Aula prática 4

Método dos mínimos quadrados

Lucas Emanuel Resck Domingues

15 de Maio de 2019

1. (a) Queremos aproximar uma reta $y = a + bx$. Considerando 1900 como $t = 0$, ficamos com o seguinte sistema:

$$\begin{cases} a + 20b = 54,1 \\ a + 30b = 59,7 \\ a + 40b = 62,9 \\ a + 50b = 68,2 \\ a + 60b = 69,7 \\ a + 70b = 70,8 \\ a + 80b = 73,7 \\ a + 90b = 75,4 \end{cases}$$

Temos as matrizes A e b:

$$A = \begin{bmatrix} 1 & 20 \\ 1 & 30 \\ 1 & 40 \\ 1 & 50 \\ 1 & 60 \\ 1 & 70 \\ 1 & 80 \\ 1 & 90 \end{bmatrix}, b = \begin{bmatrix} 54,1 \\ 59,7 \\ 62,9 \\ 68,2 \\ 69,7 \\ 70,8 \\ 73,7 \\ 75,4 \end{bmatrix}$$

A solução para o método dos mínimos quadrados, como visto em aula, é $\mathbf{x}$ tal que

$$A^T A\mathbf{x} = A^T b$$

Resolvendo computacionalmente (via Scilab), obtemos $\mathbf{x} = (a, b) \approx (50, 82; 0.29)$. Para $x = 100$ (referente ao ano 2000), obtemos $y = 79, 9$.

(b) A expectativa de vida nos Estados Unidos em 2000 foi $76, 64$. Isso mostra que nosso modelo não é muito preciso... Uma possível justificativa para isso é que o crescimento da população não é linear, como tentamos modelar.

2. (a) Queremos aproximar $s(t) = s_0 + v_0 t + \frac{1}{2}gt^2$. Formemos A e b:

$$A = \begin{bmatrix} 1 & 0,5 & 0,25 \\ 1 & 1 & 1 \\ 1 & 1,5 & 2,25 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{bmatrix}, \ b = \begin{bmatrix} 11 \\ 17 \\ 21 \\ 23 \\ 18 \end{bmatrix}$$

Resolvendo $A^T A\mathbf{x} = A^T b$, obtemos $\mathbf{x} = (s_0, v_0, \frac{g}{2}) \approx (1,92; 20,31; -4,97)$. Veja que $g \approx -9,94 \approx -9,81\text{m/s}^2$, realmente.

(b) $s_0 \approx 1,92$m, $v_0 \approx 20,31$m/s e $g \approx -9,94\text{m/s}^2$.

(c) Basta resolver $0 = s_0 + v_0 t + \frac{1}{2}gt^2$. Então $t \approx 4,18$s.

3. (a) Queremos aproximar $y = p(t) = ce^{kt}$. Podemos aproximar $\ln y = \ln c + kt$, um modelo linear. Então $\mathbf{x} = (\ln c, k)$. Calculamos b aplicando ln nas populações. Utilizando a função **Gaussian_Elimination_4**, da aula prática 1, para a resolução do sistema e considerando $t = 0$ para 1950, temos:

```
--> A
 A  =

   1.   0.
   1.   10.
   1.   20.
   1.   30.
   1.   40.
   1.   50.


--> b
 b  =

   5.0106353
   5.1873858
   5.313206
   5.42495
   5.5214609
   5.6383547


--> x = Gaussian_Elimination_4(A'*A, A'*b)
 x  =

   5.0455774
   0.0121502
```

Figura 1: Matrizes A e b e resolução do sistema $A^T A\mathbf{x} = A^T b$.

A partir disso, concluímos que $c \approx e^{5,04} \approx 155,33$ e $k \approx 0,01$. Portanto, $p(t) \approx 155,33e^{0,01t}$. Sendo assim, a taxa de crescimento é $p'(t) \approx 1,89e^{0,01t}$.

(b) A população estimada dos EUA em 2010 é $p(60) = ce^{k60} \approx 322,01$ milhões de pessoas. Em 2010, a população dos EUA foi de $309,3$ milhões de pessoas. Nosso modelo superestimou a população. Sabemos que um modelo exponencial também se adequa muito bem a um crescimento populacional de uma população já grande.

4. (a) Queremos aproximar $y = a + bx + cx^2$. Considerando 1970 como $t = 0$, temos:

```
--> A
 A  =

   1.   0.    0.
   1.   5.    25.
   1.   10.   100.
   1.   15.   225.
   1.   20.   400.
   1.   25.   625.
   1.   30.   900.
   1.   35.   1225.


--> b
 b  =

   29.3
   44.7
   143.8
   371.6
   597.5
   1110.8
   1895.6
   2476.6


--> x = Gaussian_Elimination_4(A'*A, A'*b)
 x  =

   57.0625
  -20.334643
   2.5886429
```

Figura 2: Matrizes A e b e resolução do sistema $A^T A\mathbf{x} = A^T b$.

Portanto, $y \approx 57,06 - 20,33x + 2,59x^2$

(b) Queremos $y = ce^{kt} \iff \ln y = \ln c + kt$. Aplicamos ln nas médias salariais e temos:

```
--> A2
 A2  =

   1.   0.
   1.   5.
   1.   10.
   1.   15.
   1.   20.
   1.   25.
   1.   30.
   1.   35.


--> b2
 b2  =

   3.3775875
   3.7999735
   4.9684234
   5.917818
   6.3927543
   7.0128358
   7.5472907
   7.8146419


--> x2 = Gaussian_Elimination_4(A2'*A2, A2'*b2)
 x2  =

   3.5037431
   0.1342956
```

Figura 3: Matrizes A e b e resolução do sistema $A^T A\mathbf{x} = A^T b$.

Ficamos com $y \approx e^{3,50} e^{0,13t} \approx 33,12 e^{0,13t}$.

(c) O erro é dado por $e = b - Ax$. No caso do modelo exponencial, como aplicamos ln, devemos antes calcular os exponenciais dos valores em b e em Ax. Calculamos as normas dos erros dos modelos quadrático e exponencial, respectivamente:

```
--> e1 = norm(b - A*x)
 e1  =

   174.19173


--> e2 = norm(exp(b2) - exp(A2*x2))
 e2  =

   1201.5096
```

Figura 4: Normas dos erros dos modelos quadrático e exponencial, respectivamente.

Vemos que o modelo quadrático se ajustou melhor aos dados.

(d) Utilizando a aproximação quadrática e calculando computacionalmente: em 2010, estimamos um salário de $3385,5$ milhares de dólares e, em 2015, $4384,0$.

5. No Scilab, lemos o arquivo **cancer_train.csv** utilizando a função **csvRead()**. Seja $\mathbf{w} = (c_0, c_1, \ldots, c_{10})$. Formamos a matriz A com as dez primeiras colunas de *cancer_train* (com os vetores $\mathbf{x}$ das características dos pacientes), adicionando uma primeira coluna com elementos 1. b é a matriz com a última coluna de *cancer_train*. Para encontrar o hiperplano que melhor se ajusta aos dados, resolvemos o sistema $A^T A\mathbf{w} = A^T b$ computacionalmente:

```
--> A(1:5, 1:5)
 ans  =

   1.   0.64     0.2643   0.6515   0.4006
   1.   0.7318   0.4524   0.705    0.5306
   1.   0.7005   0.541    0.6897   0.4814
   1.   0.4063   0.5188   0.4116   0.1545
   1.   0.7218   0.3651   0.7167   0.519


--> b(1:5)
 ans  =

   1.
   1.
   1.
   1.
   1.


--> w = Gaussian_Elimination_4(A'*A, A'*b)
 w  =

  -6.7579731
   29.311052
   2.0765803
  -18.730222
  -7.3665161
   1.2222756
   0.2283419
   0.0503253
   2.2385058
   0.0249405
   0.7704282
```

Figura 5: Matrizes A e b e vetor $\mathbf{w}$

Observe que o elemento z_i de $A\mathbf{w}$ é resultado do produto escalar de A_i (linha i de A) por $\mathbf{w}$. z_i indica se a paciente i tem ($z_i \geq 0$) ou não tem ($z_i < 0$) câncer de mama. Se multiplicarmos esse valor pelo resultado que já sabemos (1 ou -1), saberemos se o modelo acertou: se o produto for positivo, os dois valores têm mesmo sinal e o modelo acertou; caso contrário, o modelo errou. Ainda mais, podemos calcular o produto escalar entre $A\mathbf{w}$ e b, contar os acertos e calcular a porcentagem de acertos do modelo.

Para nosso arquivo de treinamento, o modelo obteve 93% de acertos:

```
--> result1 = A*w.*b;

--> size(result1(result1 >= 0))/size(result1)
 ans  =

   0.9300008
```

Figura 6: Cálculo da porcentagem de acertos do modelo para o arquivo de treinamento.

Ou seja, nosso modelo se ajusta bem aos dados de treinamento.

Agora vamos ver se o modelo prevê bem para os dados de testes:

```
--> result2 = A2*w.*b2;

--> size(result2(result2 >= 0))/size(result2)
 ans  =

   0.7115427
```

Figura 7: Cálculo da porcentagem de acertos do modelo para o arquivo de testes.

Obtemos 71% de acertos. É um resultado razoável. Isso mostra que o modelo tem uma boa capacidade de generalização.

Vamos observar agora alguns resultados do modelo:

	Treinamento	Testes
Verdadeiros positivos	135 (45%)	60 (23%)
Verdadeiros negativos	144 (48%)	125 (48%)
Falsos positivos	10 (3,33%)	75 (28,8%)
Falsos negativos	11 (3,67%)	0 (0%)

Tabela 1: Resultados do modelo.

Nos resultados para os dados de treinamento, observamos que tanto os acertos quanto os erros não dependem muito se a paciente tem ou não câncer de mama. Além disso, o modelo obtém uma razoável razão de acertos.

Porém, para os dados de testes, obtemos algo curioso: 0 falso negativo. Isso é, para 0 paciente com câncer de mama o modelo apontou que não havia câncer. E isso sem que o modelo apontasse todas as pacientes com câncer de mama, ou seja, sem que ele tivesse uma quantidade exorbitante de falsos positivos.

Na verdade, a razão de falsos positivos aumentou bastante em relação aos dados de testes, mas ainda assim o modelo é regular.