Tutorial: Regressao Polinomial no Matlab/Octave

Prof. Dr. Guilherme de Alencar Barreto

Maio /2014

Departmento de Engenharia de Teleinformática (DETI)
Curso de Graduação em Engenharia de Teleinformática (CGETI)
Universidade Federal do Ceará (UFC), Campus do Pici, Fortaleza-CE

gbarreto@ufc.br

OBJETIVO - Usando as medidas de velocidade do vento (m/s) e de potência gerada (kWatts) para um gerador eólico disponibilizados no arquivo `aerogerador.dat`, determinar o melhor modelo de regressão polinomial para estes dados.

1 Implementação via Linguagem Script do Matlab

Passo 1 - Visualização dos dados.

```
>> load aerogerador.dat % carrega arquivo de dados
>> v=aerogerador(:,1); % medidas de velocidades
>> P=aerogerador(:,2); % medidas de potencia
>> figure; plot(v,P,'bo'); grid; % diagrama de dispersao
>> xlabel('Velocidade do vento [m/s]'); ylabel('Potencia gerada [kWatts]');
```

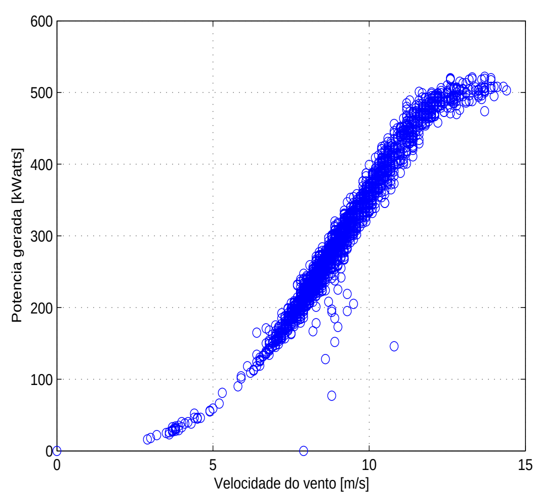

Figura 1: Diagrama de dispersão dos dados do aerogerador.

Observações 1

1. Uma análise da figura revela que a relação entre as duas variáveis é não-linear. Assim, não vale a pena avaliar um modelo de regressão linear simples.
2. Também não vale a pena avaliar um modelo polinomial de ordem 2 (ou seja, quadrático) porque percebemos duas concavidades ou curvaturas no gráfico, uma no início e outra no fim. Modelos quadráticos (i.e. parábolas) só possuem uma curvatura.
3. Portanto, vamos começar nossa análise com um modelo polinomial de ordem 3.

Passo 2 - Construção do modelo linear $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}$, em que

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}_{n\times 1}, \mathbf{X} = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^k \\ 1 & x_2 & x_2^2 & \cdots & x_2^k \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^k \end{bmatrix}_{n\times(k+1)} \quad \text{e} \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix}_{(k+1)\times 1}$$

Nas equações acima, o par (x_i, y_i), $i = 1, 2, ...n$, corresponde à i-ésima observação conjunta da velocidade do vento (x_i, variável de entrada) e da potência gerada (y_i, variável de saída) correspondente. Além disso, β_j, $j = 1, ..., k$, são os parâmetros a estimar, n é o total de observações e k é a ordem do polinômio.

```
>> k=3; % ordem inicial do polinomio
>> n=length(v); % numero de medidas
>> y=P; % vetor de observacoes da variavel de saida
>> X=[]; for l=1:k+1, X=[X v.^(l-1)]; end; % Constroi recursivamente a matriz X
```

Passo 3 - Estimação de $\boldsymbol{\beta}$ pelo método dos mínimos quadrados (MQ).

```
>> B=inv(X'*X)*X'*y % Formula de livro-texto para MQ
B =
32.6235
-58.7604
15.0519
-0.5924

>> B=X\y % Metodo alternativo via operador 'barra invertida'
B =
32.6235
-58.7604
15.0519
-0.5924
```

Passo 4 - Avaliação quantitativa do modelo por meio do coeficiente de determinação R^2:

$$R^2 = 1 - \frac{SQ_E}{S_{yy}} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{1}$$

```
>> ypred = X*B; % Predicao da saida para dados observados
>> erro=y-ypred; % Calcula erros
```

```
>> SEQ=sum(erro.^2); % Calcula a soma dos erros quadraticos
>> ymed=mean(y); % Calcula potencia media
>> Syy=sum((y-ymed).^2); % Soma dos erros para modelo baseado na media
>> R2 = 1 - SEQ/Syy
R2 =
0.9690
```

Passo 5 - Avaliação qualitativa do modelo (visual).

```
>> vv=min(v):0.1:max(v); vv=vv'; % Define faixa de valores para velocidade
>> XX=[]; for l=1:k+1, XX=[XX vv.^(l-1)]; end; % Monta nova matriz de regressao
>> ypred2=XX*B; % predicao da saida correspondente
>> hold on; plot(vv,ypred2,'r-'); hold off; % Sobrepoe curva de regressao aos dados
```

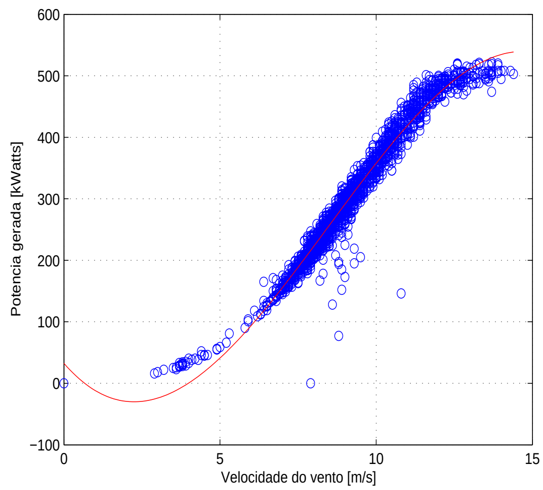

Figura 2: Curva de regressão sobreposta ao diagrama de dispersão dos dados do aerogerador.

Observações 2

1. Uma análise da curva de regressão no gráfico acima revela que o modelo de regressão polinomial de ordem 3 não é adequado, embora possua um coeficiente de determinação R^2 alto (i.e. próximo de 1).

2. O modelo de regressão polinomial de ordem 3 não é adequado porque para algumas velocidades, principalmente na faixa entre 0 e 5 m/s, o modelo prevê potências negativas, o que não é possível para um aerogerador. O modelo está dizendo que o aerogerador consome energia, em vez de gerá-la.

3. Devemos então repetir o procedimento de construção e avaliação do modelo polinomial para outros valores de k, por exemplo, $k = 4, 5, ...$, assim por diante até um valor máximo k_{max} pré-especificado pelo projetista.

4. Para cada valor de k devemos determinar o valor respectivo de $R^2(k)$ e o gráfico correspondente da curva de regressão resultante.

5. Deve-se escolher o menor valor de k que satisfaça as restrições de Engenharia do problema. No presente caso, não gerar energia com valores negativos.

2 Implementação via Funções Polyfit e Polyval

A seqüência de comandos mostrada anteriormente pode ser executada tanto em ambiente Matlab quanto no Octave. O valor da solução apresentada é principalmente didático, pois cada etapa foi discutida e realizada a partir das equações presentes na maioria dos livros-texto. Contudo, o procedimento de se obter e avaliar um modelo de regressão polinomial pode ser consideravelmente simplificado se optarmos por utilizar as funções `polyfit` e `polyval` presentes em ambas as plataformas de programação científica (Matlab e Octave). A nova seqüência de comandos é apresentada a seguir em uma só etapa. Os gráficos são omitidos por serem iguais aos mostrados anteriormente.

```
>> load aerogerador.dat % carrega arquivo de dados
>> v=aerogerador(:,1); % medidas de velocidades
>> P=aerogerador(:,2); % medidas de potencia
>> figure; plot(v,P,'bo'); grid; % diagrama de dispersao
>> xlabel('Velocidade do vento [m/s]'); ylabel('Potencia gerada [kWatts]');
>> k=3; % ordem inicial do polinomio
>> B = polyfit(v,P,k) % Estimacao do vetor de parametros
B =
   -0.5924 15.0519 -58.7604 32.6235
>> ypred=polyval(B,v); % Predicao da saida para dados observados
>> erro=y-ypred; % Calcula erros
>> SEQ=sum(erro.^2); % Calcula a soma dos erros quadraticos
>> ymed=mean(y); % Calcula potencia media
>> Syy=sum((y-ymed).^2); % Soma dos erros para modelo baseado na media
>> R2 = 1 - SEQ/Syy
R2 =
0.9690
>> vv=min(v):0.1:max(v); vv=vv'; % Define faixa de valores para velocidade
>> ypred2=polyval(B,vv); % predicao da saida correspondente
>> hold on; plot(vv,ypred2,'r-'); hold off; % Sobrepoe curva de regressao aos dados
```

Observações 3

1. Note que a ordem dos coeficientes (parâmetros) quando se usa a função `polyfit` está invertida em relação à obtida através da implementação baseada na equação do estimador dos mínimos quadrados.

3 Trabalho Computacional sobre Regressão Polinomial

Questão Única - Gerar um gráfico $R^2(k) \times k$ a fim de decidir pelo valor ótimo de k, ou seja, a fim de escolher a melhor ordem para o polinômio regressor. Em outras palavras, para cada valor de $k = 3, 4, ..., k_{max}$, determinar o valor correspondente de $R^2(k)$ e gerar o gráfico correspondente usando a função `plot` do Matlab/Octave. Em seguida, de posse do valor de k escolhido, gerar o gráfico de dispersão dos dados com a curva de regressão sobreposta a ele. Sugestão: Fazer $k_{max} = 15$.