METODOLOGIA DE SUPERFÍCIE DE RESPOSTA: UMA INTRODUÇÃO NOS SOFTWARES R E STATISTICA

Anaisa Comparini, Gabriela Passos, Helton Graziadei, Paulo H. Ferreira-Silva e Francisco Louzada

ICMC – USP – CP668 – CEP 13.566-590, São Carlos – SP – Brasil

louzada@icmc.usp.br

Resumo

A metodologia de superfície de resposta consiste em uma coleção de técnicas estatísticas e matemáticas útil para desenvolvimento, melhora e otimização de processos. Ela também tem aplicações importantes em planejamentos, desenvolvimento e formulação de novos produtos, e melhoria dos projetos e produtos existentes. A mais extensiva aplicação do RSM é na área industrial, particularmente em situações em que entram várias variáveis que potencialmente influenciam em alguma medida de desempenho ou na qualidade característica de um produto ou processo. E, essa medida de desempenho ou qualidade característica é chamada de resposta. (Myers e Montgomery, 1995).

Devido à enorme quantidade de dados disponíveis atualmente em diversas áreas, tem se tornado cada vez mais importante o desenvolvimento de métodos computacionais para obter informações relevantes desses dados de forma automática. Neste artigo, são apresentadas formas computacionais de se realizar um planejamento de experimento usando superfícies de resposta. Para isso, realizou-se uma comparação empírica entre os *softwares* R e STATISTICA, a fim de mostrar a veracidade de ambos. Para tal comparação usou-se o exemplo da lacase, que será abordado durante o artigo.

Palavras-chave: planejamento de experimento, planejamento composto central, tabela ANOVA, gráfico de contornos, superfície de resposta, *software* R, STATISTICA, lacase, análise de resíduos.

1. Introdução

A metodologia de superfície de resposta é uma técnica estatística utilizada para a modelagem e análise de problemas nos quais a variável resposta é influenciada por vários fatores, cujo objetivo é a otimização dessa resposta. O pacote RSM para R (R Development Core Team, 2009) fornece várias funções para facilitar os métodos de superfície de resposta. Até o momento, o pacote cobre apenas *designs* e métodos mais padronizados de primeira e segunda ordem para uma variável resposta.

Neste artigo, é apresentada uma visão geral do pacote RSM e de como as funções para análise de *design* das superfícies de resposta podem ser usadas. A codificação apropriada dos dados é um fator importante para a análise da superfície de resposta, portanto, o primeiro passo consiste em apresentar como são realizadas as codificações e decodificações dos níveis dos fatores. Depois disso, as gerações dos modelos padrões e a verificação da adequabilidade desses modelos são realizadas. O terceiro passo consiste em fornecer um resumo adequado da superfície de resposta e dos gráficos de contornos. No quarto e último passo, visualiza-se a superfície de resposta e a possível presença de valores críticos.

O artigo é organizado da seguinte maneira. Na Seção 2 é descrita a metodologia empregada na análise (comandos em R para codificação e decodificação dos dados, bem como os códigos para o ajuste de modelos de superfície de resposta neste *software*). Na Seção 3 são apresentados os resultados obtidos quando da aplicação da metodologia em questão a um conjunto de dados reais (dados de produção de lacase), juntamente com os resultados obtidos no *software* STATISTICA, para efeito de comparação dos dois *softwares* utilizados (R e STATISTICA). Comentários finais e conclusões, na Seção 4, finalizam o artigo.

2. Metodologia

2.1 Codificação dos dados

O pacote RSM possui várias funções que codificam e decodificam as variáveis, dentre elas estão as funções `coded.data` (transforma os valores preditos e substitui essas variáveis para a versão codificada), `decoded.data` (decodifica um objeto `coded.data`),

`as.coded.data` (cria um objeto `coded.data` a partir de dados que já estão codificados), `coded2val` (codifica matrizes e *data frames* de valores arbitrários), `val2code` (decodifica matrizes e *data frames* de valores arbitrários).

Os dados utilizados foram fornecidos por `ChemReact` (Tabela 7.6 de Myers; Montgomery; Anderson-Cook, 2009), presentes no pacote RSM.

Tabela 1 – Dados fornecidos por `ChemReact`.

Time	Temp	Block	Yield
80,00	170,00	B1	80,5
80,00	180,00	B1	81,5
90,00	170,00	B1	82,0
90,00	180,00	B1	83,5
85,00	175,00	B1	83,9
85,00	175,00	B1	84,3
85,00	175,00	B1	84,0
85,00	175,00	B2	79,7
85,00	175,00	B2	79,8
85,00	175,00	B2	79,5
92,07	175,00	B2	78,4
77,93	175,00	B2	75,6
85,00	182,07	B2	78,5
85,00	167,93	B2	77,0

Primeiramente, somente o bloco 1 (B1) foi analisado. Observa-se que os fatores tempo (Time) e temperatura (Temp) do bloco 1 têm seus valores variando 85 ± 5 e 175 ± 5, respectivamente, com 3 pontos centrais. Portanto, as variáveis codificadas são,

$$x_1 = (\text{Time} - 85)/5 \text{ e } x_2 = (\text{Temp} - 175)/5.$$

Para codificar os dados, utiliza-se a função `coded.data`, já citada anteriormente. Inicialmente, são utilizados apenas os dados do bloco 1.

```
> CR <- coded.data(ChemReact, x1 ~(Time - 85)/5, x2 ~(Temp - 175)/5)
> CR[1:7,]
```

Observa-se na Tabela 2, como a função `coded.data` transforma os valores preditos nas versões codificadas. Do mesmo modo, outras transformações podem ser feitas, utilizando as diferentes funções de codificação e decodificação de dados disponíveis no pacote RSM.

Tabela 2 – Dados codificados.

x_1	x_2	Block	Yield
-1	-1	B1	80.5
-1	1	B1	81.5
1	-1	B1	82.0
1	1	B1	83.5
0	0	B1	83.9
0	0	B1	84.3
0	0	B1	84.0

2.2 Ajustando um modelo de superfície de resposta

Para estudar o planejamento composto central considerando o conjunto de dados `ChemReact`, no qual produção (*Yield*) é a variável resposta, x_1 e x_2 são os dados já codificados e há os blocos, devem ser consideradas duas etapas. Na primeira etapa, ajusta-se um modelo de primeira ordem, utilizando os seguintes comandos em R:

```
> CR.rsm1 <- rsm(Yield~FO(x1,x2),data = CR,subset = (Block =="B1"))
> summary(CR.rsm1)
```

Tabela 3 – Saída do R ao executar o comando `summary(CR.rsm1)`.

Coefficients:				
	Estimate	**Std. Error**	**t value**	**Pr(>|t|)**
(Intercept)	82.8143	0.5472	151.346	1.14e-08***
x1	0.8750	0.7239	1.209	0.293
x2	0.6250	0.7239	0.863	0.437

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				

Analysis of Variance Table					
Response: Yield					
	Df	**Sum Sq**	**Mean Sq**	**F value**	**Pr(>F)**
FO(x1, x2)	2	4.6250	2.3125	1.1033	0.41534
Residuals	4	8.3836	2.0959		
Lack of fit	2	8.2969	4.1485	95.7335	0.01034
Pure error	2	0.0867	0.0433		

Como não há termos quadráticos e de interação no ajuste do modelo, nota-se que as variáveis x_1 e x_2 não são significativas (ver Tabela 3).

Devido à falta de ajuste significativa (p-valor ≈ 0,01), deve-se então utilizar um modelo de ordem maior. Incluem-se, então, interações:

```
> CR.rsm1.5 <- update(CR.rsm1,.~.+TWI(x1,x2))
> summary(CR.rsm1.5)
```

Tabela 4 – Saída do R ao executar o comando `summary(CR.rsm1.5)`.

Coefficients:				
	Estimate	**Std. Error**	**t value**	**Pr(>|t|)**
(Intercept)	82.8143	0.6295	131.560	9.68e-07***
x1	0.8750	0.8327	1.051	0.371
x2	0.6250	0.8327	0.751	0.507
x1:x2	0.1250	0.8327	0.150	0.890

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Analysis of Variance Table					
Response: Yield					
	Df	**Sum Sq**	**Mean Sq**	**F value**	**Pr(>F)**
FO(x1, x2)	2	4.6250	2.3125	0.8337	0.515302
TWI(x1,x2)	1	0.0625	0.0625	0.0225	0.890202
Residuals	3	8.3211	2.7737		
Lack of fit	1	8.2344	8.2344	190.0247	0.005221
Pure error	2	0.0867	0.0433		

Verifica-se novamente para este caso que a falta de ajuste é significativa, com p-valor ≈ 0,01 (ver Tabela 4). Os dados do bloco 2 são acrescentados a fim de montar um modelo de segunda ordem. Isso pode ser feito utilizando `SO(x1,x2)`, que inclui termos quadráticos e interação:

```
> CR.rsm2 <- rsm(Yield ~ Block + SO(x1, x2), data = CR)
> summary(CR.rsm2)
```

Segundo os resultados apresentados na Tabela 5, obtém-se agora falta de ajuste não significativa (p-valor ≈ 0,69). Observa-se também que o ponto de estacionariedade do modelo ajustado está em (0,37; 0,33), que é ponto de máximo, já que os autovalores deram negativos. Tem-se, provavelmente, um ponto de ótimo. Para confirmar isto, coletam-se alguns dados perto do ideal estimado: Tempo ≈ 87 e Temperatura ≈ 177.

Tabela 5 – Saída do R ao executar o comando `summary(CR.rsm2)`.

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	84.09543	0.07963	1056.067	<2e-16***
BlockB2	-4.45753	0.08723	-51.103	2.88e-10***
x1	0.93254	0.05770	16.162	8.44e-07***
x2	0.57771	0.05770	10.013	2.12e-05***
x1:x2	0.12500	0.08159	1.532	0.169
x1^2	-1.30856	0.06006	-21.786	1.08e-07***
x2^2	-0.93344	0.06006	-15.541	1.10e-06***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Analysis of Variance Table

Response: Yield

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Block	1	69.531	69.531	2611.0950	2.879e-10
FO(x1, x2)	2	9.626	4.813	180.7341	9.450e-07
TWI(x1,x2)	1	0.063	0.063	2.3470	0.1694
PQ(x1,x2)	2	17.791	8.896	334.0539	1.135e-07
Residuals	7	0.186	0.027		
Lack of fit	3	0.053	0.018	0.5307	0.6851
Pure error	4	0.133	0.033		

Stationary point of response surface:

x1	x2
0.3722954	0.3343802

Stationary point in original units:

Time	Temp
86.86148	176.67190

Eigenanalysis:

$values

-0.9233027 -1.3186949

$vectors

-0.1601375 -0.9870947

-0.9870947 0.1601375

3. Resultados

3.1 Exemplo da lacase

Foi utilizado um planejamento composto central para a investigação de como a produção de lacase depende da concentração de álcool veratrílico (variável x_1, em mM) e do

tempo de cultivo (variável x_2, em dias), em que a variável resposta é a atividade enzimática em U ml^{-1} (Barros Neto; Scarminio; Bruns, 2007).

Como há pontos centrais e axiais, ajusta-se um modelo de segunda ordem aos dados. Para isso, a função `lm` *(linear model*) deve ser utilizada da seguinte forma,

```
> fit.model <- lm(y~x1*x2+I(x1**2)+I(x2**2))
```

Ou então, usar a função `rsm`,

```
> fit.model2 <- rsm(y~SO(x1,x2))
```

Em R, x_1*x_2 considera os efeitos de interação, enquanto que x_1+x_2 considera apenas os efeitos marginais de x_1 e x_2.

A equação do modelo ajustado é dada por,

$$\hat{Y} = 4{,}93 - 1{,}18\, x_1 + 0{,}97\, x_2 - 0{,}70\, x_1^2 - 1{,}25\, x_2^2 - 0{,}21\, x_1 x_2$$

O comando `anova(fit.model)` testa a significância do modelo através da análise de variância. A tabela de análise de variância (ou tabela ANOVA) é dada por,

```
> summary(fit.model)
```

Tabela 6 – Sumário dos resíduos e estimativas dos coeficientes (saída do R).

Residuals:				
Min	**1Q**	**Median**	**3Q**	**Max**
-1.24555	-0.49263	0.04873	0.22912	2.18084

Coefficients:				
	Estimate	**Std. Error**	**t value**	**Pr(>|t|)**
(Intercept)	4.9262	0.4061	12.131	8.14e-09***
x1	-1.1824	0.2034	-5.814	4.49e-05***
x2	0.9711	0.2034	4.776	0.000296***
(x1^2)	-0.6973	0.2696	-2.586	0.021550*
(x2^2)	-1.2456	0.2696	-4.620	0.000397***
x1:x2	-0.2087	0.2872	-0.727	0.479228

Tabela 7 – Tabela ANOVA (saída do STATISTICA).

Effect Estimates										
Factor	**Effect**	**Std. Error**	**t(14)**	**p**	**-95% Cnf. Limt**	**95% Cnf Limt**	**Coeff.**	**Std. Err. Coeff.**	**-95% Cnf. Limt**	**95% Cnf. Limt**
Mean/ Interc.	4,926	0,406	12,131	0,000	4,055	5,797	4,926	0,406	4,055	5,797
(1) x1 (L)	-2,365	0,407	-5,814	0,000	-3,237	-1,492	-1,182	0,203	-1,619	-0,746
x1 (Q)	-1,395	0,539	-2,586	0,022	-2,551	-0,238	-0,697	0,270	-1,276	-0,119
(2)x2 (L)	1,942	0,407	4,776	0,000	1,070	2,815	0,971	0,203	0,535	1,407
x2 (Q)	-2,491	0,539	-4,620	0,000	-3,648	-1,335	-1,246	0,270	-1,824	-0,667
1L by 2L	-0,418	0,574	-0,727	0,479	-1,649	0,814	-0,209	0,287	-0,825	0,407

Verificando a significância do modelo, nota-se que a interação não é significante (ver Tabela 7). Um novo modelo a ser proposto excluiria o termo de interação.

3.1.1 Suposição de normalidade

A verificação da suposição de normalidade dos resíduos é essencial para prosseguir nas análises do modelo. Para isso, utiliza-se o gráfico normal probabilístico, construído a partir dos seguintes comandos em R:

```
> fit.model <- lm(y~x1*x2+I(x1**2)+I(x2**2))     #Ajuste do modelo de
2ª ordem#
> residuos <- residuals(fit.model)
> residuos <- sort(residuos)                      #Ordena os resíduos#
> n <- length(y)
> quantis <- qnorm( (1:n - 0.5)/n )       #Quantis teóricos da normal
padrão#
> plot(residuos, quantis, xlab="Resíduos", ylab="Quantis teóricos da
N(0,1)", main="Gráfico de probabilidade normal", pch=16, xlim=c(-
2,3), ylim=c(-3,3))
> abline(0,1)
```

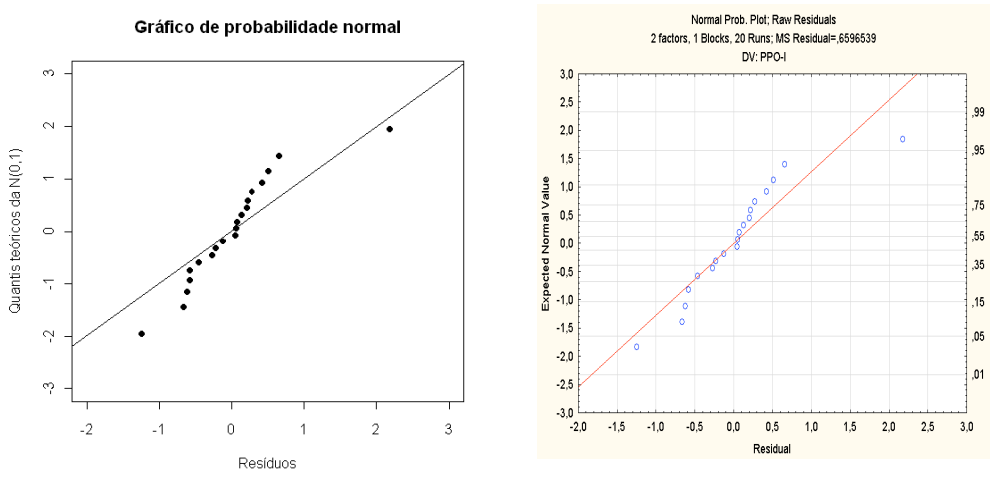

Figura 1 – Gráfico normal probabilístico dos resíduos. Painel à esquerda: *software* R. Painel à direita: *software* STATISTICA.

Analisando os gráficos acima (ver Figura 1), nota-se que há leve fuga de normalidade com possível presença de *outliers*. Mas, para efeito de exemplo, não será rejeitada aqui a suposição de normalidade.

3.1.2 Suposição de variância constante

Outra suposição importante do modelo ajustado é a de que os resíduos possuem variância constante (homocedasticidade). A verificação dessa suposição é feita analisando os gráficos de preditos x resíduos ou de nº da observação x resíduos, construídos através dos seguintes comandos em R:

```
> fit.model <- lm(y~x1*x2+I(x1**2)+I(x2**2))     #Ajuste do
modelo de 2ª ordem#
> residuos <- residuals(fit.model)
> preditos <- predict(fit.model)     #Valores preditos do modelo#
> plot(preditos, residuos, main="Valores preditos x resíduos",
pch=16)
> i <- 1:20
> plot(i,residuos, xlab="Observações",main="Número da observação
x Resíduos", pch=16)
```

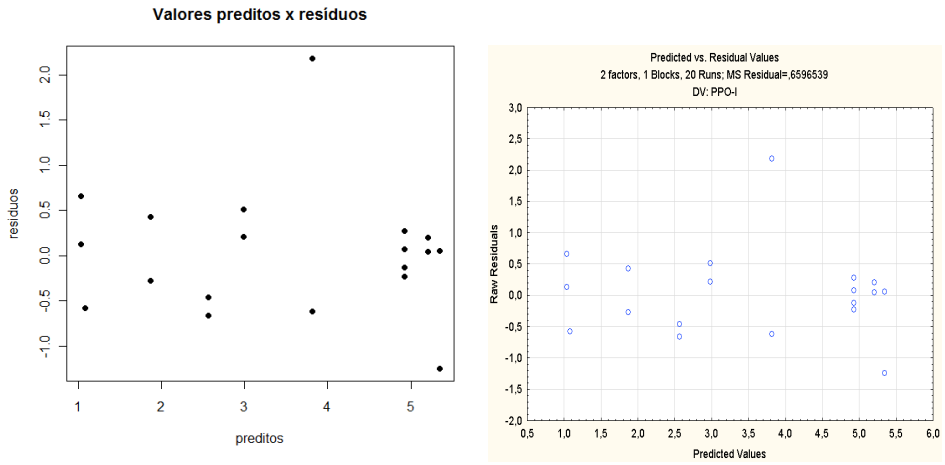

Figura 2 – Gráfico de preditos x resíduos. Painel à esquerda: *software* R. Painel à direita: *software* STATISTICA.

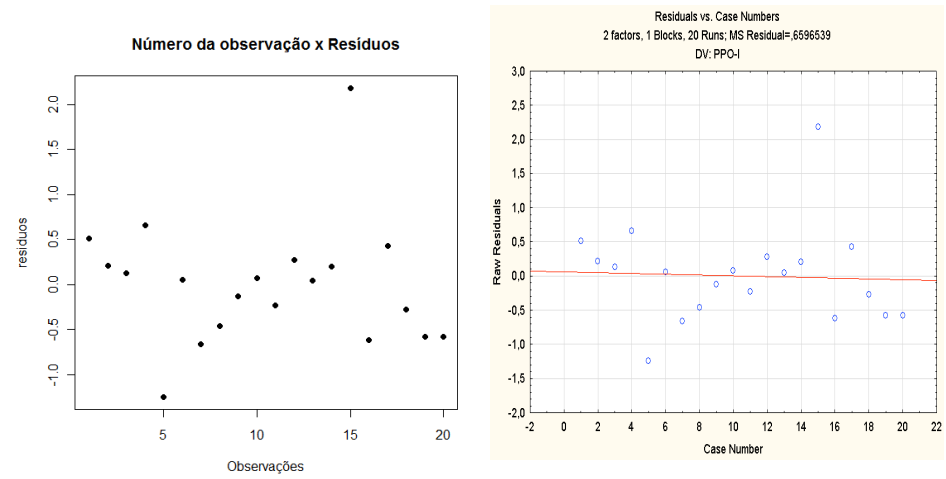

Figura 3 – Gráfico de nº da observação x resíduos. Painel à esquerda: *software* R. Painel à direita: *software* STATISTICA.

Observa-se que não há padrões aparentemente detectáveis em ambos os gráficos construídos (ver Figuras 2 e 3). Com isso, a suposição de variância constante (homocedasticidade) dos resíduos é válida para o modelo ajustado.

3.1.3 Ajuste do modelo

A qualidade do ajuste pode ser verificada através do gráfico de valores observados x valores preditos. Quanto mais os dados se ajustam à reta identidade, melhor é a qualidade do ajuste. Através dos seguintes comandos em R, constrói-se tal gráfico:

```
> preditos <- predict(fit.model)
> plot(y, preditos, main="Gráfico de valores observados x valores preditos", pch=16)
> abline(0,1)
```

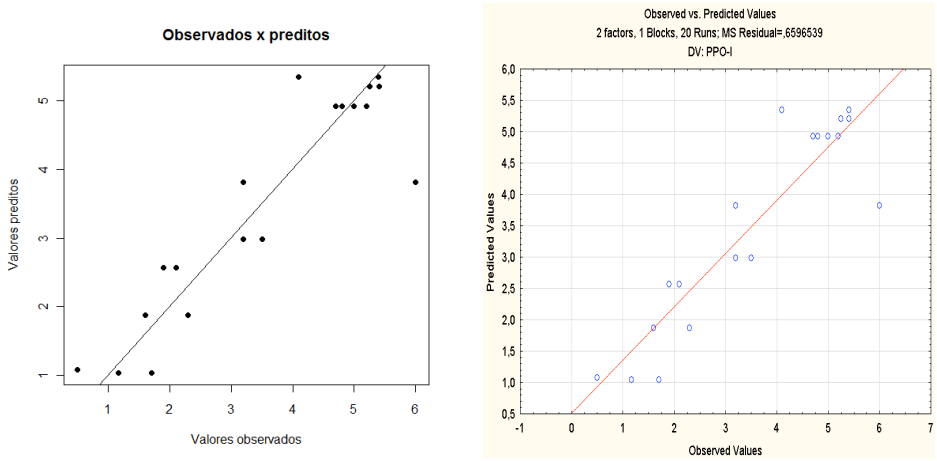

Figura 4 – Gráfico dos valores observados x valores preditos. Painel à esquerda: *software* R. Painel à direita: *software* STATISTICA.

Note que o modelo possui boa qualidade de ajuste, já que prevê corretamente boa parte das vezes e, quando comete um erro, não se distancia muito da reta identidade (ver Figura 4).

3.1.4 Gráfico de contornos e superfície de resposta

A metodologia de superfície de resposta possui algumas limitações no *software* R. Para construir as curvas de níveis e a superfície de resposta, pode ser empregada a função `lm` *(linear model*) ou a função `rsm` (*response surface methodology*).

O gráfico de contornos pode auxiliar na localização de um possível ponto de ótimo. Para isso, os comandos em R a serem utilizados, são dados por:

```
> fit.model <- rsm(y~SO(x1,x2))
> contour(fit.model, ~x1+x2, main="Gráfico de contornos")
```

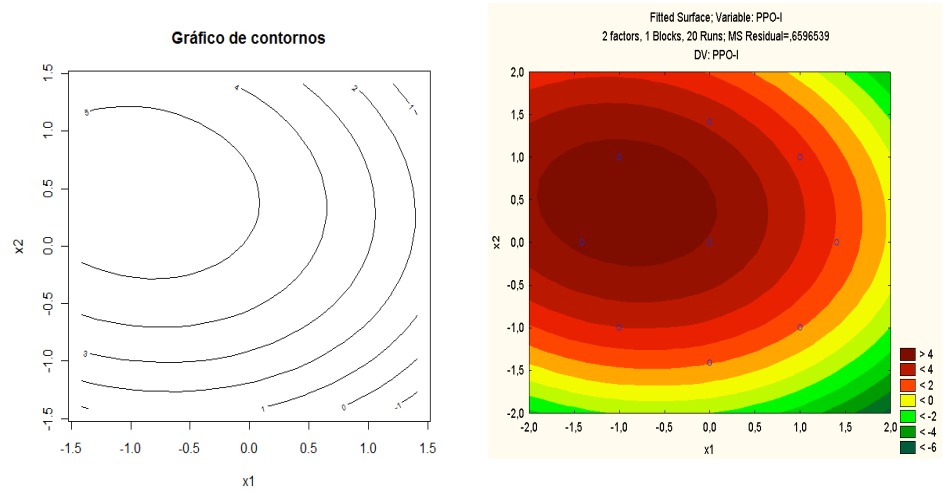

Figura 5 – Gráfico de contornos. Painel à esquerda: *software* R. Painel à direita: *software* STATISTICA.

Através dos gráficos de contornos acima (ver Figura 5), é evidente que, tomando um nível baixo de x_1 (concentração) e um nível alto de x_2 (tempo de cultivo), são obtidos rendimentos mais otimizados. Além disso, o ponto ótimo de rendimento (máximo) está na parte superior esquerda.

Observe que há diferenças entre os gráficos de contornos gerados pelo R e pelo STATISTICA (nos limitantes de x_1 e x_2), embora isso não influa na localização do ponto de máximo nesse exemplo.

A superfície de resposta gerada pelo STATISTICA é apresentada na Figura 6 (painel à direita) e a superfície de resposta em R, também apresentada na Figura 6 (painel à esquerda), foi gerada através do comando `persp`:

```
> persp(fit.model, ~x1+x2, main="Superfície de resposta",
col=rainbow(40))
```

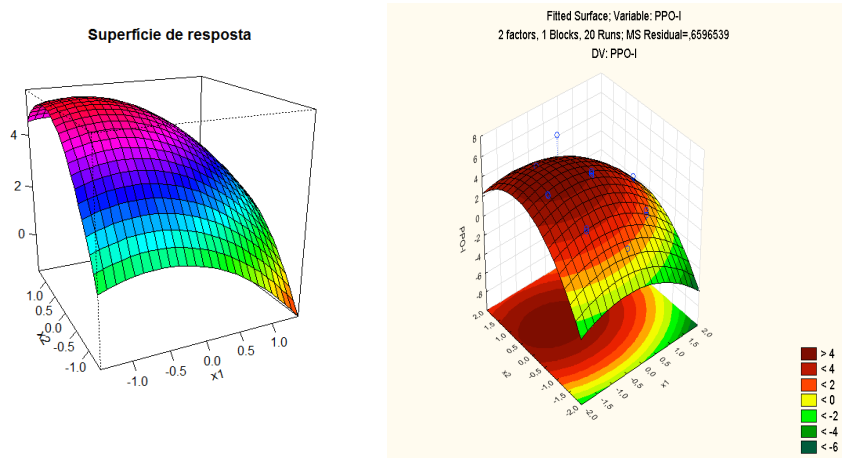

Figura 6 – Superfície de resposta. Painel à esquerda: *software* R. Painel à direita: *software* STATISTICA.

4. Comentários Finais

A metodologia de superfície de resposta é uma coleção de técnicas estatísticas úteis para a otimização de processos. O planejamento composto central é um tipo de experimento para a construção de um modelo de 2ª ordem para a variável resposta, sem a necessidade de usar um modelo com três níveis completo (*full*). É uma composição entre o planejamento fatorial 2^k (com um ou mais pontos centrais) e uma parte axial.

Inicialmente, foi desenvolvida a codificação dos dados (`ChemReact`, presente no pacote RSM) em R, de variáveis naturais para variáveis codificadas e, após isso, foi ajustado um modelo de 2ª ordem através das funções `lm` e/ou `rsm`.

No exemplo da lacase, exposto por Barros Neto, Scarminio e Bruns (2007), foi ajustado um modelo composto central aos dados e testada a significância dos componentes. Após isso, as suposições do modelo foram verificadas utilizando tanto o *software* livre R quanto o STATISTICA. Dado que o modelo ficou bem ajustado, pôde-se verificar se havia ponto de ótimo na região experimental, através do auxílio dos gráficos de contornos e da superfície de resposta. Com isso, encontrou-se uma região de rendimento máximo.

Neste artigo, também foi considerado um exemplo, encontrado em Box, Hunter e Hunter (2005), o qual não foi exposto no corpo do texto. Consiste na construção de helicópteros de papel, no qual o objetivo é maximizar o tempo médio de vôo desses helicópteros. Assim, observou-se que as estimativas do modelo, o gráfico normal

probabilístico, o gráfico de resíduos x preditos, as curvas de níveis e a superfície de resposta não sofreram alterações consideráveis ao serem executadas em R e no STATISTICA.

A construção do gráfico de contornos em R possui certas deficiências, principalmente em relação à manipulação dos limitantes das covariáveis. Apesar dessas dificuldades, foi possível encontrar uma região em que o rendimento fosse máximo.

5. Referências bibliográficas

- BARROS NETO, B.; SCARMINIO, I. S.; BRUNS, R. E. **Como fazer experimentos**. 3. ed. Campinas: Editora Unicamp , 2007.

- BOX, G. E. P.; HUNTER, J. S.; HUNTER, W. G. **Statistics for experimenters**. 2nd ed. Danvers: John Wiley Professional, 2005. (Wiley Series in Probability and Statistics).

- LENTH, R. V. **Response-Surface Methods in R, Using rsm**. Journal of Statistical Software, 32(7), 1-17, 2009. Disponível em:
<http://cran.r-project.org/web/packages/rsm/vignettes/rsm.pdf>.

- MONTGOMERY, D. C. **Design and Analysis of Experiments**. 7th Edition, John Wiley and Sons, 2008.

- MYERS, R. H.; MONTGOMERY, D. C. **Response surface methodology: process and product optimization using designed experiments**. 2nd ed. New York: John Wiley Professional, 1995. (Wiley Series in Probability and Statistics).

- MYERS, R. H.; MONTGOMERY, D. C.; ANDERSON-COOK, C. M. **Response Surface Methodology: Process and Product Optimization Using Designed Experiments**. 3rd Ed. New York: Wiley, 2009.

- R Development Core Team (2009). **R: A language and environment for statistical computing**. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.