MÉTODOS NUMÉRICOS PARA AJUSTE DE CURVAS

Evandro Pedro Alves de Mendonçaa, Marcelino José de Lima Andradea.

a Núcleo de Tecnologia (NTI), Universidade Federal de Pernambuco (UFPE), Campus Acadêmico do Agreste (CAA), Rodovia BR-104, km 59, S/N, Nova Caruaru, CEP. 55.014-900, Caruaru-PE, Brasil, <http://www.ufpe.br/caa>

**Palavras Chave:** ajuste de curvas, resíduos, polinômios, aproximação, dados experimentais, regressão linear.

**Resumo**: Em diversas situações no cotidiano científico, é necessário encontrar uma fórmula matemática que represente um conjunto finito de pontos. Este procedimento pode ser realizado através do ajuste de curvas, nele é possível realizar aproximações para um conjunto de dados ou uma função através de diversas funções simples. Neste trabalho serão exploradas as aproximações lineares através de uma reta (polinômio de primeiro grau), linearização de funções não lineares e, por fim, de um polinômio de segundo grau (cujo procedimento pode ser estendido para polinômios de graus superiores).

1. INTRODUção

Muitos estudos e observações são feitos a partir de experimentos, onde os resultados obtidos são analisados e guardados. Algumas vezes, esses dados são armazenados quase que continuamente fazendo com que seja fácil a dedução de alguma função que represente bem aquele conjunto de dados. Porém, geralmente não há espaço suficiente para guardar dados contínuos ou simplesmente não é possível obter tais dados de forma contínua. Por isso é necessário gravar um conjunto de pontos discretos.

Portanto, é preciso definir uma função, ou seja, uma curva que defina esses pontos da melhor forma possível, pois, a partir de uma equação geral fica mais fácil determinar a solução de determinado problema. Existem diversas formas de determinar uma curva para certo conjunto de dados como a interpolação e o ajuste de curvas.

Nesse trabalho é usado um método de ajuste de curvas, o método dos mínimos quadrados, que busca diminuir a diferença entre os valores reais e os valores aproximados obtidos por uma função ajustada aos pontos. Os erros, ou resíduos, são definidos como uma função, que é minimizada por meio de derivadas parciais. Mais detalhes sobre o método serão abordados na discussão dos resultados.

1. Exercícios Propostos

Segue, abaixo, a solução dos exercícios propostos sobre o tema.

* 1. 1ª questão

Dados:

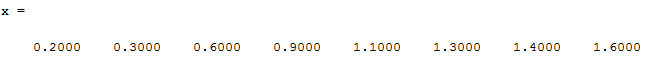


Figura 1: Vetor linha que contém todos os pontos no eixo x usados para realizar as aproximações por funções.

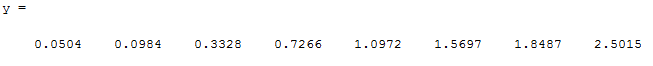


Figura 2: Vetor linha que contém os valores de Y para cada respectivo X.

* + 1. **1.a)**
* Aproximando os pontos dados nos vetores X e Y para uma função do tipo:

Usando o MMQ (Método dos Mínimos Quadrados – Anexo 1), temos:

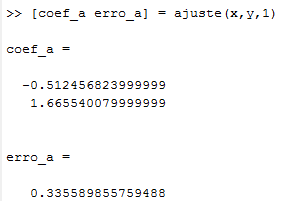


Figura 3: coeficientes a e b (de cima para baixo) da curva linear e o erro calculado.



Figura 4: gráfico contendo os pontos (x,y) e a curva linear obtida que se ajusta a esses pontos.

* Aproximando os pontos dados nos vetores X e Y para uma função do tipo:

Usando o MMQ, temos:

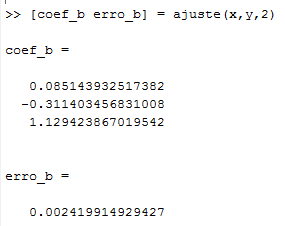


Figura 5: coeficientes a, b e c (de cima para baixo) da curva quadrática e o erro calculado.



Figura 6: gráfico contendo os pontos (x,y) e a curva quadrática obtida que se ajusta a esses pontos.

* Aproximando os pontos dados nos vetores X e Y para uma função do tipo:

Usando o MMQ, temos:

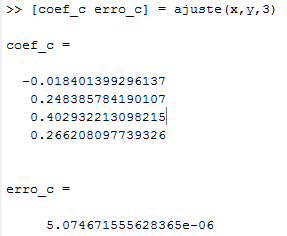


Figura 7: coeficientes a, b, c e d (de cima para baixo) da curva cúbica e o erro calculado.



Figura 8: gráfico contendo os pontos (x,y) e a curva cúbica obtida que se ajusta a esses pontos.

* Aproximando os pontos dados para uma função do tipo:

Para poder utilizar o MMQ nesta equação, que não é linear, precisamos linearizá-la. Para isso, fazemos:

Assim, temos uma função linear

Onde

Usando o MMQ, temos:

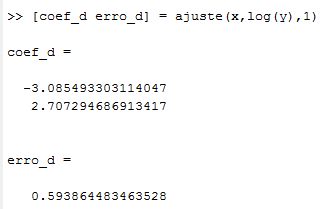


Figura 9: coeficientes b1 e a (de cima para baixo) da curva y = b1+ax e o erro calculado.

Para descobrir o valor do coeficiente b, usamos a equação .

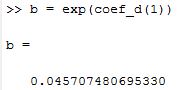


Figura 10: valor do coeficiente b.

A seguir, será mostrado o gráfico da curva original com os coeficientes encontrados.

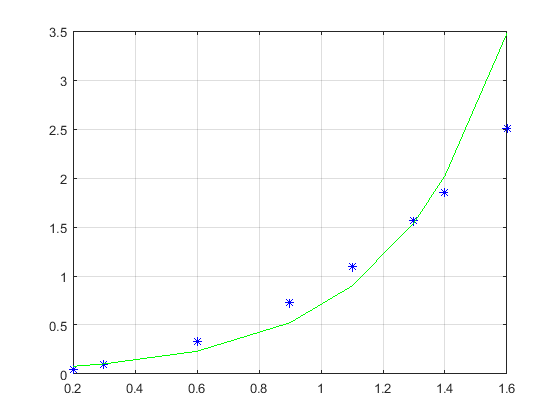


Figura 11: gráfico contendo os pontos (x,y) fornecidos (em azul) e a função não linear que se ajusta aos pontos fornecidos (em verde).

* Aproximando os pontos dados para uma função do tipo:

Aqui também realizamos a linearização:

Assim, temos uma função linear

Onde

Usando o MMQ, temos:

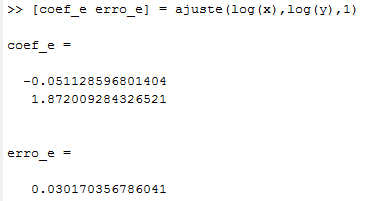


Figura 12: coeficientes b1 e a (de cima para baixo) da curva y = b1+ax1 e o erro calculado.

Para descobrir o valor do coeficiente b, usamos a equação .

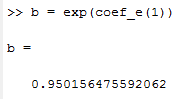


Figura 13: valor do coeficiente b.

A seguir, será mostrado o gráfico da curva original com os coeficientes encontrados.



Figura 14: gráfico contendo os pontos (x,y) fornecidos (em azul) e a função não linear que se ajusta aos pontos fornecidos (em verde).

* + 1. **1.b)**

Conforme o item anterior, vimos que o valor do erro foi mostrado para cada caso. O caso que apresentou menor erro foi o da aproximação por função polinomial cúbica, com erro da ordem de 10-6. O erro é calculado pelo quadrado da diferença entre o valor da imagem de um ponto e a imagem do mesmo ponto feito pela função de aproximação. Detalhes sobre como esse cálculo foi feito no código, vide Anexo 1.

* + 1. **1.c)**

Usando a função *polyfit* do MATLAB®, obtivemos resultados semelhantes aos encontrados com a função *ajustes*. A seguir, vemos uma tabela que compara os tempos de processamento das duas funções.

|  |  |  |
| --- | --- | --- |
| Função | *ajustes* | *polyfit* |
| Linear | 0,102323 | 0,016814 |
| Quadrática | 0,105399 | 0,007898 |
| Cúbica | 0,116782 | 0,051067 |
| Exponencial em e | 0,108034 | 0,078759 |
| Exponencial em x | 0,183830 | 0,005055 |

Tabela 1: tempo de processamento das funções de ajustes de curvas, em segundos.

Os valores obtidos pela função *polyfit* podem ser vistos no Anexo 4. Quanto ao tempo de processamento, devemos levar em conta que a função *ajustes* realizava também a plotagem a curva gerada pelo algoritmo, o que não acontece na função *polyfit*. Por isso, tem maior tempo de processamento. Porém, mesmo assim, a função *polyfit* tem tempo muito menor que *ajustes*. O que já era esperado, por ser uma função bastante otimizada pela equipe de desenvolvimento do MATLAB®. É claro que, para essa situação, não faz a menor diferença, esse ganho de tempo de processamento, mas para um volume de dados maior, fará sim toda a diferença.

* 1. 2ª questão
  2. 3ª questão
     1. **– 3.a)**

Aproximando por uma reta. Pelo Método dos Mínimos Quadrados, na aproximação por um polinômio do primeiro grau, temos um sistema linear da seguinte forma.

Onde

Calculando os produtos internos, temos:

Substituindo na equação matricial, temos:

Resolvendo o sistema no MATLAB®:

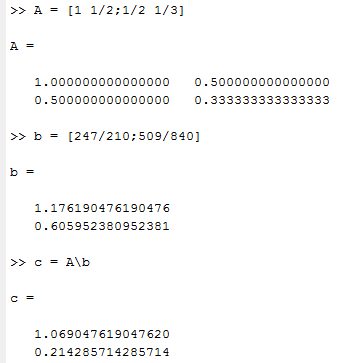


Figura 15: solução do sistema linear por matriz inversa.

Logo, a função linear que procuramos é

Cujo gráfico é



Figura 16: gráfico da função linear g(x) que é ajustada à função f(x) no intervalo de 0 a 1.

* + 1. **– 3.b)**

Aproximando por um polinômio do segundo grau. Pelo Método dos Mínimos Quadrados, na aproximação por um polinômio do segundo grau, temos um sistema linear da seguinte forma.

Onde

Calcularemos apenas os itens que não foram calculados no sistema anterior.

Substituindo na equação matricial, temos:

Resolvendo o sistema no MATLAB®:

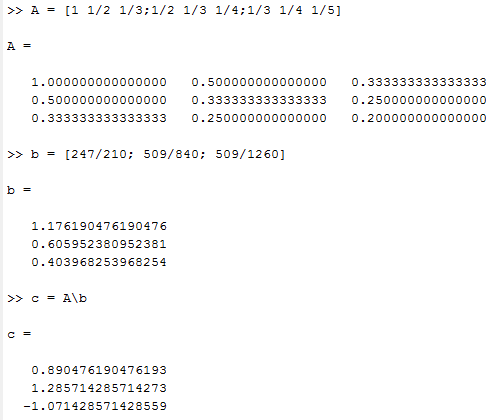


Figura 17: solução do sistema linear por matriz inversa.

Logo, a função quadrática que procuramos é

Assim, temos os gráfico:



Figura 18: gráfico da função quadrática h(x) que é ajustada à função f(x) no intervalo de 0 a 1.

1. conclusão

Ao longo do trabalho, diversas formas de regressão linear foram empregadas para realizar o ajuste de curvas. No seu caso mais simples, aproximando uma função por uma reta (ou polinômio de primeiro grau). Para aproximar por funções que não são lineares, o método pode ser aplicado desde que se reescreva a função em questão numa forma linear, como no caso da função exponencial, que teve que ser expressa como uma função logarítmica para que fosse encontrada a relação linear e, assim, usar a regressão linear. Também foi utilizada a função polinomial de segunda ordem para realizar esse mesmo procedimento, onde foi aplicada uma técnica análoga ao polinômio do primeiro grau e que pode ser estendida para polinômios de ordens superiores. Há que se ter um cuidado, contudo, com polinômios de ordens muito grandes, pois, como foi mencionado no trabalho, estes podem até passar por todos os pontos do conjunto dado, mas o erro relativo entre esses pontos torna-se grande ao ponto de não serem representativos.

O que há de comum entre todos esses métodos é o fato de que eles partem do mesmo princípio: o método dos mínimos quadrados. Isso se faz através do cálculo do erro entre a função de aproximação e a função original (ou conjunto de pontos). Quando o erro é expresso por uma equação quadrática, podemos encontrar valores em que o erro calculado será mínimo. Em termos matemáticos, isso significa que as derivadas parciais dessa função com relação às suas variáveis é nula. Assim, podem ser encontradas expressões que, combinadas, formam um sistema linear que, por sua vez, encontradas as soluções deste, são encontrados os valores dos coeficientes das funções de aproximação. Além disso, como o erro calculado nesses pontos é mínimo, garante-se que esta é a melhor solução.

Portanto, O ajuste de curvas se mostra uma ferramenta muito útil para achar fórmulas matemáticas para um conjunto finito de pontos que seja representativa e com um erro relativo pequeno.

REFERÊNCiaS

Chapra, S. C., e Canale, R. P. *Métodos Numéricos para Engenharia*. 5ª edição. Porto Alegre: AMGH, 2011.

Gilat, A., e Subramaniam, V. *Métodos Numéricos para Engenheiros e Cientistas: uma introdução com aplicações usando o MATLAB*. Porto Alegre: Bookman, 2008.

anexo 1

função para calcular os minimos quadrados

anexo 2

anexo 3