



CENTRO UNIVERSITÁRIO FEI
Leonardo Contador Neves – 118315-1

Tópicos Especiais em Aprendizagem:

Least Squares Method (Método dos Mínimos Quadrados)

São Bernardo do Campo, SP

2018

SUMÁRIO

1 INTRODUÇÃO.....	11
2 REVISÃO BIBLIOGRÁFICA.....	11
3 METODOLOGIA.....	11
4 DESENVOLVIMENTO.....	14
4.1 CLASSE PARA MANIPULAR MATRIZ.....	14
4.1.1 Método de Transposição.....	14
.....	14
4.1.2 Método de Cálculo de Determinante.....	14
4.1.3 Método de Cálculo de Menor da Matriz.....	15
4.1.4 Método de Cálculo de Inversão da Matriz.....	15
4.2 CLASSE DOS MÍNIMOS QUADRADOS.....	15
4.2.1 Método Ajuste de Curva.....	16
4.2.2 Método de Previsão de Valores.....	16
5 RESULTADOS.....	17
5.1 IMPLEMENTAÇÃO LINEAR.....	17
5.2 IMPLEMENTAÇÃO QUADRÁTICA.....	17
5.3 IMPLEMENTAÇÃO PLANAR.....	18
5 Conclusão.....	19
REFERÊNCIAS.....	19

1 INTRODUÇÃO

Este relatório busca a implementação do algoritmo de Mínimos Quadrados implementado com alguns parâmetros visando a partir de amostras de dados, gerar uma equação que pode descrever com um erro aceitável o seu conjunto de dados, podendo ser usada para prever valores novos, com base em dada entrada.

2 REVISÃO BIBLIOGRÁFICA

O Método dos Mínimos Quadrados é uma técnica de otimização matemática que procura encontrar o melhor ajuste para um conjunto de dados tentando minimizar a soma dos quadrados das diferenças entre o valor estimado e os dados observados (tais diferenças são chamadas resíduos). É a forma de estimação mais amplamente utilizada na econometria. Consiste em um estimador que minimiza a soma dos quadrados dos resíduos da regressão, de forma a maximizar o grau de ajuste do modelo aos dados observados.

Um requisito para o método dos mínimos quadrados é que o fator imprevisível (erro) seja distribuído aleatoriamente e essa distribuição seja normal. O Teorema Gauss-Markov garante (embora indiretamente) que o estimador de mínimos quadrados é o estimador não-enviesado de mínima variância linear na variável resposta. Outro requisito é que o modelo é linear nos parâmetros, ou seja, as variáveis apresentam uma relação linear entre si. Caso contrário, deveria ser usado um modelo de regressão não-linear.

Credita-se Carl Friedrich Gauss como o desenvolvedor das bases fundamentais do método dos mínimos quadrados, em 1795, quando Gauss tinha apenas dezoito anos. Entretanto, Adrien-Marie Legendre foi o primeiro a publicar o método em 1805, em seu *Nouvelles méthodes pour la détermination des orbites des comètes*. Gauss publicou suas conclusões apenas em 1809.

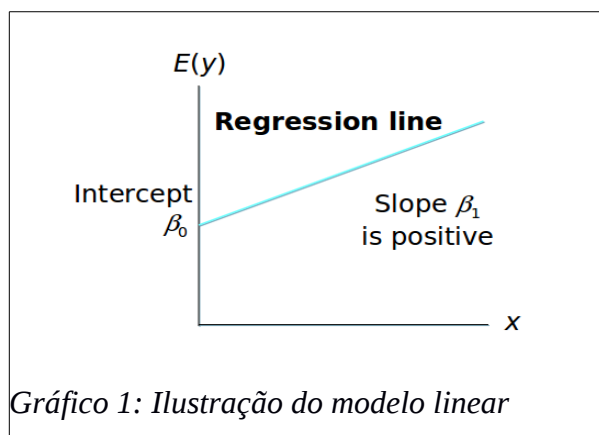
3 METODOLOGIA

Para implementar o Método dos Mínimos Quadrados, vamos nos embasar à principio na equação de modelos lineares descrita à baixo:

Equação 1: Modelo Linear

$$\hat{Y} = X^T \hat{\beta}$$

Na equação, as variáveis vetoriais são denominadas como o vetor de saída Y , o vetor de entrada transposto X e o vetor de pesos $Beta$. Podemos ainda acrescentar ao lado direito da equação o vetor “Beta zero”, referente à intersecção do eixo da imagem com a função linear. Com base na equação linear então, podemos chegar ao seguinte gráfico:



Temos agora que achar um modo de ajustar o modelo linear ao meu conjunto e informações de entrada da minha amostra. O método para esse ajuste, que é denominado mínimos quadrados, busca minimizar o erro quadrado do conjunto de entrada (Soma do erro ao quadrado residual) à minha reta. Para começarmos, vamos primeiro obter a Soma dos erros ao quadrado pela seguinte equação:

Equação 2: Erro residual

$$RSS(\beta) = \sum_{i=1}^N (y_i - x_i^T \beta)^2.$$

Então, adotamos como critério o mínimo da somatória dos meus dados da amostra menos meus dados previstos pelo modelo linear, como mostra a seguinte equação:

Equação 3: Mínimo da somatória do erro ao quadrado

$$\min \sum (y_i - \hat{y}_i)^2$$

Transmormando a equação de soma dos erros residuais ao quadrado na notação matricial, temos que:

Equação 4: Forma matricial do erro residual

$$RSS(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta)$$

Para encontrarmos o mínimo da equação da soma dos erros residuais, vamos aplicar a derivada e igualar à zero, encontrando assim o mínimo da função, como é mostrado à seguir:

Equação 5: Derivada da soma residual dos erros ao quadrado.

$$\frac{\partial \text{RSS}}{\partial \beta} = -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta)$$

Equação 6: Derivada do erro igualada a zero

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0$$

A partir disso, basta agora isolar o componente *Beta* da equação a cima que nos diz o vetor de coeficientes do nosso modelo, como mostra na equação à baixo:

Equação 7: Vetor de coeficientes do modelo

$$\hat{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

Com os coeficientes da equação, para encontrar novos valores basta agora, pegar nosso vetor de coeficientes *Beta* e multiplicarmos pelo vetor de entrada. Com isso, computamos um vetor de saída com os valores ajustados à nossa equação.

Equação 8: Equação para encontrar novos valores à partir dos coeficientes descobertos

$$\hat{y}(x_0) = x_0^T \hat{\beta}$$

4 DESENVOLVIMENTO

Para implementar o Método dos Mínimos quadrados, foram feitas à princípio duas Classes na linguagem de programação *Python*. Uma das classes foram desenvolvidos métodos para manipulação de matrizes como Transposição, Determinante, Menores da Matriz e o Método de Inversão. A segunda classe é a principal, onde implementamos o método dos Mínimos Quadrados em dois métodos, o de Ajuste de curva (fit) e o de Previsão de valores (Predict).

4.1 CLASSE PARA MANIPULAR MATRIZ

Classe para o auxílio dos metodos aqui implementados para ajuste de curva, contendo metodos de Transposição, Determinante, Menores da Matriz e o Método de Inversão descritos à seguir.

4.1.1 Método de Trasposição

Metodo que devolve a transposta de uma entrada matricial.

```
def transpM(m):  
    return map(list,zip(*m))
```

4.1.2 Método de Cálculo de Determinante

Método que devolve o determinante de uma entrada matricial, onde resume à um retorno simples para o caso de uma matriz 2x2 e usa recursão para entradas de n linhas e n colunas maiores.

```
def gMatrixDeterminant(m):  
    #base case for 2x2 matrix  
    if len(m) == 2:  
        return m[0][0]*m[1][1]-m[0][1]*m[1][0]  
  
    determinant = 0  
    for c in range(len(m)):  
        determinant += ((-1)**c)*m[0][c]*gMatrixDeterminant(gMMin(m,0,c))  
    return determinant
```

4.1.3 Método de Cálculo de Menor da Matriz

Método que retorna o menor da matriz à partir de uma entrada matricial e de o valor da linha e da coluna do número que será descontado para o calculo.

```
def gMMin(m,i,j):  
    return [row[:j] + row[j+1:] for row in (m[:i]+m[i+1:])]
```

4.1.4 Método de Cálculo de Inversão da Matriz

Método que retorna a matriz inversa à partir de uma entrada matricial. Este método usa os métodos anteriores e faz um cálculo simples de inversão caso a matriz for 2 x 2 ou um cálculo mais complexo usando o método do Menor já descrito.

```
def getMatrixInverse(m):  
    determinant = gMatrixDeterminant(m)  
    #special case for 2x2 matrix:  
    if len(m) == 2:  
        return [[m[1][1]/determinant, -1*m[0][1]/determinant],  
                [-1*m[1][0]/determinant, m[0][0]/determinant]]  
  
    #finding matrix of cofactor  
    cofac = []  
    for r in range(len(m)):  
        cofactorRow = []  
        for c in range(len(m)):  
            minor = gMMin(m,r,c)  
            cofactorRow.append((-1)**(r+c) * gMatrixDeterminant(minor))  
        cofac.append(cofactorRow)  
    cofac = list(transpM(cofac))  
    for r in range(len(cofac)):  
        for c in range(len(cofac)):  
            cofac[r][c] = cofac[r][c]/determinant  
    return cofac
```

4.2 CLASSE DOS MÍNIMOS QUADRADOS

Essa classe tem dois métodos, o método *Fit*, responsável por aplicar as teorias de ajuste de curva e o método *Predict*, responsável por prever novos valores com base nos coeficientes da equação ajustada.

4.2.1 Método Ajuste de Curva

Esse método tem como entrada o conjunto de amostra das informações e através da implementação da equação de ajuste de curva dos mínimos quadrados, descobre o valor dos coeficientes através do vetor *Beta*. Essa função tem como parâmetros também o valor do bias e os pesos dos coeficientes (vetor *W*) para melhorar o ajuste de curva às informações.

```
def fit(self, X, Y, W=[], bias=0):  
  
    #Checking weight  
    if len(W) == 0:  
        W = np.ones([len(np.array(X).T),1])  
    else:  
        W = np.array(W).T  
  
    if bias != 0:  
        self.X = np.append(np.ones([len(np.array(X).T),1])*bias, np.array(X).T, axis=1)  
    else:  
        self.X = np.array(X).T  
  
    self.Y = np.array(Y).T  
    #transposing X ...  
    self.Xt = (self.X).T  
    #doing Xt * W * X  
    XtX = (self.Xt).dot(W*self.X)  
  
    #doing Xt * W * y  
    XtY = (self.Xt).dot(W*self.Y)  
  
    #Inverting (Xt * W * X)  
    invM = np.array(mt.getMatrixInverse(XtX.tolist()))  
  
    #doing (XtX)-1 * XtY  
    self.beta = invM.dot(XtY)
```

4.2.2 Método de Previsão de Valores

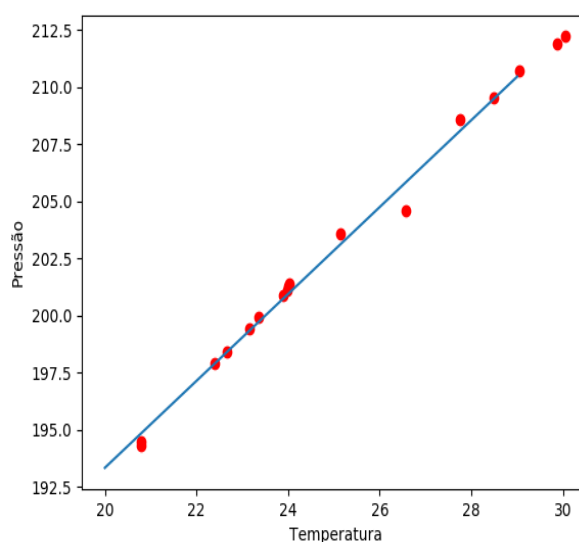
Esse método, tem como entrada o valor de *Xo* a ser previsto e sua saída é o valor *Yo* que representa ao ajuste à curva. Para a previsão, o valor do vetor *Beta* é usado, com base no método anterior.

```
def predict(self, X):  
    #Predicting values  
    Xt = np.array(X)  
    #doing Xt * beta  
    y = Xt.dot(self.beta)  
    print(y)  
    return y
```


5 RESULTADOS

5.1 IMPLEMENTAÇÃO LINEAR

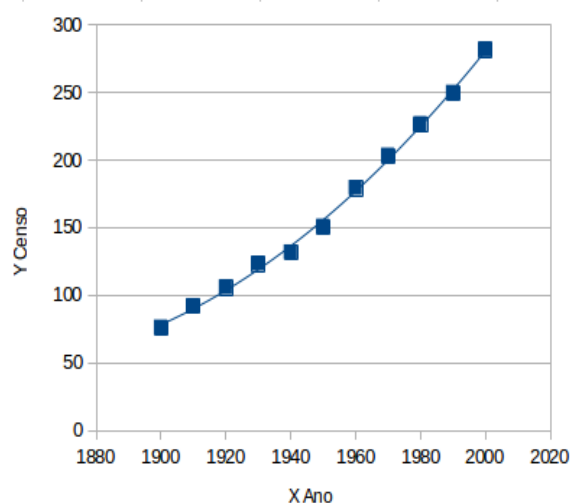
Com base no conjunto de informações da base “alpewater.xlsx”, foram usados os métodos de “fit” da classe dos Mínimos Quadrados, onde a entrada foi a pressão (valor de X) e seu respectivo valor de temperatura (valor de Y). Assim, pudemos obter os valores de beta e a curva ajustada aos dados como mostra na figura à seguir.



Usando o método “*predict()*” podemos então ver novos valores de Y (pressão) à partir do valor de entrada X (Temperatura). No exemplo, foi usado o valor de Temperatura **10** e o retorno foi **174.31**, ajustado à curva.

5.2 IMPLEMENTAÇÃO QUADRÁTICA

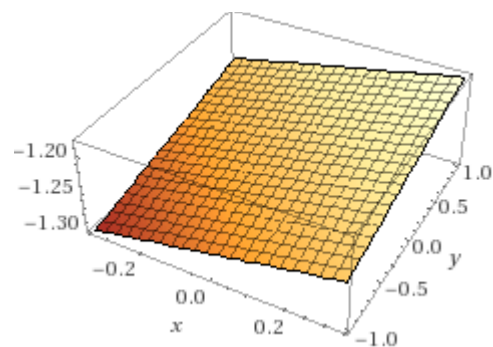
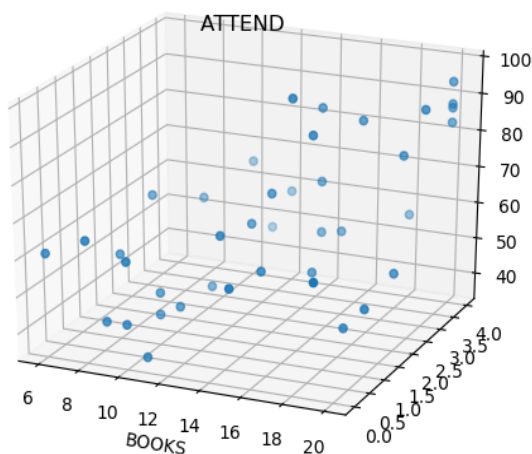
Com base no conjunto de informações da base “USCensus.xls”, foram usados os métodos de “fit” da classe dos Mínimos Quadrados, onde as entradas foram o ano (valor de X) e o valor do ano ao quadrado (valor de X^2) e seus respectivos valores de Censo (valor de Y). Assim, pudemos obter os valores de beta e a curva ajustada aos dados como mostra na figura à seguir.



Usando o método “*predict()*” podemos então ver novos valores de Y (Censo) à partir dos valores de entrada X (Ano) e X^2 . No exemplo, foi usado o valor de Ano **1919** e o retorno foi **102.23**, ajustado à curva quadrática.

5.3 IMPLEMENTAÇÃO PLANAR

Com base no conjunto de informações da base “Books_attend_grade.xls”, foram usados os métodos de “fit” da classe dos Mínimos Quadrados, onde as entradas foram a Quantidade de livros lidos (valor de X) e o valor da Presença do aluno (valor de Y) e seus respectivos valores de Nota (valor de Z). Assim, pudemos obter os valores de beta e o plano ajustado aos dados como mostra na figura à seguir.



Usando o método “*predict()*” podemos então ver novos valores de Y (Nota) à partir dos valores de entrada X (Quantidade de livros lidos) e Y (Presença). No exemplo, foi usado o valor de **1** para os livros lidos, **20** para a presença e o retorno foi **67.08** de nota, ajustado à curva quadrática.

5 CONCLUSÃO

Com base na implementação do algoritmo de Mínimos Quadrados e na análise dos resultados em diversas aplicações, foi visto que se faz necessário para alguns casos o uso do parâmetro “W” descrito no trabalho, para corrigir a função ajustada à amostra. Foi notado também que, dada as funções de inversão e multiplicação de matrizes, uma amostra com um número elevado, pode resultar em um esforço computacional muito grande para achar o vetor de coeficientes para ajuste da função à amostra, sendo assim é recomendável que nem todos os dados sejam usados para a determinação da função de ajuste, mas que poucas amostras também podem não descrever muito bem o comportamento dos dados.

REFERÊNCIAS

MÉTODO DOS MÍNIMOS QUADRADOS. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2018. Disponível em: <https://pt.wikipedia.org/w/index.php?title=M%C3%A9todo_dos_m%C3%A9nimos_quadrados&oldid=51604442>. Acesso em: 24 mar. 2018.

MATRIZ INVERSA. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2018. Disponível em: <https://pt.wikipedia.org/w/index.php?title=Matriz_inversa&oldid=52683977>. Acesso em: 17 jul. 2018.