

1. Classificação INPE.COM.2 /NTI C.D.U.: 518.12	2. Período	4. Critério de Distribuição: interna <input checked="" type="checkbox"/> externa <input type="checkbox"/>
3. Palavras Chaves (selecionadas pelo autor) <i>Interpolação</i> <i>Mínimos Quadrados</i> <i>Mínimos Quadrados Ponderados</i> <i>Curvas de Nível</i>		
5. Relatório nº <i>INPE-1278-NTI/106</i>	6. Data <i>Junho, 1978</i>	7. Revisado por <i>C. R. Souza</i> <i>Celso de Renna e Souza</i>
8. Título e Sub-Título <i>MÉTODO DE INTERPOLAÇÃO POR MÍNIMOS QUADRADOS PONDERADOS</i>		9. Autorizado por <i>N. Parada</i> <i>Nelson de Jesus Parada</i> <i>Diretor</i>
10. Setor <i>DSE/DIN</i>	Código	11. Nº de cópias <i>06</i>
12. Autoria <i>Nandamudi L. Vijay Kumar</i>		14. Nº de páginas <i>48</i>
13. Assinatura Responsável <i>N. L. Vijaykumar</i>		15. Preço
16. Sumário/Notas <i>O trabalho descreve um método de interpolação por computador digital. Este método é baseado na técnica de aproximação por mínimos quadrados ponderados, que utiliza pesos para se obter os coeficientes de um polinômio quadrático, que age como uma aproximação a uma superfície dada. Esses pesos dependem das distâncias dos pontos da superfície ao ponto onde o polinômio está sendo ajustado.</i>		
17. Observações		

ÍNDICE

ABSTRACT	ii
1. INTRODUÇÃO	1
2. MÉTODO DOS MÍNIMOS QUADRADOS	1
3. MÉTODO DOS MÍNIMOS QUADRADOS PONDERADOS	7
4. IMPLEMENTAÇÃO	13
5. FUNÇÃO PESO	15
6. EQUAÇÕES TESTADAS	17
6.1. PARTE DE UMA ESFERA	18
6.2. SUBIDA INGREME SAINDO DE UM PLANO	20
6.3. SUBIDA MENOS INGREME	21
6.4. UM PLANALTO E UM PLANO SEPARADOS POR UMA PAREDE INGREME	22
6.5. RELEVO DE TESTE	23
7. CONCLUSÕES	25
AGRADECIMENTOS	27
REFERÊNCIAS BIBLIOGRÁFICAS	28
APÊNDICE A	A.1
APÊNDICE B	B.1
APÊNDICE C	C.1

ABSTRACT

The report describes an interpolating method by a digital computer. This method is based on weighted least squares approximation technique, using weights in order to obtain a quadratic polinomial, which acts as an approximation to a given surface. These weights depend on the distances of the surface data points to the point where the polinomial is being fit.

1. INTRODUÇÃO

O problema de interpolação em uma superfície, irregularmente amostrada, é ainda objeto de constantes pesquisas, pois aparentemente não existe uma solução "ótima", sendo cada caso estudado individualmente.

Tendo surgido no INPE, o problema do traçado de isolinhas, surgiu também, como consequência, o da interpolação de superfícies. Um método de fácil implantação e de bons resultados, no caso geral, recomendado pela literatura, é o de mínimos quadrados ponderados, que é o objeto do presente trabalho no que tange a sua implantação e testes. Um outro, também proposto, é o de ajuste por transformadas de Fourier, que será objeto de trabalho futuro.

Este trabalho consiste na implantação de uma rotina para o ajuste por mínimos quadrados ponderados. O polinômio usado para o ajuste é de segundo grau e bi-dimensional. Graus mais elevados não são usados por dificultarem a especificação do polinômio.

O programa de mínimos quadrados ponderados basicamente consiste em se montar as equações na forma de matrizes para obter-se os coeficientes do polinômio quadrático. Assim, para qualquer ponto (a,b), encontra-se o valor do polinômio $z = f(a,b)$.

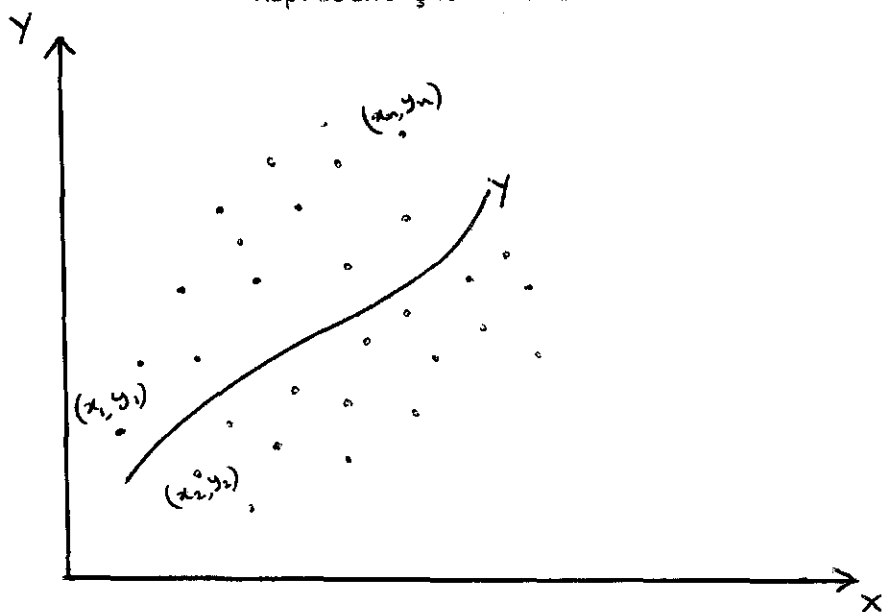
2. MÉTODO DOS MÍNIMOS QUADRADOS

Suponhamos ser nosso problema representar por um polinômio de grau n

$$Y(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n \text{ e}$$

um conjunto de n pontos $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$.

Representação Gráfica:



O método de mínimos quadrados é um dispositivo de ajustar uma curva a um conjunto de pontos dados. Em geral, os pontos usualmente obtidos de observação são dados pelos pares: (x_1, y_1) , (x_2, y_2) , ..., (x_n, y_n) , e, por exemplo x_1 , há uma diferença entre o valor observado, y_1 , e o valor correspondente da curva Y . As distancias podem ser positivas, negativas ou zero. Um "ajuste" é determinado pela soma das diferenças ao quadrado.

$$(y_1 - y(x_1))^2 + (y_2 - y(x_2))^2 + \dots + (y_n - y(x_n))^2$$

a maior ou menor grandeza desta soma indica uma pior ou melhor aproximação a uma curva dada.

Então, de todas as curvas aproximando um conjunto de pontos dados, a curva tendo a propriedade de que $(y_1 - y(x_1))^2 + (y_2 - y(x_2))^2 + \dots + (y_n - y(x_n))^2$ é mínimo é chamada de "melhor curva de ajuste".

Uma curva que tenha esta propriedade é suposta ajustar os dados, no "sentido de mínimos quadrados", e é chamada de "curva de mínimos quadrados".

O método matemático segue abaixo:

$$Q = \sum_{i=1}^n (Y(\underline{x_i}) - Y_i)^2$$

Derivando com respeito aos coeficientes vem:

$$\frac{\partial Q}{\partial a_0} = 0 \rightarrow \sum_{i=1}^n (Y(\underline{x_i}) - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_n x_i^n)) = 0$$

$$\frac{\partial Q}{\partial a_1} = 0 \rightarrow \sum_{i=1}^n (Y(\underline{x_i}) - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_n x_i^n)) \cdot x_i = 0$$

$$\frac{\partial Q}{\partial a_2} = 0 \rightarrow \sum_{i=1}^n (Y(\underline{x_i}) - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_n x_i^n)) \cdot x_i^2 = 0 \quad (1)$$

$$\vdots \quad \quad \quad \vdots \quad \quad \quad \vdots$$

$$\frac{\partial Q}{\partial a_n} = 0 \rightarrow \sum_{i=1}^n (Y(\underline{x_i}) - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_n x_i^n)) x_i^n = 0$$

onde a derivada igual a zero indica um máximo ou mínimo local.

Agrupando os termos, temos:

$$\sum_{i=1}^n Y(x_i) - \sum_{i=1}^n a_0 - a_1 \sum_{i=1}^n x_i - a_2 \sum_{i=1}^n x_i^2 - \dots - a_n \sum_{i=1}^n x_i^n = 0$$

$$\sum_{i=1}^n x_i Y(x_i) - a_0 \sum_{i=1}^n x_i - a_1 \sum_{i=1}^n x_i^2 - a_2 \sum_{i=1}^n x_i^3 - \dots - a_n \sum_{i=1}^n x_i^{n+1} = 0$$

$$\sum_{i=1}^n x_i^2 Y(x_i) a_0 - \sum_{i=1}^n x_i^2 - a_1 \sum_{i=1}^n x_i^3 - a_2 \sum_{i=1}^n x_i^4 - \dots - a_n \sum_{i=1}^n x_i^{n+2} = 0 \quad (2)$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots$$

$$\sum_{i=1}^n x_i^n Y(x_i) a_0 - \sum_{i=1}^n x_i^n - a_1 \sum_{i=1}^n x_i^{n+1} - a_2 \sum_{i=1}^n x_i^{n+2} - \dots - a_n \sum_{i=1}^n x_i^{2n} = 0$$

ou ainda:

$$n a_0 + a_1 \sum_{i=1}^n x_i + a_2 \sum_{i=1}^n x_i^2 + \dots + a_n \sum_{i=1}^n x_i^n = \sum_{i=1}^n Y(x_i)$$

$$a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 + a_2 \sum_{i=1}^n x_i^3 + \dots + a_n \sum_{i=1}^n x_i^{n+1} = \sum_{i=1}^n x_i Y(x_i)$$

$$a_0 \sum_{i=1}^n x_i^2 + a_1 \sum_{i=1}^n x_i^3 + a_2 \sum_{i=1}^n x_i^4 + \dots + a_n \sum_{i=1}^n x_i^{n+2} = \sum_{i=1}^n x_i^2 Y(x_i) \quad (3)$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots$$

$$a_0 \sum_{i=1}^n x_i^n + a_1 \sum_{i=1}^n x_i^{n+1} + a_2 \sum_{i=1}^n x_i^{n+2} + \dots + a_n \sum_{i=1}^n x_i^{2n} = \sum_{i=1}^n x_i^n Y(x_i)$$

Colocando em forma matricial vem:

$$\begin{array}{c}
 \begin{matrix}
 n & \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \dots & \sum_{i=1}^n x_i^n \\
 \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i^3 & \dots & \sum_{i=1}^n x_i^{n+1} \\
 \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^4 & \dots & \sum_{i=1}^n x_i^{n+2} \\
 \vdots & \vdots & \vdots & & \vdots \\
 \sum_{i=1}^n x_i^n & \sum_{i=1}^n x_i^{n+1} & \sum_{i=1}^n x_i^{n+2} & \dots & \sum_{i=1}^n x_i^{2n}
 \end{matrix}
 \end{array}
 \cdot
 \begin{bmatrix}
 a_0 \\
 a_1 \\
 a_2 \\
 \vdots \\
 a_n
 \end{bmatrix}$$

$$=
 \begin{bmatrix}
 \sum_{i=1}^n Y(x_i) \\
 \sum_{i=1}^n x_i Y(x_i) \\
 \sum_{i=1}^n x_i^2 Y(x_i) \\
 \vdots \\
 \sum_{i=1}^n x_i^n Y(x_i)
 \end{bmatrix}$$

(4)

Observamos que a matriz quadrada é simétrica. Para chegar às equações existe uma regra mnemônica, que é a seguinte:

Tomando o polinômio na sua forma original

$$Y(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n,$$

escrevendo-o, para melhor visualização da regra mnemônica, em forma matricial, temos

$$\begin{pmatrix} 1 & x & x^2 & \dots & x^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = Y(x) \quad (5)$$

Multiplicando a equação (5) pelo seu primeiro elemento que é 1 e aplicando a somatória $\sum_{i=1}^n$ obtemos a primeira equação do sistema 3.

$$a_0 + a_1 \sum_{i=1}^n x_i + a_2 \sum_{i=1}^n x_i^2 + \dots + a_n \sum_{i=1}^n x_i^n = \sum_{i=1}^n Y(x_i)$$

Para obtermos a segunda equação do sistema 3, multiplicamos a equação 5, pelo seu segundo elemento que é x e aplicamos a somatória $\sum_{i=1}^n$,

$$a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 + a_2 \sum_{i=1}^n x_i^3 + \dots + a_n \sum_{i=1}^n x_i^{n+1} = \sum_{i=1}^n x_i Y(x_i)$$

Procedendo da maneira análoga, para obtermos a k-ésima e quação do sistema 3, multiplicamos a equação 5 pelo seu k-ésimo elemento e aplicamos a somatória $\sum_{i=1}^n$.

$$a_0 \sum_{i=1}^n x_i^k + a_1 \sum_{i=1}^n x_i^{k+1} + a_2 \sum_{i=1}^n x_i^{k+2} + \dots + a_n \sum_{i=1}^n x_i^{n+k} = \sum_{i=1}^n x_i^k Y(x_i)$$

3. MÉTODOS DOS MÍNIMOS QUADRADOS PONDERADOS

Quando se trata de duas variáveis e de um grande número de pontos (x_i, y_i, z_i) , $i = 1, \dots, n$, não é aconselhável usar o método clássico de ajustar um polinômio $z = f(x, y)$ de grau muito alto para fornecer um ajuste exato em todos os n pontos. Haveria uma dificuldade de especificar o tal polinômio e, além disso, a superfície resultante seria extremamente "dobrada" e "montanhosa". Portanto trataremos do caso de polinômios de grau dois.

Supondo que queremos estimar a superfície $f(a, b)$ no ponto (a, b) , então temos o polinômio quadrático $P(x, y) = c_{00} + c_{10}x + c_{01}y + c_{20}x^2 + c_{11}xy + c_{02}y^2$, que será utilizado no sentido do método usual de mínimos quadrados. A diferença entre o método usual de mínimos quadrados e o método presente é que, neste, são necessários aqueles pontos (x_i, y_i) próximos ao (a, b) para "carregar mais peso" do que os pontos mais distantes.

O método matemático segue abaixo, na mesma sequência:

$$Z(x, y) = c_{00} + c_{10}x + c_{01}y + c_{20}x^2 + c_{11}xy + c_{02}y^2$$

$$Q = \sum_{i=1}^n (Z(x_i, y_i) - z_i)^2 w(x_i, y_i) \text{ onde } w \text{ é a função peso}$$

$$\frac{\partial Q}{\partial c_{00}} = 0 \rightarrow \sum_{i=1}^n w(x_i, y_i) (Z - (c_{00} + c_{10}x_i + c_{01}y_i + c_{20}x_i^2 + c_{11}x_iy_i + c_{02}y_i^2)) = 0$$

$$\frac{\partial Q}{\partial c_{10}} = 0 \rightarrow \sum_{i=1}^n w(x_i, y_i) (Z - (c_{00} + c_{10}x_i + c_{01}y_i + c_{20}x_i^2 + c_{11}x_iy_i + c_{02}y_i^2)) x_i = 0$$

$$\frac{\partial Q}{\partial c_{01}} = 0 \rightarrow \sum_{i=1}^n w(x_i, y_i) (Z - (c_{00} + c_{10}x_i + c_{01}y_i + c_{20}x_i^2 + c_{11}x_iy_i + c_{02}y_i^2)) y_i = 0 \quad (6)$$

$$\frac{\partial Q}{\partial c_{20}} = 0 \rightarrow \sum_{i=1}^n w(x_i, y_i) (Z - (c_{00} + c_{10}x_i + c_{01}y_i + c_{20}x_i^2 + c_{11}x_iy_i + c_{02}y_i^2)) x_i^2 = 0$$

$$\frac{\partial Q}{\partial c_{11}} = 0 \rightarrow \sum_{i=1}^n w(x_i, y_i) (Z - (c_{00} + c_{10}x_i + c_{01}y_i + c_{10}x_i^2 + c_{11}x_iy_i + c_{02}y_i^2)) x_i y_i = 0$$

$$\frac{\partial Q}{\partial c_{02}} = 0 \rightarrow \sum_{i=1}^n w(x_i, y_i) (Z - (c_{00} + c_{10}x_i + c_{01}y_i + c_{10}x_i^2 + c_{11}x_iy_i + c_{02}y_i^2)) y_i^2 = 0$$

$$\sum_{i=1}^n w(x_i, y_i) Z - \sum_{i=1}^n w(x_i, y_i) c_{00} - c_{10} \sum_{i=1}^n w(x_i, y_i) x_i - c_{01} \sum_{i=1}^n w(x_i, y_i) y_i -$$

$$- c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^2 - c_{11} \sum_{i=1}^n w(x_i, y_i) x_i y_i - c_{02} \sum_{i=1}^n w(x_i, y_i) y_i^2 = 0$$

$$\sum_{i=1}^n w(x_i, y_i) x_i Z - c_{00} \sum_{i=1}^n w(x_i, y_i) x_i - c_{10} \sum_{i=1}^n w(x_i, y_i) x_i^2 - c_{01} \sum_{i=1}^n w(x_i, y_i) x_i y_i -$$

$$- c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^3 - c_{11} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i - c_{02} \sum_{i=1}^n w(x_i, y_i) x_i y_i^2 = 0$$

$$\sum_{i=1}^n w(x_i, y_i) y_i Z - c_{00} \sum_{i=1}^n w(x_i, y_i) y_i - c_{10} \sum_{i=1}^n w(x_i, y_i) x_i y_i - c_{01} \sum_{i=1}^n w(x_i, y_i) y_i^2 -$$

$$- c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i - c_{11} \sum_{i=1}^n w(x_i, y_i) x_i y_i^2 - c_{02} \sum_{i=1}^n w(x_i, y_i) y_i^3 = 0$$

(7)

$$\sum_{i=1}^n w(x_i, y_i) x_i^2 Z - c_{00} \sum_{i=1}^n w(x_i, y_i) x_i^2 - c_{10} \sum_{i=1}^n w(x_i, y_i) x_i^3 - c_{01} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i -$$

$$- c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^3 - c_{11} \sum_{i=1}^n w(x_i, y_i) x_i^3 y_i - c_{02} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i^2 = 0$$

$$\sum_{i=1}^n w(x_i, y_i) x_i y_i Z - c_{00} \sum_{i=1}^n w(x_i, y_i) x_i y_i - c_{10} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i -$$

$$- c_{01} \sum_{i=1}^n w(x_i, y_i) x_i y_i^2 - c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^3 y_i - c_{11} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i^2 -$$

$$- c_{02} \sum_{i=1}^n w(x_i, y_i) x_i y_i^3 = 0$$

$$\sum_{i=1}^n w(x_i, y_i) y_i^2 Z - c_{00} \sum_{i=1}^n (x_i, y_i) y_i^2 - c_{10} \sum_{i=1}^n w(x_i, y_i) x_i y_i - c_{01} \sum_{i=1}^n w(x_i, y_i) y_i^3 -$$

$$- c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i - c_{11} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i - c_{02} \sum_{i=1}^n w(x_i, y_i) y_i^4 = 0$$

$$\sum_{i=1}^n w(x_i, y_i) c_{00} + c_{10} \sum_{i=1}^n w(x_i, y_i) x_i + c_{01} \sum_{i=1}^n w(x_i, y_i) y_i + c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^2 +$$

$$+ c_{11} \sum_{i=1}^n w(x_i, y_i) x_i y_i + c_{02} \sum_{i=1}^n w(x_i, y_i) y_i^2 = \sum_{i=1}^n w(x_i, y_i) Z$$

$$c_{00} \sum_{i=1}^n w(x_i, y_i) x_i + c_{10} \sum_{i=1}^n w(x_i, y_i) x_i^2 + c_{01} \sum_{i=1}^n w(x_i, y_i) x_i y_i + c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^3 +$$

$$+ c_{11} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i + c_{02} \sum_{i=1}^n w(x_i, y_i) x_i y_i^2 = \sum_{i=1}^n w(x_i, y_i) x_i Z$$

$$c_{00} \sum_{i=1}^n w(x_i, y_i) y_i + c_{10} \sum_{i=1}^n w(x_i, y_i) x_i y_i + c_{01} \sum_{i=1}^n w(x_i, y_i) y_i^2 + c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i +$$

$$+ c_{11} \sum_{i=1}^n w(x_i, y_i) x_i y_i^2 + c_{02} \sum_{i=1}^n w(x_i, y_i) y_i^3 = \sum_{i=1}^n w(x_i, y_i) y_i Z$$

$$c_{00} \sum_{i=1}^n w(x_i, y_i) x_i^2 + c_{10} \sum_{i=1}^n w(x_i, y_i) x_i^3 + c_{01} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i + c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^4 +$$

$$+ c_{11} \sum_{i=1}^n w(x_i, y_i) x_i^3 y_i + c_{02} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i^2 = \sum_{i=1}^n w(x_i, y_i) x_i^2 Z$$

$$c_{00} \sum_{i=1}^n w(x_i, y_i) x_i y_i + c_{10} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i + c_{01} \sum_{i=1}^n w(x_i, y_i) x_i y_i^2 +$$

$$+ c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^3 y_i + c_{11} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i^2 + c_{02} \sum_{i=1}^n w(x_i, y_i) x_i y_i^3 =$$

$$= \sum_{i=1}^n w(x_i, y_i) x_i y_i Z$$

$$c_{00} \sum_{i=1}^n w(x_i, y_i) y_i^2 + c_{10} \sum_{i=1}^n w(x_i, y_i) x_i y_i^2 + c_{01} \sum_{i=1}^n w(x_i, y_i) y_i^3 +$$

$$+ c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i^2 + c_{11} \sum_{i=1}^n w(x_i, y_i) x_i y_i^3 + c_{02} \sum_{i=1}^n w(x_i, y_i) y_i^3 = \sum_{i=1}^n w(x_i, y_i) y_i^2 Z$$

$\sum_{i=1}^n w(x_i, y_i) x_i$	$\sum_{i=1}^n w(x_i, y_i) y_i$	$\sum_{i=1}^n w(x_i, y_i) x_i^2$	$\sum_{i=1}^n w(x_i, y_i) x_i y_i$	$\sum_{i=1}^n w(x_i, y_i) y_i^2$	c_{00}
$\sum_{i=1}^n w(x_i, y_i) x_i^2$	$\sum_{i=1}^n w(x_i, y_i) x_i y_i$	$\sum_{i=1}^n w(x_i, y_i) x_i^3$	$\sum_{i=1}^n w(x_i, y_i) x_i^2 y_i$	$\sum_{i=1}^n w(x_i, y_i) x_i y_i^2$	c_{01}
$\sum_{i=1}^n w(x_i, y_i) x_i y_i$	$\sum_{i=1}^n w(x_i, y_i) y_i^2$	$\sum_{i=1}^n w(x_i, y_i) x_i^2 y_i$	$\sum_{i=1}^n w(x_i, y_i) x_i y_i^2$	$\sum_{i=1}^n w(x_i, y_i) y_i^3$	c_{10}
$\sum_{i=1}^n w(x_i, y_i) x_i^2$	$\sum_{i=1}^n w(x_i, y_i) x_i^3$	$\sum_{i=1}^n w(x_i, y_i) x_i^4$	$\sum_{i=1}^n w(x_i, y_i) x_i^3 y_i$	$\sum_{i=1}^n w(x_i, y_i) x_i^2 y_i^2$	c_{20}
$\sum_{i=1}^n w(x_i, y_i) x_i y_i$	$\sum_{i=1}^n w(x_i, y_i) x_i^2 y_i$	$\sum_{i=1}^n w(x_i, y_i) x_i^3 y_i$	$\sum_{i=1}^n w(x_i, y_i) x_i^2 y_i^2$	$\sum_{i=1}^n w(x_i, y_i) x_i y_i^3$	c_{11}
$\sum_{i=1}^n w(x_i, y_i) y_i^2$	$\sum_{i=1}^n w(x_i, y_i) x_i y_i^2$	$\sum_{i=1}^n w(x_i, y_i) x_i^2 y_i^2$	$\sum_{i=1}^n w(x_i, y_i) x_i y_i^3$	$\sum_{i=1}^n w(x_i, y_i) y_i^4$	c_{02}

(9)

$\sum_{i=1}^n w(x_i, y_i) Z$
$\sum_{i=1}^n w(x_i, y_i) x_i Z$
$\sum_{i=1}^n w(x_i, y_i) y_i Z$
$\sum_{i=1}^n w(x_i, y_i) x_i^2 Z$
$\sum_{i=1}^n w(x_i, y_i) x_i y_i Z$
$\sum_{i=1}^n w(x_i, y_i) y_i^2 Z$

Observemos que a matriz quadrada também é simétrica. Para chegar às equações, aqui também, como no caso anterior existe uma regra mnemônica.

Tomando o polinômio na sua forma original $Z = c_{00} + c_{10} x + c_{01} y + c_{20} x^2 + c_{11} xy + c_{02} y^2$ e transformando-o na forma matricial temos:

$$(1 \quad x \quad y \quad x^2 \quad xy \quad y^2) \begin{bmatrix} c_{00} \\ c_{10} \\ c_{01} \\ c_{20} \\ c_{11} \\ c_{02} \end{bmatrix} = Z(x,y) \quad (10)$$

Então, multiplicando a equação 2.5 pelo seu primeiro elemento, que é 1, e aplicando a $\sum_{i=1}^n w(x_i, y_i)$ obtemos a primeira equação do sistema 7:

$$\sum_{i=1}^n w(x_i, y_i) c_{00} + c_{10} \sum_{i=1}^n w(x_i, y_i) x_i + c_{01} \sum_{i=1}^n w(x_i, y_i) y_i + c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^2 +$$

$$+ c_{11} \sum_{i=1}^n w(x_i, y_i) x_i y_i + c_{02} \sum_{i=1}^n w(x_i, y_i) y_i^2 = \sum_{i=1}^n w(x_i, y_i) Z(x_i, y_i)$$

Para obter a segunda equação do sistema 2.3, e multiplicar a equação 10 pelo seu segundo elemento que é x e aplicar a $\sum_{i=1}^n w(x_i, y_i)$

$$c_{00} \sum_{i=1}^n w(x_i, y_i) x_i + c_{10} \sum_{i=1}^n w(x_i, y_i) x_i^2 + c_{01} \sum_{i=1}^n w(x_i, y_i) x_i y_i + c_{20} \sum_{i=1}^n w(x_i, y_i) x_i^3 +$$

$$+ c_{11} \sum_{i=1}^n w(x_i, y_i) x_i^2 y_i + c_{02} \sum_{i=1}^n w(x_i, y_i) x_i y_i^2 = \sum_{i=1}^n w(x_i, y_i) x_i Z(x_i, y_i)$$

Procedendo da mesma forma, acharemos qualquer equação do sistema 8.

Essa regra pode ser aplicada a um polinômio de qualquer grau e serve tanto para o método de mínimos quadrados, como para o método de mínimos quadrados ponderados.

4. IMPLEMENTAÇÃO

O algoritmo de mínimos quadrados ponderados consiste de uma sub-rotina, que será chamada do programa principal, no momento da interpolação.

O objetivo do algoritmo é determinar o valor de Z no ponto (a,b) , mas antes determinando os coeficientes do sistema 8.

Aqui segue uma explicação mais detalhada do algoritmo. A listagem da "procedure" de MÍNIMOS QUADRADOS PONDERADOS está anexada, no Apêndice I.

real procedure mínimos quadrados ponderados;

O algoritmo usa uma matriz e (6×6) que serve para formar a matriz principal da equação 9 e um vetor $v(6)$ que contém os valores do vetor resultante da equação 9.

Primeiro, o algoritmo zera a matriz e e o vetor v

Para $i = 1$ até 6 $v = 0$;

Para $i = 1$ até 6

Para $j = 1$ até 6 $e = 0$;

A partir disso, o algoritmo formará a matriz principal através dos pontos x,y,z .

Para $i = 1$ até n

início

$t_{\text{term}} = w(x, y)$

O w é uma função peso que é calculada numa sub-rotina e a variável t_{term} tem o valor de w .

$x \cdot t_{\text{term}} = x * t_{\text{term}}$

$y \cdot t_{\text{term}} = y * t_{\text{term}}$

$x^2 \cdot t_{\text{term}} = x^2 * t_{\text{term}}$

$y^2 \cdot t_{\text{term}} = y^2 * t_{\text{term}}$

$xy \cdot t_{\text{term}} = x * y * t_{\text{term}}$

O algoritmo multiplica x , y , x^2 , xy pela variável t_{term} para diminuir o tempo de máquina.

Nessa parte, o algoritmo preenche a matriz e . Como a matriz é simétrica, o algoritmo coloca os valores obtidos somente nos lugares a apropriados e então fará a obtenção do vetor v .

$z \cdot t_{\text{term}} = z * t_{\text{term}}$

$v[1] = v[1] + z \cdot t_{\text{term}}$

$v[2] = v[2] + x * t_{\text{term}}$

$v[3] = v[3] + y * t_{\text{term}}$

$v[4] = v[4] + x^2 * t_{\text{term}}$

$v[5] = v[5] + x * y * t_{\text{term}}$

$v[6] = v[6] + y^2 * t_{\text{term}}$

fim;

Neste ponto, o algoritmo já tem, através dos n pontos de x , y e z , o sistema completo. Como no bloco acima a matriz e só foi formada por alguns elementos, agora os demais elementos da matriz serão preenchi dos.

Para $i = 1$ até 5

Para $j = i + 1$ até 6 e $|j,i| = e |i,j|$

Para resolver o sistema, já existe no INPE uma rotina LINEQNIMPRV, que resolve, aplicando o método de Gauss, as seis equações lineares fornecendo as seis incógnitas, no caso os seis coeficientes no vetor coef $|1, \dots, 6|$.

Para calcular z num ponto (a,b) :

$$\begin{aligned} \text{mínimos quadrados ponderados} = & \text{coef } |1| + q * (\text{coef } |2| + b * \text{coef } |4| + \\ & + a * \text{coef } |5|) + b * (\text{coef } |3| + b * \text{coef } |6|) \end{aligned}$$

5. FUNÇÃO PESO

w é uma função peso que depende de x e y . Há flexibilidade na escolha da função w , dando o peso de um ponto como uma função de quadrado da sua distância ao ponto sendo considerado.

O uso do simples $1/d^2$, ou, para evitar "overflow", aritmético, $(1/(d^2 + \epsilon))$ para ϵ pequeno, provou não ser satisfatório. Isto talvez ocorra por terem os pontos remotos de (a,b) muita influência. As funções mais rapidamente decrescentes, como $w = 1/(d^2 + \epsilon)^4$, tem mostrado dar resultados mais precisos. O mesmo acontece com as funções decrescentes exponenciais $w = \exp(-\alpha d^2)$ para um α constante. Se os pontos estão sujeitos a um erro experimental, o uso de tal função exponencial, com um valor pequeno de α , talvez leve à suavização da superfície. Contudo, se os dados são exatos, quando (a,b) está muito próximo a algum (x_i, y_i) deveríamos esperar que a suavização dominasse absolutamente; então uma função como $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$ é geralmente melhor e é mais recomendada.

Damos, a seguir uma explicação mais detalhada de como a função peso foi usada no programa. A listagem está anexada, no Apêndice II.

A função peso foi colocada como uma sub-rotina, que é chamada pelo "procedure" de mínimos quadrados ponderados, como vimos no item 3.

Para usarmos a função $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$ temos que definir α . Como w depende de x e y , o α também deve ter uma relação com x e y . Portanto o α foi definido como sendo o inverso da média aritmética da soma das distâncias quadráticas dos pontos.

Tendo α este valor, a função peso w influencia bem melhor a determinação dos coeficientes, porque α depende da densidade dos pontos ou seja, o número dos pontos que estão bem próximos, ao ponto escolhido. O valor de α com maior densidade influi na função w , dando assim maior peso aos pontos próximos ao ponto escolhido.

A determinação de α é feita por uma subrotina:

```
real procedure  $\alpha$ 
 $\alpha = 0$ ;
 $d = 0$ 
Para  $i = 1$  até  $n$ 
  início
 $d = \sqrt{(a-x)^2 + (b-y)^2}$ 
 $\alpha = \alpha + d$ ;
  fim
 $\alpha = \alpha/n$ ;
 $\alpha = 1/\alpha$ ;
```

Para implantar esta sub-rotina, usamos uma outra variável β , como mostrado na listagem, porque o compilador do computador B-6700 acusa um erro devido ao fato de que α aparenta ser recursiva.

Definimos outra sub-rotina w , que é a função de distâncias, e α , que é o inverso da média aritmética das distâncias quadráticas dos pontos.

real procedure w

$$w = \exp(-\alpha d^2)/(d^2 + \epsilon);$$

Observa-se que o d^2 é a distância $d^2 = (a-x)^2 + (b-y)^2$.

Também testamos os programas com α sendo a média aritmética das distâncias quadráticas dos pontos. Os resultados com a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$ com este α foram bons, mas com o outro α os resultados foram melhores porque a função peso depende, automaticamente, da densidade dos pontos.

Em geral, o "procedure" mínimos quadrados ponderados foi usado com quatro funções peso.

- i) $w = 1/(d + \epsilon)^2$, onde d é a distância dos pontos.
- ii) $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$, onde d é a distância dos pontos e α , a média aritmética das distâncias quadráticas.
- iii) $w = \exp(-\alpha d^2)/(d^2 + \epsilon)^2$, onde d é a distância dos pontos e α , a média aritmética das distâncias quadráticas.
- iv) $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$, onde d é a distância dos pontos e α , o inverso da média aritmética das distâncias quadráticas.

As listagens das quatro funções peso estão anexadas, no A pêndice II.

6. EQUAÇÕES TESTADAS

Testamos algumas equações usando as quatro funções peso. Verificamos que com as tres últimas funções peso, as equações mostraram resultados melhores que aqueles obtidos pela primeira função peso.

Na maioria dos casos, os valores de x e y foram obtidos a través da função RANDOM, que fornece valores entre 0 e 1. O número de pontos escolhidos foi 500.

Dependendo da equação da curva, escolhemos uma região própria para a interpolação. Foram impressos os valores interpolados ($\ell = m$ mínimos quadrados ponderados), valores reais ($m =$ equação da curva nos pontos interpolados), a diferença entre o valor interpolado e o valor real, ($n = \ell - m$) e o erro ($0 = n/m$).

Depois de termos verificado a melhoria nos resultados reais as tres últimas funções peso, fizemos o seguinte, para melhor visualizar o resultado: calculamos o erro (0) e extraímos a sua potencia de 10 ($\log_0 = \log(0)$). Definimos nove matrizes. Se a potencia de 10 (\log_0) fosse 1 então a primeira matriz seria preenchida de "*****" e o contador de primeira matriz teria o valor 1. O mesmo acontece com o resto das matrizes, acumulando nos contadores o número de vezes que a potencia ocorreu. Se a potencia for 0 ou maior que 9, o programa continuará normalmente sem preencher matrizes. Assim, no final do programa, fazemos imprimir o resultado dos contadores para sabermos quantas vezes um erro ocorreu; por exemplo, se o contador 3 tiver o número 7, entenderemos que o erro 10^{-3} ocorreu 7 vezes. Mais ainda, o programa imprime as matrizes para vermos o mapeamento da região.

Em geral foram plotados 500 pontos interpolados sobre 100 pontos.

6.1. PARTE DE UMA ESFERA

A equação da curva é:

$$(x - 5,5)^2 + (y - 5,5)^2 + z^2 = 65$$

o primeiro passo do programa principal é buscar os 500 pontos de x e y e, através de x e y, chegarmos a z.

```
Para i = 1 até 500
    início

    x |i| = 11 * random;

    y |i| = 11 * random;

    z |i| =  $\sqrt{64 - ((x-5,5)^2 + (y-5,5)^2)}$  ;

    fim;
```

O uso do fator 11 para multiplicar a função RANDOM é por que esta função nos dá valores entre 0 e 1 e o raio da equação é 8,0. Então, x e y estão limitados, e fora desse limite a esfera não é definida. Se colocarmos valores fora desse limite, poderão ocorrer erros, por exemplo, divisão por zero.

Depois que x, y e z foram plotados, o programa faz interpolação nos 100 pontos. A região escolhida para interpolação foi

```
Para i = 1 até 10
    Para j = 1 até 10
        início

        l = mínimos quadrados ponderados (i, j, x, y, z, 500);

        m =  $\sqrt{64 - ((i-5,5)^2 + (j-5,5)^2)}$  ;

        n = l - m;

        O = n/m;

        write (l,m,n,O);

    fim;
```

Ainda existe o mapeamento citado no início deste capítulo, e que não está descrito acima.

Usando a primeira função peso $w = 1/(d-\epsilon)^2$, os resultados obtidos foram satisfatórios. Os erros, da ordem de 10^{-4} e 10^{-5} estão bem concentrados. Mas nas três outras funções peso, os resultados melhoraram, tanto α sendo a média aritmética da distância quadrática dos pontos como sendo o inverso da média aritmética da distância quadrática dos pontos, os erros sendo concentrados da ordem de 10^{-4} e 10^{-5} .

6.2. SUBIDA INGREME SAINDO DE UM PLANO

A equação da curva é:

$$z = \exp - ((x-5)^2 + (y-5)^2)$$

Essa equação foi testada em três regiões. Usando $w(d) = 1/(d-\epsilon)^2$. A primeira região apresentou melhores resultados, mostrando que a interpolação foi melhor executada.

O mesmo aconteceu usando as funções peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$ e $w = \exp(-\alpha d^2)/(d^2 + \epsilon)^2$, com α sendo a média aritmética da distância quadrática dos pontos. Com essas duas funções peso, a média do erro foi de 10^{-1} , embora aparecendo alguns erros de 10^{-2} e 10^{-3} .

Usando a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$ com α sendo o inverso da média aritmética da distância quadrática dos pontos, a primeira região apresentou resultados melhores do que as das outras duas regiões. O erro de 10^{-1} foi o que mais ocorreu, mas erros de 10^{-2} e 10^{-3} também ocorreram.

As regiões testadas foram:

Primeira:

Para $i = 4,5$ de $0,1$ até $5,5$

Para $j = 4,5$ de $0,1$ até $5,5$

Segunda:

Para $i = 0,1$ de $0,1$ até 1

Para $j = 0,1$ de $0,1$ até 1

Terceira:

Para $i = 1$ até 10

Para $j = 1$ até 10

6.3. SUBIDA MENOS INGREME

A equação da curva é

$$z = \exp - \left(\frac{1}{4} (x-5)^2 + (y-5)^2 \right)$$

A equação também foi testada nas mesmas regiões que anteriormente. A função peso $w(d) = 1/(d-\epsilon)^2$ não apresentou resultados satisfatórios.

Primeiro discutiremos os resultados usando α como média aritmética das distâncias quadráticas dos pontos:

Na primeira região, com a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$, a concentração de erros foi em volta de 10^{-2} , embora se apresentasse um bom número de erros da ordem de 10^{-4} . Com a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)^2$ houve uma distribuição melhor, embora a concentração tenha sido em erros da ordem de 10^{-2} .

Na segunda região, com a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$, a concentração foi em erros da ordem de 10^{-3} , de onde concluímos que, nessa região, o algoritmo interpola melhor. Com a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)^2$ a concentração foi em erros da ordem de 10^{-4} , o que melhorou mais ainda a interpolação.

Na terceira região, com a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$, o erro de 10^{-3} , foi o que mais ocorreu. O mesmo aconteceu usando a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)^2$.

Discutiremos, a seguir, os resultados obtidos usando α como sendo o inverso da média aritmética das distâncias quadráticas dos pontos:

Na primeira região, os erros que mais ocorreram foram da ordem de 10^{-2} mas também aparecendo os erros de 10^{-3} .

Na segunda região a interpolação foi bem melhorada, aparecendo o maior número de vezes os erros de 10^{-3} . Essa foi a melhor região para a interpolação.

Na terceira região, os resultados foram satisfatórios, ocorrendo mais os erros de 10^{-2} . Mas ocorreram também alguns erros de 10^{-1} e 10^{-3} .

6.4. UM PLANALTO E UM PLANO SEPARADOS POR UMA PAREDE INGREME

A equação dessa curva é

$$z = \tan h (x + y) + 2$$

A equação foi testada com a função peso $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$, com α sendo o inverso da média aritmética da distância quadrática dos pontos.

Nesta, os pontos não foram gerados pela função RANDOM.

Para $i = 1$ até 10

Para $j = 1$ até 10

início

```
ll = ll + 1;  
x|ll| = i;  
y|ll| = j;  
z|ll| = tanh(x |ll| + y|ll|) + 2;  
  
fim;
```

A interpolação foi feita na região:

```
Para i = 0 até 10  
Para j = 0 até 10
```

Os resultados foram bons, mas depois de algumas saídas o correu o erro de "INVALID ALOG ARGUMENT" no cálculo de $\log_0 = \log(0)$. En quanto o erro (0) foi impresso os erros eram de 10^{-1} , 10^{-2} , 10^{-3} , 10^{-4} , 10^{-5} , inclusive chegou-se até a 10^{-11} . Logo depois deste último, ocorreu o erro de argumento invalido do logaritmo, porque o erro (0) realmente foi zero, ou seja, a interpolação caiu em cima do ponto, de onde concluímos que o teste foi muito bom.

6.5. RELEVO DE TESTE

Este teste foi muito importante para o presente trabalho porque se trata de uma superfície irregular e na prática, encontraremos muitas superfícies desse tipo. O trabalho foi feito, realmente, para que as superfícies irregulares sejam bem interpoladas.

Os dados de x e y não foram gerados pela função RANDOM, mas sim lidos. Tínhamos 10.000 pontos.

Foram criados x e y através de 10.000 pontos e através de x e y foi criado z .

```
Para i = 1 de 2 até 100  
Para j = 1 de 2 até 100  
início
```

```
ll = ll + 1;  
x[ll] = i;  
y[ll] = j;  
z[ll] = ZREAL [i,j] ; onde ZREAL contém os 10.000 pontos lidos  
  
fim;
```

Foi feita a interpolação nos pontos alternados.

```
Para i = 2 de 2 até 100  
Para j = 2 de 2 até 100  
  início
```

```
  ll = mínimos quadrados ponderados (i,j,x,y,z,2500);
```

```
  m = ZREAL [i,j] ;
```

```
  n = ll - m;
```

```
  O = n/m;
```

```
  write (ll,m,n,O);
```

```
  fim;
```

Devemos observar que a interpolação foi feita em 2500 pontos e não em 100 pontos, como nas outras superfícies. A função peso usada para essa superfície foi $w = \exp(-\alpha d^2)/(d^2 + \epsilon)$, com α sendo o inverso da média aritmética da distância quadrática dos pontos.

Os resultados foram muito bons. A maioria se fixou em erros da ordem de 10^{-2} , também apresentou alguns erros da ordem de 10^{-3} .

Essa superfície também foi testada por outra função peso, $w = 1/(d - \epsilon)^2$. Usando esta função, os resultados foram satisfatórios. Neste caso os valores de x e y foram criados através da função RANDOM. Primeiro foram lidos os pontos e depois foram obtidos x, y e z.

Para i = 1 até 500

início

x |i| = 100*RANDOM;

y |i| = 100*RANDOM;

z |i| = ZREAL |x |i| ,y |i||; onde ZREAL contém os 10.000 pontos li
dos

fim;

A interpolação foi feita em 100 pontos com a região

Para i = 1 até 10

Para j = 1 até 10

Aparecem erros de 10^{-1} e 10^{-2} , mas apresentando também er
ros de 10^{+1} , isso porque os pontos aproximados, com o erro relativo maior
que 10^{-1} , são da borda, uniformizada para 10, e altamente descontínua.

7. CONCLUSÕES

Os resultados obtidos, em média, através desta rotina de
interpolação por MINIMOS QUADRADOS PONDERADOS foram muito bons. O uso ime
diato desta rotina será na plotagem de curvas de contorno, já que a inter
polação através deste trabalho é graficamente aceitável. É perfeita para
visualização, inclusive servindo para visualizar superfícies em tres di
mensões.

Como dito no início deste trabalho, nosso objetivo era pro
duzir rotinas para interpolação aplicáveis à regiões com superfícies irre
gulares. Na prática, usaremos este trabalho para cartas náuticas e mapas
de menor escala.

Nossa futura pesquisa, que começará em breve, será o estudo da interpolação por séries de Fourier, e a comparação dos respectivos resultados com aqueles obtidos por mínimos quadrados ponderados.

O produto final, a longo prazo, será um "pacote" para traçado de cartas convencionais, perspectivas e outras, capaz de receber e tratar convenientemente, a grande parte dos dados científicos deste Instituto, bem como aqueles referentes a cartas geográficas e de navegação.

No Apêndice C estão anexadas as listagens do programa principal e os resultados, como exemplo, da equação da parte de uma esfera. Também aparece na listagem como foram impressas as matrizes A1 até A9.

AGRADECIMENTOS

Ao Dr. Celso de Renna e Souza pela oportunidade que me deu para fazer este trabalho de graduação.

A Mag Pacheco pela colaboração na redação do T.G..

Ao José Antonio Gonçalves Pereira pela sua valiosa orientação sem a qual o trabalho não teria sido realizado.

REFERENCIAS BIBLIOGRÁFICAS

- LI, JEROME C.R. - Least Squares, In: *Numerical Mathematics*. Michigan, Edwards Brothers Inc., 1966. 493-499.
- MCLAIN, D.H. - Drawing contours from arbitrary data points. *The Computer Journal*. London, England 17(4): 318-324, Dec. 1971.
- SPIEGEL, MURRAY R. - Curve fitting and the method of Least Squares, In: *Statistics*. New York, McGraw-Hill, 1961. 217-240.

APÊNDICE A


```

43  P1,AL1E = + V1*AL1E
44  P1,AL1E = + V1*AL1E
45  P1,AL1E = + V1*AL1E
46  P1,AL1E = + V1*AL1E
47  P1,AL1E = + V1*AL1E
48  P1,AL1E = + V1*AL1E
49  P1,AL1E = + V1*AL1E
50  P1,AL1E = + V1*AL1E
51  P1,AL1E = + V1*AL1E
52  P1,AL1E = + V1*AL1E
53  P1,AL1E = + V1*AL1E
54  P1,AL1E = + V1*AL1E
55  P1,AL1E = + V1*AL1E
56  P1,AL1E = + V1*AL1E
57  P1,AL1E = + V1*AL1E
58  P1,AL1E = + V1*AL1E
59  P1,AL1E = + V1*AL1E
60  P1,AL1E = + V1*AL1E
61  P1,AL1E = + V1*AL1E
62  P1,AL1E = + V1*AL1E
63  P1,AL1E = + V1*AL1E
64  P1,AL1E = + V1*AL1E
65  P1,AL1E = + V1*AL1E
66  P1,AL1E = + V1*AL1E
67  P1,AL1E = + V1*AL1E
68  P1,AL1E = + V1*AL1E
69  P1,AL1E = + V1*AL1E
70  P1,AL1E = + V1*AL1E
71  P1,AL1E = + V1*AL1E
72  P1,AL1E = + V1*AL1E
73  P1,AL1E = + V1*AL1E
74  P1,AL1E = + V1*AL1E
75  P1,AL1E = + V1*AL1E
76  P1,AL1E = + V1*AL1E
77  P1,AL1E = + V1*AL1E
78  P1,AL1E = + V1*AL1E
79  P1,AL1E = + V1*AL1E
80  P1,AL1E = + V1*AL1E
81  P1,AL1E = + V1*AL1E
82  P1,AL1E = + V1*AL1E
83  P1,AL1E = + V1*AL1E
84  P1,AL1E = + V1*AL1E
85  P1,AL1E = + V1*AL1E
86  P1,AL1E = + V1*AL1E
87  P1,AL1E = + V1*AL1E
88  P1,AL1E = + V1*AL1E
89  P1,AL1E = + V1*AL1E
90  P1,AL1E = + V1*AL1E
91  P1,AL1E = + V1*AL1E
92  P1,AL1E = + V1*AL1E
93  P1,AL1E = + V1*AL1E
94  P1,AL1E = + V1*AL1E
95  P1,AL1E = + V1*AL1E
96  P1,AL1E = + V1*AL1E
97  P1,AL1E = + V1*AL1E
98  P1,AL1E = + V1*AL1E
99  P1,AL1E = + V1*AL1E
100 P1,AL1E = + V1*AL1E
101 P1,AL1E = + V1*AL1E

```

APENDICE B

1. The first step is to identify the problem or question that needs to be answered. This involves understanding the context and the specific requirements of the task.

```

1  DO I=1,NPTS
2  DO J=1,NPTS
3  IF (I.EQ.J) GO TO 10
4  X1=X(I)
5  Y1=Y(I)
6  X2=X(J)
7  Y2=Y(J)
8  DIST=SQRT((X1-X2)**2+(Y1-Y2)**2)
9  DIST**2=1.0/DIST**2
10  DIST**2=1.0/DIST**2
11  DIST**2=1.0/DIST**2
12  DIST**2=1.0/DIST**2
13  DIST**2=1.0/DIST**2
14  DIST**2=1.0/DIST**2
15  DIST**2=1.0/DIST**2
16  DIST**2=1.0/DIST**2
17  DIST**2=1.0/DIST**2
18  DIST**2=1.0/DIST**2
19  DIST**2=1.0/DIST**2
20  DIST**2=1.0/DIST**2
21  DIST**2=1.0/DIST**2
22  DIST**2=1.0/DIST**2

```



```

1  SUBROUTINE ALFA(XPTS,YPTS,NPTS)
2  INTEGER NPTS; REAL XPTS; ARRAY X,Y(*)
3  REAL
4  INTEGER I; REAL D;
5  REAL XPTS;
6  DATA D;
7  D=1.0;
8  DO I=1,NPTS
9  D=1.0+D*(XPTS(I)-XPTS(1))**2 + (YPTS(I)-YPTS(1))**2;
10 D=SQRT(D);
11 D=1.0/D;
12 D=1.0/D;
13 D=1.0/D;
14 D=1.0/D;
15 D=1.0/D;
16 D=1.0/D;
17 D=1.0/D;
18 D=1.0/D;
19 D=1.0/D;
20 D=1.0/D;
21 D=1.0/D;
22 D=1.0/D;
23 D=1.0/D;

```

APENDICE C


```

1  ARRAY X(.7),Z(.500);
2  REAL I,J,X,L,M,N,O;
3  ARRAY A1,A2,A3,A6,A5,A6,A7,A8,A9(0:11,0:11);
4  ARRAY C(1:A(0:10));
5  INTEGER I1,J1;
6  INTEGER LOGO;
7  FILE P(ATTN=PRINT);
8  FOR I=0 STEP 1 UNTIL 11 DO
9      FOR J=0 STEP 1 UNTIL 11 DO
10         BEGIN
11             A(I1,J1)=0;
12             A(11,J1)=0;
13             A(11,J1)=0;
14             A(11,J1)=0;
15             A(11,J1)=0;
16             A(11,J1)=0;
17             A(11,J1)=0;
18             A(11,J1)=0;
19             A(11,J1)=0;
20         END;
21     END;
22     X(I2)=19-15;
23     FOR I1=1 STEP 1 UNTIL 500 DO
24         BEGIN
25             X(I1)=1 + RANDOM(K);
26             Y(I1)=11 + RANDOM(K);
27             Z(I1)=SQRT(A4-((X(I1)-5.5)**2+(Y(I1)-5.5)**2));
28         END;
29     END;
30     I1=1;
31     FOR I1=1 STEP 1 UNTIL 10 DO
32         BEGIN
33             J1=1;
34             FOR J1=1 STEP 1 UNTIL 10 DO
35                 BEGIN
36                     L=INT(SQRT(A4)*COS(ACOS(I,J,X,Y,Z,500)));
37                     M=SQRT(A4-((I-5.5)**2+(J-5.5)**2));
38                     N=1;
39                     IF M=0 THEN O=1;
40                     ELSE O=N/M;
41                     LOGO=ENTER(ABS(LOG(ABS(O)))));
42                     CASE LOGO OF
43                         BEGIN
44                             BEGIN
45                                 A(I1,J1)=*****;
46                                 C(1:A(10:10)):=* + 1;
47                             END;
48                         BEGIN
49                             BEGIN
50                                 A(I1,J1)=*****;
51                                 C(1:A(10:10)):=* + 1;
52                             END;
53                         END;
54                     END;
55                 END;
56             END;
57         END;
58     END;
59     END;
60     END;
61     END;
62     END;
63     END;
64     END;
65     END;
66     END;
67     END;
68     END;
69     END;
70     END;
71     END;
72     END;
73     END;
74     END;
75     END;
76     END;
77     END;
78     END;
79     END;
80     END;
81     END;
82     END;
83     END;
84     END;
85     END;
86     END;
87     END;
88     END;
89     END;
90     END;
91     END;
92     END;
93     END;
94     END;
95     END;
96     END;
97     END;
98     END;
99     END;
100    END;

```

52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105

4. 3336884631E+00	4. 3474798574E+00	-1. 339011801465E+02	-2. 8861619450E+00
5. 6123746680E+00	5. 6123746680E+00	-4. 3879924646E+03	-1. 3381733469E+03
6. 4220493634E+00	6. 4220493634E+00	-4. 5860215255E+03	-7. 48849493255E+04
6. 5958529791E+00	6. 5958529791E+00	-2. 8293737052E+03	-4. 3842784274E+04
6. 5958529791E+00	6. 5958529791E+00	-4. 5176908569E+03	-6. 8497048970E+04
6. 5958529791E+00	6. 5958529791E+00	-1. 3222940783E+03	-2. 0048984722E+04
6. 4204936344E+00	6. 4204936344E+00	-0. 7456237505E+03	-1. 3626740101E+04
6. 1237243370E+00	6. 1237243370E+00	-8. 6417370767E+04	-1. 44111397553E+04
5. 6124866002E+00	5. 6124866002E+00	-1. 9192312480E+02	-3. 8195741791E+03
4. 8476798374E+00	4. 8476798374E+00	-1. 54669634337E+02	-3. 3277208050E+03
5. 6124866002E+00	5. 6124866002E+00	-1. 9387693770E+02	-3. 3543560766E+03
6. 2849025450E+00	6. 2849025450E+00	-7. 3685111689E+02	-1. 1717821282E+03
7. 0356236397E+00	7. 0356236397E+00	-1. 7580795102E+03	-2. 6063504711E+04
7. 1733500472E+00	7. 1733500472E+00	-3. 6052703682E+03	-5. 1375974376E+04
7. 1733500472E+00	7. 1733500472E+00	-6. 8206877108E+05	-9. 5046746130E+06
7. 1733500472E+00	7. 1733500472E+00	-7. 63543552313E+05	-1. 8639719330E+05
7. 0356236397E+00	7. 0356236397E+00	3. 10787644005E+04	4. 4713431661E+05
6. 7453687816E+00	6. 7453687816E+00	-1. 1765384537E+02	-1. 7442166497E+03
6. 2684925450E+00	6. 2684925450E+00	-2. 6792362837E+02	-4. 2630315029E+04
5. 6124866002E+00	5. 6124866002E+00	-1. 8329932684E+02	-3. 2658122644E+03
6. 1237243370E+00	6. 1237243370E+00	-8. 9944100345E+03	-1. 4468781661E+03
4. 7453687816E+00	4. 7453687816E+00	-6. 5315583265E+04	-9. 7126726251E+05
7. 1733500472E+00	7. 1733500472E+00	4. 24674742146E+03	5. 9173715593E+04
7. 4983221299E+00	7. 4983221299E+00	-2. 0829831192E+04	-2. 9025562147E+05
7. 1733500472E+00	7. 1733500472E+00	4. 0777118446E+03	6. 0443354548E+04
6. 1237243370E+00	6. 1237243370E+00	-1. 4007765938E+02	-2. 2874591705E+03
6. 4070493639E+00	6. 4070493639E+00	-3. 3574140525E+02	-5. 2171713219E+04
7. 0356236397E+00	7. 0356236397E+00	-1. 0609137005E+03	-1. 5079163627E+04
7. 4983221299E+00	7. 4983221299E+00	-1. 8995603713E+03	-2. 5498025224E+04
7. 1733500472E+00	7. 1733500472E+00	1. 3557648053E+03	1. 5756233592E+04
7. 8421935707E+00	7. 8421935707E+00	4. 6072840632E+04	5. 8749940584E+05
7. 8421935707E+00	7. 8421935707E+00	6. 06562729879E+03	7. 3399907139E+04
7. 1733500472E+00	7. 1733500472E+00	4. 3248201747E+03	4. 4399384901E+04
7. 8421935707E+00	7. 8421935707E+00	-2. 5467115648E+03	-3. 4184851824E+04
7. 9686887253E+00	7. 9686887253E+00	7. 59400482153E+04	1. 0793781909E+04
7. 8421935707E+00	7. 8421935707E+00	-1. 2559353266E+03	-1. 57608482971E+04
7. 5629754001E+00	7. 5629754001E+00	6. 98546774495E+03	8. 7599554668E+05
7. 1733500472E+00	7. 1733500472E+00	1. 9763957640E+03	2. 6063935453E+05
6. 5958529791E+00	6. 5958529791E+00	1. 2166005472E+03	1. 8952915329E+04
6. 5958529791E+00	6. 5958529791E+00	-3. 4837582490E+03	-5. 2820062269E+04
6. 5958529791E+00	6. 5958529791E+00	-1. 0535528127E+03	-1. 5973926526E+04
7. 1733500472E+00	7. 1733500472E+00	1. 9558831846E+03	2. 7254567738E+04
7. 5829754001E+00	7. 5829754001E+00	4. 2539566468E+04	5. 6117474219E+05
7. 8421935707E+00	7. 8421935707E+00	1. 4595566468E+03	1. 8611741183E+04
7. 9686887253E+00	7. 96868		

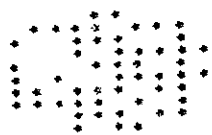
7.0357581205+00
7.45175675645+00
7.7132319101E+00
7.6433101301E+00
7.8442052555E+00
7.7136695323E+00
7.4495717533E+00
7.23209109E+00
6.4350935702E+00
6.123308124E+00
6.7584717504E+00
7.1763138915E+00
7.450657232E+00
7.5846326295E+00
7.5674980050E+00
7.452708345E+00
7.1757858021E+00
6.7441370083E+00
5.154153022E+00
5.598153920E+00
6.2789442765E+00
6.743571438E+00
7.0287897230E+00
7.1748738678E+00
7.1754752548E+00
7.0335155102E+00
6.7429002258E+00
5.5335493295E+00
4.803393979E+00
5.6032573363E+00
6.1241778141E+00
6.4421955635E+00
6.593752707E+00
6.5878941076E+00
4.4385277716E+00
5.1161044797E+00
5.3953345377E+00
4.8032170328E+00
0.
4.

7.0356236397E+00
7.4498322129E+00
7.7136243103E+00
7.8421933570E+00
7.8421933570E+00
7.7136243103E+00
7.4498322129E+00
7.0356236397E+00
6.4420493634E+00
6.1237243570E+00
6.7453687816E+00
7.1763500472E+00
7.4498322129E+00
7.5828754441E+00
7.5828754441E+00
7.4498322129E+00
7.1763500472E+00
6.7453687816E+00
6.1237243570E+00
5.6124860802E+00
6.2849025450E+00
6.7453687816E+00
7.0356236397E+00
7.1763500472E+00
7.1763500472E+00
7.0356236397E+00
6.7453687816E+00
6.2849025450E+00
5.6124860802E+00
4.847698524E+00
5.6124860802E+00
6.1237243570E+00
6.4420493634E+00
6.593752707E+00
6.5878941076E+00
6.4420493634E+00
6.1237243570E+00
5.6124860802E+00
4.847698524E+00
0.
0.

1.3117313210E+04
1.9265405608E+03
6.0767073592E+04
1.1245594214E+03
2.0149552584E+03
1.0866220769E+03
2.6045757113E+04
2.3232288077E+03
6.9557936002E+03
4.195464903E+04
5.1029387456E+03
3.4155687899E+05
8.2551031664E+04
1.7570187920E+03
6.235609079E+04
2.8767216281E+03
5.6464510271E+04
2.7682268809E+03
9.5650548174E+03
1.4376688150E+02
5.9582684626E+03
1.7116377858E+03
6.8338467099E+03
1.4761794300E+03
8.7475236796E+04
2.1081290179E+03
2.4681558571E+03
1.6415431310E+03
1.8936750697E+02
4.488595535E+02
9.2287438601E+03
4.7345712665E+04
1.4720010586E+04
1.7002734239E+03
7.5588715263E+03
3.5215917887E+03
7.6198772877E+03
2.7151542498E+02
4.4462824677E+02
22.
0.

1.8644137154E+05
2.5833381398E+04
7.477886783E+05
1.4330857991E+04
2.5693771012E+04
1.3827768030E+04
3.4961535198E+05
3.3020336403E+04
1.0797485088E+03
6.8511347764E+05
7.5651441913E+04
4.7594790771E+06
1.1080924953E+04
2.3170877656E+04
8.2232788259E+05
3.8614562794E+04
7.8681376844E+05
4.1038522712E+04
1.5619669110E+03
2.5526455644E+03
9.4802877529E+04
2.5375006782E+04
9.7132047601E+04
2.0570358878E+04
1.2189377082E+04
2.9963641119E+04
3.6590376849E+04
2.6118831903E+04
3.3740396724E+03
9.1354959151E+03
1.6443236969E+03
7.7315225024E+05
2.2849894121E+05
2.5779479125E+04
1.1460731432E+03
5.4665706383E+04
1.2443207505E+03
4.8377033118E+03
9.1719804081E+03
20.
0.

- 0.7 -



- C.8 -

•
•
•

