep}{SQT})$ , no caso avalia a porcentagem máxima de informação do modelo que pode ser explicado (CHUI, 2007).

#### 4 ROTINA PROGRAMACIONAL EM PYTHON

A biblioteca "pde.py" é um algoritmo construído em programação a orientação à objetos, portanto as rotinas adaptadas do programa livre Octave foram divididas em três classes, sendo estas: "Fabi\_efeito", "Regression2" e "Super\_fabi". Vale salientar que as iniciais das classes tem o primeiro caractere maiúsculo devido a convenção adotada pela PEP8 (*Python Enhancement Proposal 8*). Outro aspecto importante a ser comentado é que dentro das classes possuem os métodos, que são as funções que pertencem à classe, deste modo serão estas que

portam a funcionalidade adaptada das rotinas propostas por Pereira e Pereira-Filho (2018). Em suma, os métodos "fabi\_efeito", "regression2" e "super\_fabi", oriundos de suas próprias classes, respectivamente, possuem como funcionalidade: construir gráfico de probabilidade e porcentagem dos efeitos (1) gerar a equação do modelo empírico e verificação de falta de ajuste através da análise de variância (2) plotar gráfico de superfície de resposta e contorno com o valor de resposta máxima e também calcular as derivadas parciais (3). Além disso, a biblioteca possui uma classe auxiliadora nomeada de "CP" (*Central Points*), esta foi criada para gerar valores, como o valor de  $t$  para 95% de confiança ou a Soma Quadrática do Erro Puro (SQep), que são utilizadas como parâmetros nas classes "Fabi\_efeito" e "Regression2".

## 5 EXEMPLO 1: PLANEJAMENTO FATORIAL COMPLETO

### 5.1 Descrição do método: Exemplo 1

Neste exemplo, os autores estudaram a influência de 3 variáveis em um planejamento fatorial completo  $2^3$  com o objetivo de maximizar a intensidade de fluorescência do antimônio. Desta forma, buscaram identificar as variáveis mais importantes por meio de uma etapa de triagem (SANTOS et al., 2018). Assim, a Tabela 2 indica os 11 experimentos com os valores codificados e suas respectivas respostas, no qual 3 experimentos são réplicas do ponto central. Portanto, é possível calcular 7 efeitos, sendo estes: 3 efeitos principais; 3 secundários, interação entre duas variáveis; 1 terciário, interação mútua entre as 3 variáveis. Tal procedimento é descrito com o uso do Microsoft Excel no vídeo48 play24 (vide Apêndice A) além do processo de importação de dados ao editor de código no *Jupyter Notebook*. Desta forma, foram criadas as duas variáveis: matriz  $\mathbf{X}$  (matriz contendo valores codificados dos efeitos) e vetor  $y$  (valores da resposta em questão).

Então, os cálculos do erro de um efeito, descrito pela Equação 3.5, e valor de  $t$  para 95% de confiança podem ser obtidos utilizando a classe "CP", bem como descrito no vídeo48 play24 até o vídeo50 play24 (vide Apêndice A). Tendo em vista os comandos instruídos no vídeo, obtém-se 0,46 e 4,30 para o erro de um efeito e valor de  $t$ , respectivamente. Com isso, para plotar os gráficos de Probabilidade (Figura 4a) e Porcentagem em relação aos efeitos (Figura 4b) é utilizado o método "fabi\_efeito" que não possui nenhum parâmetro, todavia, na entrada de dados são fornecidas os atributos da classe, ou seja, os valores da matriz  $\mathbf{X}$ , vetor  $y$ , erro de um efeito e valor de  $t$  que devem ser inseridos no momento de instância da classe, tal como descrito no vídeo51 play24 (vide Apêndice A).

Tabela 2 – Planejamento fatorial completo  $2^3$  para verificar a influência de 3 variáveis (v1 a v3) nos sinais de fluorescência de Sb.

Exp.	HCl (v1)		NaBH <sub>4</sub>		Tempo (v3)		Intensidade de fluorescência do Sb
	Cod.	mol/L	Cod.	%, m/v	Cod.	min	
1	-1	3	-1	1	-1	10	178,4
2	-1	3	-1	1	1	30	167,5
3	-1	3	1	3	-1	10	225,7
4	-1	3	1	3	1	30	218,1
5	1	5	-1	1	-1	10	86,6
6	1	5	-1	1	1	30	91,0
7	1	5	1	3	-1	10	195,6
8	1	5	1	3	1	30	189,2
9	0	4	0	2	0	20	137,5
10	0	4	0	2	0	20	135,7
11	0	4	0	2	0	20	137,8

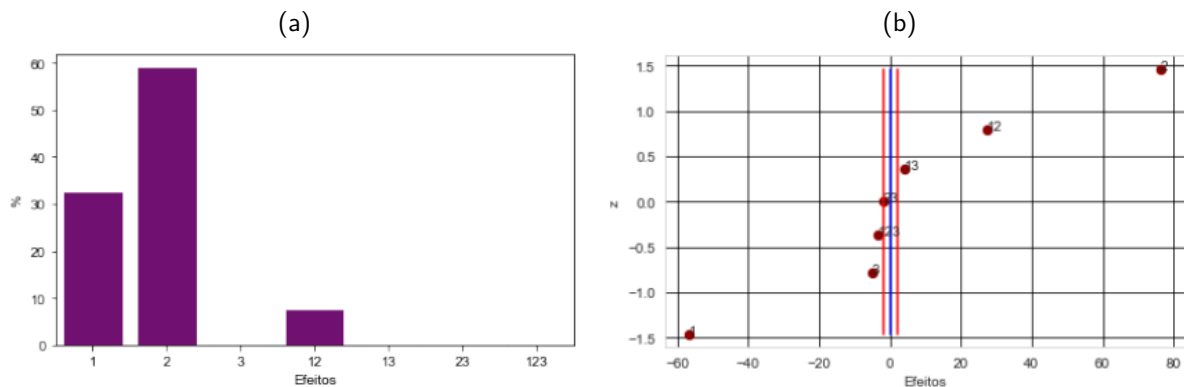
Fonte: Adaptado, Pereira e Pereira-Filho, 2018.

## 5.2 Resultados e discussão: Exemplo 1

A Figura 4a apresenta três efeitos que possuem maior importância ao experimento. Estes efeitos incluem dois principais, variáveis 1 e 2, e um de interação, efeito 12. O gráfico de probabilidade representado pela Figura 4b indica que as variáveis mencionadas estão fora do intervalo de confiança, e, portanto, indicam a mesma conclusão do gráfico de porcentagem em relação aos efeitos. Então, para finalizar a triagem das variáveis, precisa-se plotar novamente os gráficos do método “fabi\_efeito” somente com as variáveis importantes ao planejamento fatorial. Deste modo, é produzido réplicas ao excluir a variável 3 dos cálculos, sendo necessário a reorganização da matriz **X** e vetor **y**, tal procedimento é mostrado no vídeo52 play24 (vide Apêndice A).

Conclui-se que para maximizar os valores dos sinais de fluorescência é preciso, idealmente, reduzir e aumentar as condições de trabalho para as variáveis 1 e 2, respectivamente. De modo que, os efeitos da variável 3 são insignificante em relação à resposta, isto é, dentro da faixa analisada de 10 a 30 minutos, sendo possível trabalhar em qualquer valor constante dentro deste intervalo, segundo Pereira e Pereira-Filho (2018).

Figura 4 – Imagens geradas pela rotina programacional no Python. Gráfico de porcentagem relativa aos efeitos calculados ao exemplo 1 (a). Gráfico de probabilidade aplicado ao exemplo 1(b).



Fonte: De autoria própria, 2022.

## 6 EXEMPLO 2: PLANEJAMENTO FATORIAL FRACIONÁRIO

### 6.1 Descrição do método: Exemplo 2

O segundo exemplo se trata de um estudo do comportamento da interação de nanocompósitos de látex acrílico e carbono para observar a multifuncionalidade de propriedades físico-química, como, por exemplo, elétrica, mecânica, magnética e química (MORAIS et al., 2017). Dentro deste contexto, o objetivo dos autores foi minimizar o tamanho das partículas (resposta 1) e maximizar a distribuição das mesmas (resposta 2). Assim, será aplicado novamente a classe "Fabi\_efeito" em cálculo de contrastes em um planejamento fatorial fracionário para cada resposta.

Semelhante ao exemplo discutido anteriormente, a Tabela 3 indica os valores codificados juntamente com as 3 réplicas do ponto central e respectivas respostas. O planejamento fatorial fracionário é do tipo  $2^{4-1}$  devido a presença do fator geratriz 1234 para a variável 4. Outro fator importante é a presença de contrastes no lugar dos efeitos, logo, os efeitos principais são confundidos com os terciários e os efeitos secundários são confundidos entre si (NETO; SCARMINIO; BRUNS, 2010). Isto é, há 7 contrastes: 1+234, 2+134, 3+124, 4+123, 12+34, 13+24, 14+23. Os cálculos dos contrastes são descritos no vídeo53 play24 (vide Apêndice A).

Também neste exemplo o valor de erro de um efeito e valor de  $t$  são obtidos através das réplicas do ponto central pela classe "CP", assim, tem-se os valores de 0,62 e 4,30, respectivamente. Destacam-se que estes valores podem ser substituídos por zero no comando do método, com isso, os gráficos serão gerados, no entanto sem a construção do intervalo de

Tabela 3 – Planejamento fatorial fracionário  $2^{4-1}$  para analisar a influência de 4 variáveis (v1 a v4) em duas respostas

Exp	Poielileno glicol (PEG) (v1)		Fase Aquosa (v2)		Álcool polivinílico (v3)		Fase Aquosa (v4=1234)		Respostas monitoradas	
	Cod.	mg	Cod.	mL	Cod.	%	Cod.	mL	Diâmetro	Distribuição
1	1	200	-1	1	-1	0,5	1	100	28,4	2,02
2	-1	50	-1	1	1	2	1	100	26,0	1,29
3	1	200	1	3	1	2	1	100	14,1	1,62
4	-1	50	1	3	-1	0,5	1	100	8,1	2,51
5	-1	50	1	3	1	2	-1	30	13,0	4,76
6	-1	50	-1	1	-1	0,5	-1	30	14,7	1,79
7	1	200	-1	1	1	2	-1	30	7,8	3,23
8	1	200	1	3	-1	0,5	-1	30	6,7	3,61
9	0	125	0	2	0	1,25	0	65	20,8	1,56
10	0	125	0	2	0	1,25	0	65	18,6	1,35
11	0	125	0	2	0	1,25	0	65	22,9	1,76

Fonte: Adaptado, Pereira e Pereira-Filho, 2018.

confiança. Uma vez carregados os dados experimentais em variáveis, os gráficos de porcentagem *versus* efeitos e probabilidades são gerados após o uso do comando abaixo:

```
1 import pde
   pde.Fabi_efeito(X,y,0.62,4.3).fabi_efeito()
```

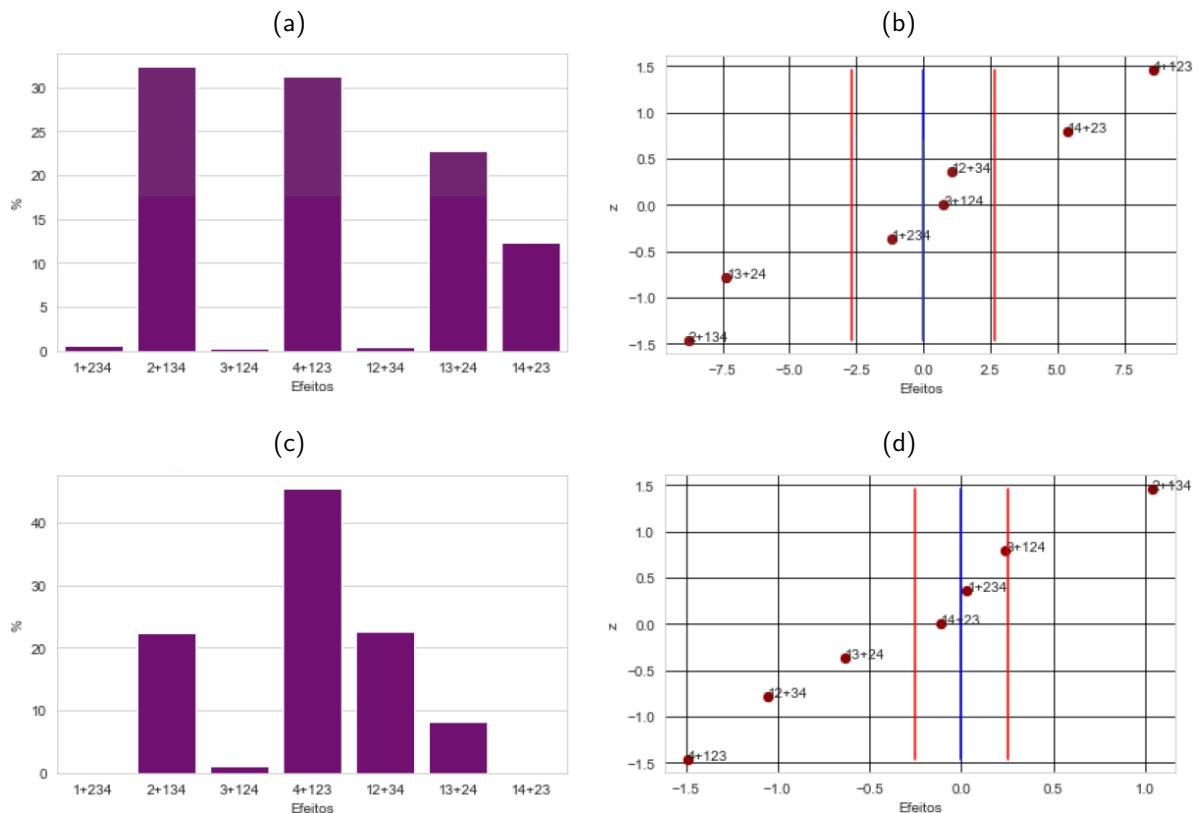
## 6.2 Resultados e discussão: Exemplo 2

Observa-se, nas Figuras 5a e 5b, que os contrastes significativos para o planejamento são os 2+134, 4+123, 13+24 e 14+23 para a primeira resposta (diâmetro das partículas). Dessa maneira, para reduzir o tamanho das partículas, precisa-se operar com o nível baixo e alto com as variáveis 2 e 1, respectivamente.

Enquanto a Figura 5c e 5d são os resultados para a segunda resposta (distribuição das partículas), dessa forma os resultados obtidos indicam que os contrastes mais importantes são 4+123, 12+34, 2+134 e 13+24. Então, para maximizar a distribuição das partículas, deve-se operar em níveis opostos para as variáveis 1 e 2 e também para as variáveis 1 e 3, sendo que as variáveis 2 e 3 precisam estar em nível alto enquanto as variáveis 1 e 4 em nível baixo. (PEREIRA; PEREIRA-FILHO, 2018).



Figura 5 – Imagens geradas pela rotina programacional no Python. Gráfico de porcentagem relativa aos contrastes calculados ao exemplo 2 para resposta 1 (a) e 2 (c). Gráfico de probabilidade aplicado ao exemplo 2 para resposta 1 (b) e 2 (d).



Fonte: De autoria própria, 2022.

## 7 EXEMPLO 3: PLANEJAMENTO FATORIAL DOEHLERT

### 7.1 Descrição do Método: Exemplo 3

Este exemplo é a continuidade do exemplo 1 envolvendo o planejamento fatorial completo. Anteriormente, com o auxílio da classe “Fabi\_efeito” foi concluído que a terceira variável poderia ser desconsiderada. Desse modo, o planejamento em questão envolve agora duas variáveis: concentrações de HCl e de NaBH<sub>4</sub>. Desta forma, foi proposto um planejamento fatorial do tipo *Doerlert* com o objetivo de identificar as melhores condições de trabalho para obter os maiores sinais de fluorescência de Sb (PEREIRA; PEREIRA-FILHO, 2018). Este tipo de planejamento tem como objetivo testar as variáveis em quantidades diferentes de níveis, isto é, quando as variáveis não possuem importâncias equivalentes (SANTOS et al., 2018). Assim, neste caso, observando a Figura 4 é possível concluir que a variável 2 possui um efeito maior que a variável 1. Tendo em vista isso, os autores configuraram o planejamento Doehlert

em questão para 5 níveis na variável 2 (-1, -0,5, 0, 0,5, 1) e 3 níveis na variável 1 (-0,866, 0, 0,866). A Tabela 4 indica os valores codificados do experimento efetuado bem como as suas respectivas respostas.

Tabela 4 – Planejamento fatorial Doehlert para analisar a influência de 2 variáveis no rendimento na intensidade de fluorescência de antimônio.

Exp	HCl (v1)		NaBH <sub>4</sub> (v2)		Intensidade de fluorescência de Sb
	Cod.	mol/L	Cod.	%, m/v	
1	0,866	5	0,5	1,8	367
2	0,866	5	-0,5	1,4	660
3	0	4	-1	1,2	762
4	-0,866	3	-0,5	1,4	787
5	-0,866	3	0,5	1,8	434
6	0	4	1	2,0	167
7	0	4	0	1,6	651
8	0	4	0	1,6	643
9	0	4	0	1,6	652

Fonte: Adaptado, Pereira e Pereira-Filho, 2018.

A primeira etapa para construção do modelo de regressão está no cálculo dos coeficientes do modelo. Esta etapa também é realizada com o uso Microsoft Excel onde é determinado 6 coeficientes que descrevem a Equação 7.1, no qual: o coeficiente constante é  $b_0$ , lineares são  $b_1$  e  $b_2$ , quadráticos  $b_{11}$  e  $b_{22}$  e interação  $b_{12}$ . Desse modo, a matriz  $\mathbf{X}$  conterá os valores codificados de 9 linhas contendo os experimentos e 6 colunas com os coeficientes, tal procedimento apresentado no vídeo53 play24 (vide Apêndice A).

$$y = b_0 + b_1x_1 + b_2x_2 + b_{11}x_1^2 + b_{22}x_2^2 + b_{12}x_1x_2 \quad (7.1)$$

Após a importação da matriz  $\mathbf{X}$  e do vetor  $y$  para o editor de código, precisa-se calcular a soma quadrática do erro puro (SQep), indicado pela Tabela 1 na seção 3.3.1. Através dos valores das réplicas do ponto central é possível calcular o valor de SQep e também o número de graus de liberdade ( $n^\circ$  experimentos –  $n^\circ$  experimentos independentes, ou seja,  $n - m$ ). Embora seja possível o cálculo de SQep por meio da etapa anterior com uso do Microsoft Excel, a classe “CP” carrega o método “SQPE” que propriamente determina este resultado, conforme o trecho de código abaixo:

```
import pde
pde.CP([651, 643, 652]).SQPE()
```

Note que o método “fabi\_efeito” possui a entrada de dados no momento de instância da classe. Dessa forma, para o exemplo 3, tem-se os valores de 48,7 e 2 para SQep e graus de liberdade, respectivamente. Com a matriz  $\mathbf{X}$ , vetor  $y$ , SQep e graus de liberdade é possível dar seguimento no estabelecimento do modelo de regressão utilizando a classe “Regression2”, que por sua vez carrega o método “regression2”. Esta rotina tem como objetivo calcular uma equação do modelo empírico e avaliar se há falta de ajuste através da análise de variância (ANOVA). Abaixo segue o comando descrito no editor de código:

```
import pde
pde.Regression2(X,y,SSPE,2,self_check=True,auto=False).regression2()
```

Observe que há dois atributos opcionais na classe “Regression2”. O primeiro, “self\_check”, trata-se de um comando que verifica automaticamente se o modelo possui ou não falta de ajuste com a relação matemática dos Testes F1 e F2 da ANOVA. Enquanto o segundo atributo, “auto”, trata-se de um comando de recálculo do modelo automaticamente, ou seja, é retirado os coeficientes insignificantes e os gráficos são gerados novamente para a análise. Com os coeficientes da equação do modelo de regressão é possível determinar as condições ótimas de experimentação para obtenção de altos sinais fluorescência de antimônio. Nesta determinação são empregadas derivadas parciais ou a visualização dos gráficos de superfície de resposta e contorno, uma vez que se trata de um modelo envolvendo duas variáveis. Assim, a Figura 8 indica os gráficos de superfície de resposta e contorno gerados pelo método “superficie” da classe “Super\_fabi”, indicado pelo trecho de código abaixo:

```
import pde
pde.Super_fabi(coefs, 5, 3, 2, 1.2, 0.866, -0.866, 1, -1).superficie()
```

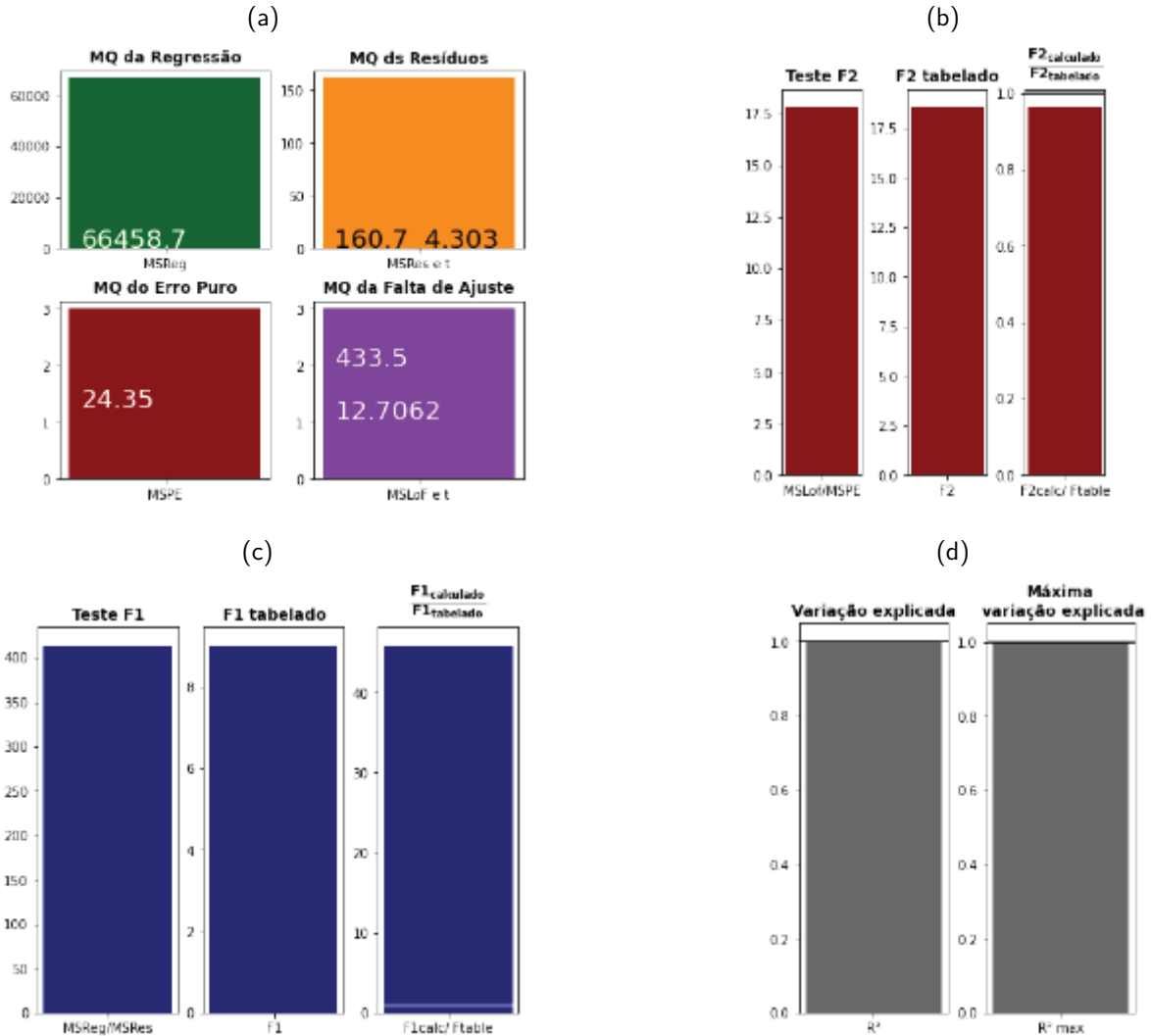
Os atributos presentes no trecho do código são: as listas dos coeficientes válidos da equação do modelo; valores reais máximos e mínimos para a variável 1 (5,3) e 2 (2, 1,2; e codificados máximo e mínimos para variável 1 (0,866, -0,866) e 2 (1, -1). Então, após inserir o comando mencionado é gerado uma imagem com dois gráficos contendo a equação do modelo juntamente com as coordenadas que representam as condições ideais de experimentação e o respectivo valor máximo previsto para a resposta.

## 7.2 Resultados e Discussão: Exemplo 3

A Figura 6 mostra quatro gráficos, estes são: a média quadrática da regressão (MSreg, coluna verde); resíduo (MSres, coluna amarela); erro puro (MSPE, vermelho) e falta de ajuste (MSLoF, lilás). Ainda na Figura 6a são apresentados os valores de  $t$  para 95% de confiança para

as médias quadráticas dos resíduos e falta de ajuste. Os valores de  $t$  para 95% de confiança são escolhidos através da avaliação dos gráficos apresentados pela Figura 6b e 6c.

Figura 6 – Imagens geradas pela rotina programacional no Python. Gráficos das médias quadráticas (a), teste F1 para MQR e MQR (b) teste F2 para MQfaj e MQep (c) e máxima variação explicada  $R^2$  e explicável  $R_m^2$  máximo aplicado ao exemplo 3 (d).



Fonte: De autoria própria, 2022.

A Figura 6b apresenta três colunas: Teste F2 ( $\frac{MSLoF}{MSPE}$ ), F2Tabelado para 95% de confiança e relação entre F2 calculado e F2 tabelado. Idealmente, o modelo não indica falta de ajuste quando a razão dos valores de F2 calculado e tabelado são inferiores a 1. Dessa forma, o modelo não apresenta falta de ajuste, já que o F2 calculado é da ordem de 17 e o tabelado é próximo de 19, logo, a razão  $\frac{17}{19} < 1$ . Enquanto a Figura 6c indica o Teste F1, semelhante ao Teste F2, há três colunas sendo a primeira o Teste F1 ( $\frac{MSR}{MSRes}$ ), a segundo o F2 tabelado para 95% de confiança e a relação de  $\frac{F2calculado}{F2tabelado}$ . Assim, o modelo apresenta uma boa regressão, pois a razão dos valores de F1 calculado e tabelado são superiores a 10. Então, conclui-se

que o modelo reafirma a não falta de ajuste, uma vez que o  $F_{\text{calculado}}$  é da ordem de 400 e  $F_{\text{tabelado}}$  é próximo de 9, logo, a razão  $\frac{400}{9} > 10$ . E por fim, a Figura 6d indica os valores de variação explicada ( $R^2$ ) e máxima variação explicada ( $R^2_{\text{max}}$ ), no qual apontam que o modelo é ajustado já que seus valores estão próximos à 1. (PEREIRA-FILHO, 2015).

Tendo em vista a interpretação dos gráficos gerados, pode-se afirmar que o modelo não apresenta indícios de falta de ajuste. Com isso, para a segunda etapa do método da "regression2", será escolhido a média quadrática dos resíduos para os cálculos, tal cálculo é realizado automaticamente após a escolha pelo usuário na seguinte mensagem no terminal do editor do código:

O modelo possui falta de ajuste? [S/N]

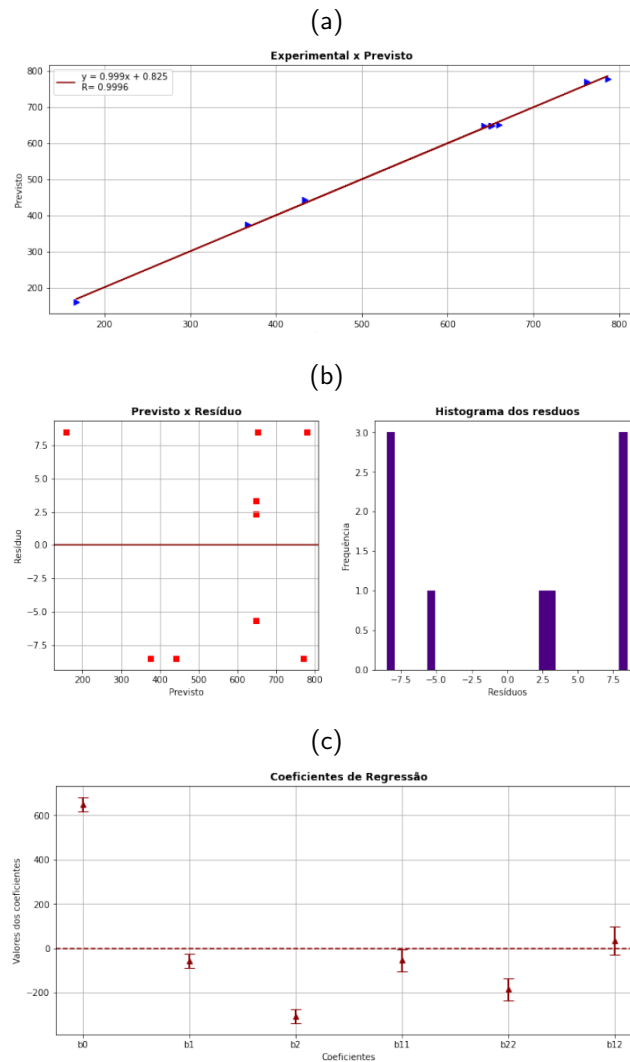
A Figura 7 mostra os gráficos gerados pela "regression2" após inserir a informação se há ou não falta de ajuste. O primeiro gráfico, indicado pela Figura 7a, são os valores previstos pelo modelo em relação aos dados experimentais. Tal gráfico tem como objetivo visualizar a discrepância dos valores do modelo que compreendem os valores experimentais. A segunda imagem, indicada pela Figura 7b, são dois gráficos: previsto *versus* resíduo e o histograma dos resíduos, ambos gráficos visam analisar a homocedasticidade do modelo de forma que procura-se, idealmente, uma maior distribuição aleatória no gráfico à esquerda e valores concentrados próximos a zero para o gráfico à direita. E por último, o terceiro gráfico (Figura 7c) que são os coeficientes da equação do modelo empírico e seus respectivos intervalos de confiança. Este tem como objetivo determinar quais coeficientes podem ser desconsiderados na equação. Por conseguinte, a Figura 7c aponta que o coeficiente  $b_{12}$  é insignificante uma vez que este tem o seu intervalo de confiança que contém o valor nulo, portanto, será necessário recalcular o modelo assumindo somente os coeficientes que são significativos ao modelo (SANTOS et al., 2018).

A Equação 7.2 se refere ao modelo e seus respectivos valores de variância para 95% confiança. Esta é gerada após o recálculo do modelo com o coeficiente  $b_{12}$  excluído nos dados. Tal procedimento foi realizado no *video58 play24* (vide Apêndice A). Dessa maneira, concluiu-se com os gráficos gerados possuem uma leve falta de ajuste com a reprovação do Teste F2 com um valor da razão F2 calculado e tabelado próximo de 1,40 e com uma aprovação do Teste F1 com uma relação próxima de 40. Deste modo, manteve-se o cálculo com a média quadrática dos resíduos, assumindo a não falta de ajuste para obtenção da Equação 7.2

$$Resposta = 648 \pm 34 - 56v_1 \pm 34,2 - 306v_2 \pm 34 - 54,2v_1^2 \pm 54 - 184v_2^2 \pm 54 \quad (7.2)$$

Dessa forma, o método "superfície" aponta que o valor máximo de fluorescência de antimônio

Figura 7 – Imagens geradas pela rotina programacional no Python. Gráficos valores experimentais *versus* previstos (a), previsto *versus* resíduos e histograma de resíduos (b) teste F2 para MQfaj e MQep (c) e coeficientes de regressão.



**Fonte:** De autoria própria, 2022.

é de 790 quando aplicado 3,4 mol/L de HCl (v1) e 1,3% m/v de NaBH<sub>4</sub> (v2). Note que na Figura 8 há uma estrela vermelha no gráfico de contorno indicando estas condições de trabalho em valores codificados. Em relação aos resultados obtidos pelo autores, foi encontrado valores próximos ao modelo, sendo estes: 3,07 mol/L de HCl e 1,24% m/v de NaBH<sub>4</sub>.

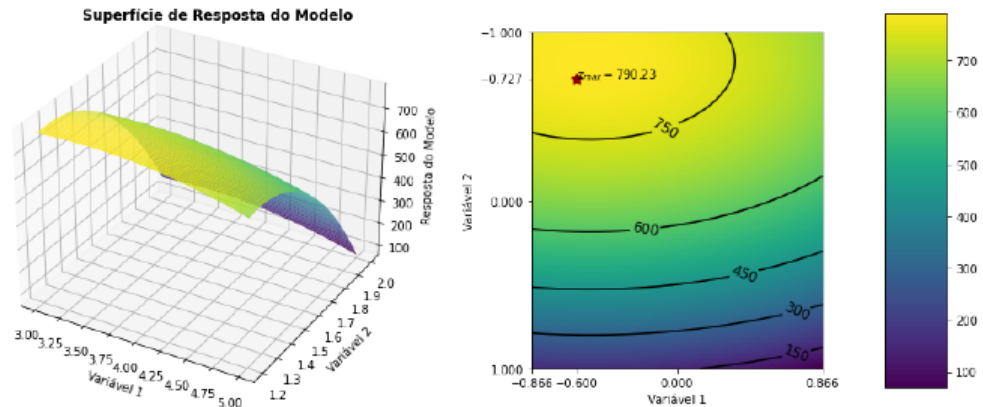
Com os resultados dos 29 experimentos apresentados na Tabela 5, é possível calcular os coeficientes do modelo empírico, que neste caso, tratam-se de 15 coeficientes: constante ( $b_0$ ); 4 lineares ( $b_1$ ,  $b_2$ ,  $b_3$  e  $b_4$ ); 4 quadráticos ( $b_{11}$ ,  $b_{22}$ ,  $b_{33}$  e  $b_{44}$ ) e 6 de interação secundária ( $b_{12}$ ,  $b_{13}$ ,  $b_{14}$ ,  $b_{23}$ ,  $b_{24}$  e  $b_{34}$ ). Enquanto os ensaios 25 a 29 são as réplicas do ponto central que permite calcular a soma quadrática do erro puro.

Figura 8 – Imagens geradas pela rotina programacional no Python. Gráficos de superfície de resposta e contorno para o modelo gerado pelo exemplo 3

$$\text{Resposta} = 648.66667 + -56.00462v_1 + -306.0v_2 + -54.16984v_1^2 + -184.16667v_2^2 + 0v_1v_2$$

$$R_{\text{max}}(-0.60, -0.73) = 790.2$$

$$v_1^{\text{max}} = 3.4 \text{ e } v_2^{\text{max}} = 1.3$$



Fonte: De autoria própria, 2022.

No Apêndice A, do *video61 play24* ao *video63 play24*, mostra todo o procedimento realizado no Python para gerar os gráficos pela “Regression2” que são importantes para estabelecer a equação do modelo. Tal procedimento é semelhante ao realizado no Exemplo 3. Assim, conclui-se com os gráficos que o modelo não possui falta de ajuste, gerando a Equação [7.3](#):

$$\begin{aligned} \text{Resposta} = & 92,28 \pm 1,14 + 6,17v_1 \pm 0,73 + 2,14v_2 \pm 0,73 + 2,79v_3 \pm 0,73 \\ & + 1,99v_4 \pm 0,73 - 5,70v_1^2 \pm 1,00 - 3,41v_2^2 \pm 1,00 - 2,00v_3^2 \pm 1,00 \\ & - 2,58v_4^2 \pm 1,00 - 1,32v_{12} \pm 1,27 - 3,15v_{13} \pm 1,27 + 0,19v_{14} \pm 1,27 \\ & - 2,34v_{23} \pm 1,27 - 2,49v_{24} \pm 1,27 + 0,22v_{34} \pm 1,27 \quad (7.3) \end{aligned}$$

Após o modelo gerado e carregado em uma variável tipo lista com os valores dos coeficientes da equação, é possível calcular as derivadas parciais e, por sua vez, obter as condições ideais de trabalho seguindo o comando abaixo:

```
1 import pde
   pde.Super_fabi(coefs).solver_diff(k=4, printf=False)
```

Note que há dois parâmetros para o método “solver\_diff”. O primeiro é o fator  $k$ , que é o número de variáveis envolvidos no experimento, a sua inserção é obrigatória para o funcionamento do código. Já o segundo é opcional, quando em *True* é impresso no terminal uma equação tipo  $f'(x, y, z, w) = \max$ , já quando em *False* uma tabela convencional para a visualização dos resultados.

Tabela 5 – Planejamento fatorial Box-Behnken para analisar a influência de 4 variáveis no rendimento na produção do benzaldeído.

Exp	Catalisador (v1)		H <sub>2</sub> O <sub>2</sub> em álcool benzílico (v2)		Tempo de reação (v3)		Volume de água (v4=1234)		Respostas monitorada
	Cod.	g	Cod.	%	Cod.	h	Cod.	mL	Rendimento (%)
1	-1	0,7	-1	50	0	4	0	20	73,00
2	1	0,9	-1	50	0	4	0	20	88,15
3	-1	0,7	1	75	0	4	0	20	80,98
4	1	0,9	1	75	0	4	0	20	90,82
5	0	0,8	0	66	-1	3	-1	15	84,55
6	0	0,8	0	66	1	5	-1	15	88,23
7	0	0,8	0	66	-1	3	1	25	86,86
8	0	0,8	0	66	1	5	1	25	91,41
9	-1	0,7	0	66	0	4	-1	15	75,38
10	1	0,9	0	66	0	4	-1	15	88,50
11	-1	0,7	0	66	0	4	1	25	80,24
12	1	0,9	0	66	0	4	1	25	94,13
13	0	0,8	-1	50	-1	3	0	20	80,17
14	0	0,8	1	75	-1	3	0	20	88,09
15	0	0,8	-1	50	1	5	0	20	91,47
16	0	0,8	1	75	1	5	0	20	90,03
17	-1	0,7	0	66	-1	3	0	20	72,34
18	1	0,9	0	66	-1	3	0	20	89,63
19	-1	0,7	0	66	1	5	0	20	84,61
20	1	0,9	0	66	1	5	0	20	89,31
21	0	0,8	-1	50	0	4	-1	15	79,08
22	0	0,8	1	75	0	4	-1	15	88,30
23	0	0,8	-1	50	0	4	1	25	88,02
24	0	0,8	1	75	0	4	1	25	87,29
25	0	0,8	0	66	0	4	0	20	91,61
26	0	0,8	0	66	0	4	0	20	91,70
27	0	0,8	0	66	0	4	0	20	93,00
28	0	0,8	0	66	0	4	0	20	92,11
29	0	0,8	0	66	0	4	0	20	93,00

Fonte: Adaptado, Pereira e Pereira-Filho, 2018.

### 7.3 Resultados e discussão: Exemplo 4

O primeiro modelo gerado apontou que os coeficientes b<sub>14</sub> (0,19±1,27) e b<sub>34</sub> (0,22±1,27) são insignificantes ao nível de 95% de confiança. Com a exclusão destes coeficientes é necessário gerar um novo modelo de regressão, onde não ocorreu a formação de novas réplicas. Tal procedimento do recálculo do modelo está descrito nos vídeos preparados disponibilizados no Apêndice A. Dessa forma, com modelo final foram gerados os coeficientes e os respectivos



intervalos de confiança indicados pela Equação [7.4](#).

$$\begin{aligned}
 Resposta = & 92,28 \pm 1,14 + 6,17v_1 \pm 0,73 + 2,14v_2 \pm 0,73 + 2,79v_3 \pm 0,73 \\
 & - 5,70v_1^2 \pm 1,00 - 3,41v_2^2 \pm 1,00 - 2,00v_3^2 \pm 1,00 \\
 & + 1,99v_4 \pm 0,73 - 2,58v_4^2 \pm 1,00 - 1,32v_{12} \pm 1,27 \\
 & - 3,15v_{13} \pm 1,27 - 2,34v_{23} \pm 1,27 + 0,22v_{34} \pm 1,27 \quad (7.4)
 \end{aligned}$$

Assim, com os valores dos coeficientes gerados pela Equação [7.4](#) é possível determinar as condições ótimas de experimentação para obter o valor máximo de rendimento de benzaldeído através do cálculo das derivadas parciais, utilizando o método “solver\_diff” na biblioteca “pde.py”. Com isso, para obtenção de 94,53% de benzaldeído é preciso trabalhar com 0,84g de catalisador ( $v_1$ ), 62% v/v de  $H_2O_2$  em álcool benzílico, 4,4 horas de tempo de reação e 22 mL de água. Tais valores obtidos pela rotina “pde.py” foram obtidos pelo Octave por Pereira e Pereira-Filho (2018).

## 8 CONSIDERAÇÕES FINAIS

A linguagem Python vem sendo popularizada desde 2008 juntamente com a área de *data science*, do mesmo modo a procura por ferramentas computacionais livres que são intuitivas e que possibilitam um leque abrangente para diferentes contextos dentro da área de Quimiometria. Assim, a biblioteca “pde.py” tem o intuito de trazer um módulo externo que tenha todos os recursos necessários para execução do planejamento fatorial, permitindo identificar as variáveis determinantes para experimentação e determinar as condições ideais de trabalho através do modelo de regressão e, posteriormente, o cálculo de derivadas parciais ou a visualização de gráficos de superfície de resposta e contorno.

A finalidade deste trabalho de conclusão de curso foi implementar esta nova ferramenta em exemplos reais dentro do contexto do experimentador de forma clara e sucinta sobre todas etapas necessárias para o processamento de dados envolvidos nos quatro exemplos abordados, sendo estes: a construção de gráficos de porcentagem de efeitos *versus* efeitos e probabilidade *versus* efeitos para os exemplos de planejamento fatorial completo  $2^3$  e fracionário  $2^{4-1}$ ; já os dois últimos exemplos envolvem a determinação das condições ideais de trabalho com o uso de modelo de regressão para os planejamentos fatoriais tipo *Doehlert* e *Box-Behnken*. Dessa maneira, com intuito de tornar o uso da linguagem Python acessível para leitores que não tiveram o devido contato com esta ferramenta, foram preparados 63 vídeos que mostram tópicos desde assuntos básicos até a aplicação do Python em planejamentos fatoriais (vide Apêndice A). Além disso, também foi preparado um material suplementar que disponibiliza a rotina programacional “pde.py”, apostilas da playlist e outros materiais de estudo.

Tendo em vista a apresentação da resolução dos exemplos, pode-se afirmar que os

objetivos gerais e específicos foram alcançados, uma vez que os resultados foram reproduzidos pelo artigo publicado pela revista da *Química Nova* realizado por Pereira e Pereira-Filho (2018).

A rotina programacional atende as aplicações abordadas, mas ainda existe um grande leque de expansão de suas funções, por exemplo, tratamento de dados para planejamento de misturas ou até para aplicações envolvendo Análise de Componentes Principais (PCA) ou Análise Hierárquica de Agrupamentos (HCA). Devido a isso o projeto está aberto para novas implementações futuras e sugestões com uso do protocolo de sistema *open source*.

## REFERÊNCIAS

- BOX, G. E. et al. *Statistics for experimenters*. [S.l.]: John Wiley and sons New York, 1978. v. 664. Citado na página [18](#).
- BRO, R.; SMILDE, A. K. Principal component analysis. *Analytical methods*, Royal Society of Chemistry, v. 6, n. 9, p. 2812–2831, 2014. Citado na página [9](#).
- BRUNS, R. E.; SCARMINIO, I. S.; NETO, B. de B. *Statistical design-chemometrics*. [S.l.]: Elsevier, 2006. Citado na página [17](#).
- CHUI, Q. S. Uncertainties related to linear calibration curves: a case study for flame atomic absorption spectrometry. *Journal of the Brazilian Chemical Society*, SciELO Brasil, v. 18, p. 424–430, 2007. Citado na página [18](#).
- FERREIRA, M. M. C. *Quimiometria: conceitos, métodos e aplicações*. [S.l.]: Editora da UNICAMP, 2015. Citado na página [9](#).
- GUARDIA, M. d. I. An integrated approach of analytical chemistry. *Journal of the Brazilian Chemical Society*, SciELO Brasil, v. 10, p. 429–437, 1999. Citado na página [12](#).
- KOWALSKI, B. R. Chemometrics: views and propositions. *Journal of Chemical Information and Computer Sciences*, ACS Publications, v. 15, n. 4, p. 201–203, 1975. Citado na página [9](#).
- LEARDI, R. Experimental design in chemistry: A tutorial. *Analytica chimica acta*, Elsevier, v. 652, n. 1-2, p. 161–172, 2009. Citado na página [9](#).
- MORAIS, W. A. et al. Coencapsulation of trans-dehydrocrotonin and trans-dehydrocrotonin: hydroxypropyl- $\beta$ -cyclodextrin into microparticles. *Journal of the Brazilian Chemical Society*, SciELO Brasil, v. 28, p. 1494–1505, 2017. Citado 2 vezes nas páginas [16](#) e [21](#).
- NETO, B. B.; SCARMINIO, I. S.; BRUNS, R. E. *Planejamento e otimização de experimentos*. [S.l.]: Ed. da UNICAMP, 1996. Citado na página [16](#).
- NETO, B. B.; SCARMINIO, I. S.; BRUNS, R. E. *Como Fazer Experimentos-: Pesquisa e Desenvolvimento na Ciência e na Indústria*. [S.l.]: Bookman Editora, 2010. Citado 2 vezes nas páginas [16](#) e [21](#).
- NOVAES, C. et al. Otimização de métodos analíticos usando metodologia de superfícies de resposta-parte i: variáveis de processo. *Revista Virtual Química*, v. 9, n. 3, p. 1284–1215, 2017. Citado na página [17](#).

ORTIZ, M. C. Teaching chemometrics. *Analytical and bioanalytical chemistry*, Springer, v. 388, n. 8, p. 1557–1560, 2007. Citado na página [12](#).

PEREIRA, F. M. V.; PEREIRA-FILHO, E. R. Aplicação de programa computacional livre em planejamento de experimentos: um tutorial. *Química Nova*, SciELO Brasil, v. 41, p. 1061–1071, 2018. Citado 3 vezes nas páginas [9](#), [22](#) e [23](#).

PEREIRA-FILHO, E. R. Planejamento fatorial em química: maximizando a obtenção de resultados. *Edufscar: São Carlos*, 2015. Citado 4 vezes nas páginas [13](#), [14](#), [17](#) e [27](#).

PEREIRA-FILHO, E. R.; PEREIRA, F. M. V.; LEMOS, S. G. Introducing factorial design concepts using experiments with multi-colored chocolates. *Chem. Educ.*, v. 14, p. 200–203, 2009. Citado na página [14](#).

SANTOS, G. S. d. et al. Analytical strategies for determination and environmental impact assessment of inorganic antimony species in natural waters using hydride generation atomic fluorescence spectrometry (hg afs). *Journal of the Brazilian Chemical Society*, SciELO Brasil, v. 29, p. 185–190, 2018. Citado 3 vezes nas páginas [19](#), [23](#) e [27](#).

ZAVISKA, F. et al. Experimental design methodology applied to electrochemical oxidation of the herbicide atrazine using ti/iro<sub>2</sub> and ti/sno<sub>2</sub> circular anode electrodes. *Journal of hazardous materials*, Elsevier, v. 185, n. 2-3, p. 1499–1507, 2011. Citado na página [14](#).

## APPENDIX A – PLAYLIST 24: PYTHON

A Tabela 6, indica os exemplos tratados neste trabalho de conclusão de curso. Ao longo do desenvolvimento do trabalho foi preparada uma playlist com a finalidade de auxiliar o leitor com as etapas do uso do Python em planejamento fatorial. Além dos exemplos, há 47 vídeos iniciais (vídeos de 1 a 47) preparados que tratam os assuntos básicos da linguagem de programação para iniciantes. No link a seguir é possível acessar por meio do canal do YouTube do Professor Dr. Edenir Pereira Rodrigues Filho: <https://www.youtube.com/c/EdenirPereiraFilho>.

Tabela 6 – Lista de vídeos preparados para melhor entendimento do uso do Python em planejamento fatorial em química.

Identificação de vídeo	Descrição
vídeo48 play24	Introdução à Python em Planejamento Fatorial
vídeo49 play24	Importando matriz $\mathbf{X}$ e vetor $y$ do Exemplo 1
vídeo50 play24	Cálculo de erro de um efeito e valor de $t$ (Ex1)
vídeo51 play24	Uso da rotina "Fabi_efeito"ao Exemplo 1
vídeo52 play24	Recálculo e conclusão do Exemplo 1
vídeo53 play24	Importando matriz $\mathbf{X}$ e vetor $y$ do Exemplo 2
vídeo54 play24	Cálculo de erro de um efeito e valor de $t$ (Ex2)
vídeo55 play24	Uso da rotina "Fabi_efeito"ao Exemplo 2
vídeo56 play24	Importando matriz $\mathbf{X}$ e vetor $y$ do Exemplo 3
vídeo57 play24	Uso da rotina "Regression2"ao Exemplo 3
vídeo58 play24	Recalculando o modelo de regressão
vídeo59 play24	Uso da rotina "Super_fabi"
vídeo60 play24	Derivadas parciais do Exemplo 3
vídeo61 play24	Importando matriz $\mathbf{X}$ e vetor $y$ do Exemplo 4
vídeo62 play24	Uso da rotina "Regression2"ao Exemplo 4
vídeo63 play24	Derivadas parciais e conclusão do Exemplo 4

**Fonte:** De autoria própria, 2022.