Aprendizagem de Máquina

Modelos Lineares de Regressão

Telmo de Menezes e Silva Filho
tmfilho@gmail.com/telmo@de.ufpb.br
www.de.ufpb.br

UFPB Departamento de ESTATÍSTICA

Sumário

Introdução

- Um modelo de regressão linear assume que a função $\mathbb{E}(Y|X) = f(X)$ é linear em relação às entradas $X_1, \ldots, X_p$
- Vários desses modelos foram desenvolvidos na era pre-computacional
- São modelos simples, mas podem nos ajudar a interpretar como os dados de entrada afetam a saída

Introdução

- Mesmo com a existência de modelos mais complexos, os lineares ainda são frequentemente úteis quando tempos conjuntos de dados pequenos e dados ruidosos ou esparsos
- Além disso, existem transformações dos dados que permitem usar métodos lineares mesmo que o problema não seja originalmente linear
  - Kernels e funções de base radial (RBF)

Regressão Linear com Mínimos Quadrados

- Suponha que dado um vetor de entrada $\mathbf{x}^T = (x_1, x_2, \ldots, x_p)$, nós queremos predizer um valor de saída y. Podemos usar uma função da seguinte forma:

$$f(\mathbf{x}) = \beta_0 + \sum_{j=1}^{p} x_j \beta_j$$

- Os coeficientes β_j são também chamados de **parâmetros** desconhecidos
- O modelo é linear nos parâmetros

Regressão Linear com Mínimos Quadrados

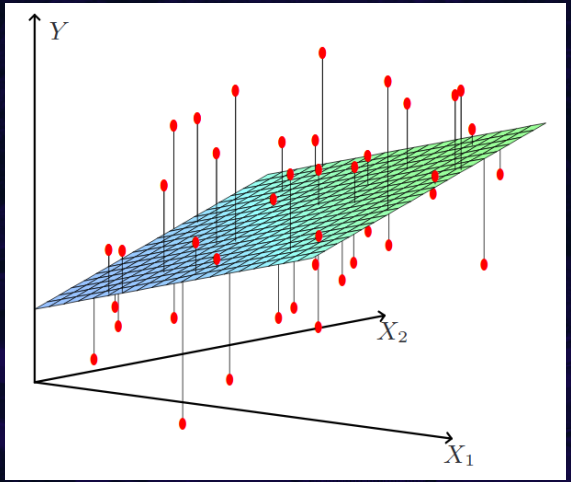
- Como mencionado na aula passada, estimamos nossos modelos usando dados
- Podemos representar nossos dados como um conjunto de tuplas $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$, em que cada $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})$ é um vetor associado a um valor alvo y_i
- Para estimar os coeficientes $\beta = (\beta_0, \beta_1, \ldots, \beta_p)^T$, o método mais popular é o dos **mínimos quadrados**

Regressão Linear com Mínimos Quadrados

- O objetivo do método dos mínimos quadrados é minimizar a soma dos quadrados dos resíduos

$$
\begin{aligned}
S(\beta) &= \sum_{i=1}^{N} (y_i - f(x_i))^2 \\
&= \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2
\end{aligned}
$$

Ajuste por Mínimos Quadrados



Minimizando os Mínimos Quadrados

- O ajuste do modelo é intuitivamente satisfatório
  - Ele simplesmente tenta encontrar o melhor ajuste linear dos dados por meio de um critério que avalia o erro médio de ajuste
- Mas como minimizamos o critério S?

Minimizando os Mínimos Quadrados

- Começamos definindo $\mathbf{y}$, o vetor de tamanho N dos valores alvo de treinamento e uma matriz $\mathbf{X}_{N\times(p+1)}$, em que cada linha representa um vetor de entrada dos nossos dados, com uma coluna a mais com o valor 1

i	x_1	x_2	y
1	-1.75	3.34	-19.56
2	1.15	2.75	10.54
3	0.98	3.51	5.53
4	0.22	1.93	0.50
5	-0.19	3.26	-5.24
6	-0.46	3.44	-7.54
7	-0.58	3.82	-9.71
8	0.67	2.90	5.26
9	-0.53	4.03	-10.69
10	-0.44	1.88	-5.10

$\rightarrow$

i	x_0	x_1	x_2	y
1	1	-1.75	3.34	-19.56
2	1	1.15	2.75	10.54
3	1	0.98	3.51	5.53
4	1	0.22	1.93	0.50
5	1	-0.19	3.26	-5.24
6	1	-0.46	3.44	-7.54
7	1	-0.58	3.82	-9.71
8	1	0.67	2.90	5.26
9	1	-0.53	4.03	-10.69
10	1	-0.44	1.88	-5.10

Minimizando os Mínimos Quadrados

- Com **X** e **y** podemos escrever o critério S como:

$$
\begin{aligned}
S(\beta) &= \sum_{i=1}^{N} \left(y_i - \sum_{\mathbf{j=0}}^{p} x_{ij}\beta_j \right)^2 \\
&= (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) \\
&= \mathbf{y}^T\mathbf{y} - \mathbf{y}^T\mathbf{X}\beta - \beta^T\mathbf{X}^T\mathbf{y} + \beta^T\mathbf{X}^T\mathbf{X}\beta
\end{aligned}
$$

Minimizando os Mínimos Quadrados

- Agora, para minimizar o critério podemos **derivar** $S(\beta)$ em relação a β e igualar a 0

$$\begin{aligned}\frac{\partial S}{\partial \beta} &= \frac{\partial(\mathbf{y}^T\mathbf{y} - \mathbf{y}^T\mathbf{X}\beta - \beta^T\mathbf{X}^T\mathbf{y} + \beta^T\mathbf{X}^T\mathbf{X}\beta)}{\partial \beta} = 0 \\ &= -2\mathbf{X}^T\mathbf{y} + 2\mathbf{X}^T\mathbf{X}\beta = 0 \\ &= \mathbf{X}^T\mathbf{X}\beta = \mathbf{X}^T\mathbf{y} \\ \hat{\beta} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}\end{aligned}$$

- Para predizer o valor alvo para um novo vetor **x** fazemos:

$$\hat{y} = \hat{f}(\mathbf{x}) = (1 : \mathbf{x})^T\hat{\beta}$$

Aplicando ao Nosso Exemplo

- Relembrando, tínhamos os seguintes dados:

i	x_1	x_2	y
1	-1.75	3.34	-19.56
2	1.15	2.75	10.54
3	0.98	3.51	5.53
4	0.22	1.93	0.50
5	-0.19	3.26	-5.24
6	-0.46	3.44	-7.54
7	-0.58	3.82	-9.71
8	0.67	2.90	5.26
9	-0.53	4.03	-10.69
10	-0.44	1.88	-5.10

- Após ajustar o modelo, temos como resultado

$$\hat{\beta} = (2.9098, 9.9125, -1.8111)^T$$

- Este é um bom momento para dizer que os coeficientes usados para gerar os dados foram

$$\hat{\beta} = (3, 10, -2)^T$$

Massa! Mas e quando dá errado?

- Quando **X** tem colunas altamente correlacionadas, ou talvez perfeitamente correlacionadas, e.g. $x_2 = 3 * x_1$, $\mathbf{X}^T\mathbf{X}$ é singular, ou seja **não invertível**
- Isso é ruim? Sim, é péssimo
  - Nesses casos, $\hat{\beta}$ não é unicamente definido
- Problemas também podem aparecer quando $p > N$
- Assim, é muito comum que implementações de regressão linear calculem a pseudo-inversa de $\mathbf{X}^T\mathbf{X}$

Propriedades Amostrais de $\hat{\beta}$

- Agora vamos ver uma das características mais legais do modelo de regressão linear com mínimos quadrados: a possibilidade de diagnosticar a **importância das variáveis independentes** X para determinar o valor alvo Y
- Vamos começar assumindo que as observações y_i tem variância σ^2 e que os $\mathbf{x}_i$ não são aleatórios
- Podemos estimar σ^2 fazendo

$$\hat{\sigma}^2 = \frac{1}{N - p - 1} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

Propriedades Amostrais de $\hat{\beta}$

- Vamos supor também que o valor esperado condicional de Y é linear em relação a $X_1, \ldots, X_p$ e que quaisquer desvios de Y ao redor de seu valor esperado são representados pelo acréscimo de um erro Gaussiano

$$Y = \mathbb{E}(Y|X_1, \ldots, X_p) + \epsilon$$
$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$

- Com isso:

$$\hat{\beta} \sim \mathcal{N}(\beta, (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2)$$

Propriedades Amostrais de $\hat{\beta}$

- Com isso:

$$\hat{\beta} \sim \mathcal{N}(\beta, (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2)$$

- Com a hipótese nula de que $\beta_j = 0$, e como não conhecemos σ^2 (precisamos estimá-lo a partir dos dados), temos que a estatística t_j é

$$t_j = \frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{a_{jj}}} \sim t(N - p - 1)$$

- Onde a_{jj} é o j-ésimo elemento da diagonal de $\mathbf{X}^T\mathbf{X}$
- Naturalmente, se o **número de instâncias for muito grande** (e muito maior que p), podemos usar a distribuição **Normal**, ao invés da t de Student

Voltando ao Nosso Exemplo

	y	$\hat{y}$
1	-19.56	-20.486004
2	10.54	9.328740
3	5.53	6.267213
4	0.50	1.595192
5	-5.24	-4.877636
6	-7.54	-7.879998
7	-9.71	-9.757699
8	5.26	4.299086
9	-10.69	-9.642397
10	-5.10	-4.856498

- A variância estimada é

$$\hat{\sigma}^2 = \frac{1}{N-p-1}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

$$\hat{\sigma}^2 = 0.9138$$

Voltando ao Nosso Exemplo

- Temos que $\mathbf{X}^T\mathbf{X}$ é

$$\mathbf{X}^T\mathbf{X} = \begin{bmatrix} 10.0 & -0.93 & 30.86 \\ -0.93 & 6.9013 & -4.2556 \\ 30.86 & -4.2556 & 100.002 \end{bmatrix}$$

j	$\hat{\beta}_j$	t_j	p-value
0	2.9098	0.9626	0.3678
1	9.9125	3.9471	0.0056
2	-1.8111	-0.1894	0.8551

- Para rejeitar $\beta_j = 0$ com $\alpha = 0{,}05$ e $N - p - 1 = 7$ graus de liberdade, precisamos de $t_j \geq 2{,}36$ ou $t_j \leq 2{,}36$

$$t_j = \frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{a_{jj}}}$$

E Tem Mais...

- Também é possível avaliar se grupos de variáveis independentes são úteis para explicar a variável dependente
- Isso pode ser muito útil para testar a importância de variáveis categóricas transformadas em variáveis dummy/one-hot encoding
- Ver Capítulo 3.2 do *Elements of Statistical Learning*, a partir da página 48

Regressão Multivariada

- Suponha que existam K valores alvo $Y_1, \ldots, Y_K$ que você deseja predizer dadas as suas entradas $X_1, \ldots, X_p$
- Assumimos um modelo linear para cada saída

$$\begin{aligned} Y_k &= \beta_{0k} + \sum_{j=1}^{p} X_j \beta_{jk} + \epsilon_k \\ &= f_k(\mathbf{x}) + \epsilon_k \end{aligned}$$

- Com N instâncias, podemos reescrever o modelo em notação de matriz

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E}$$

Regressão Multivariada

- Com N instâncias, podemos reescrever o modelo em notação de matriz

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E}$$

- Aqui, $\mathbf{Y}$ é uma matriz de respostas $N \times K$, $\mathbf{B}$ é uma matriz $(p+1) \times K$ de parâmetros, $\mathbf{E}$ é uma matriz $N \times K$ de erros e $\mathbf{X}$ é a mesma matriz de entrada $N \times (p+1)$ do caso univariado

Regressão Multivariada

- Com N instâncias, podemos reescrever o modelo em notação de matriz

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E}$$

- Podemos então generalizar a soma dos mínimos quadrados

$$\begin{aligned} S(\mathbf{B}) &= \sum_{k=1}^{K} \sum_{i=1}^{N} (y_{ik} - f_k(x_i))^2 \\ &= tr\left[(\mathbf{Y} - \mathbf{XB})^T (\mathbf{Y} - \mathbf{XB}) \right] \end{aligned}$$

Regressão Multivariada

- A solução dos mínimos quadrados vai ter exatamente a mesma forma do caso univariado

$$\hat{\mathbf{B}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

- Ou seja, os coeficientes equivalem aos que seriam obtidos por K regressões separadas
- Esse resultado supõe que os erros das K respostas não são correlacionados* e suas covariâncias **não variam ao longo das observações**

Controlando os Pesos

Regressão Ridge

- Regressão ridge é um tipo de regressão que busca controlar os coeficientes impondo uma penalidade aos seus valores
- Essa abordagem é também chamada de **regularização** ou de **weight decay** nas redes neurais

Regressão Ridge

- Quando existem múltiplas variáveis correlacionadas em um modelo de regressão linear, os coeficientes podem exibir alta variância
  - Um coeficiente **altamente positivo** em uma variável pode ser cancelado por um **altamente negativo** em outra variável correlacionada
- O objetivo da regressão ridge é aliviar esse problema

Regressão Ridge

- A soma dos mínimos quadrados penalizada é dada por

$$\hat{\beta}^{\text{ridge}} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

- $\lambda \geq 0$ é um parâmetro de complexidade que define o tamanho da penalização
- Essa penalização também é chamada de L_2
- A solução da regressão ridge se beneficia de uma prévia normalização dos dados de entrada

Regressão Ridge

- A soma dos mínimos quadrados penalizada é dada por

$$\hat{\beta}^{\text{ridge}} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

- Note que o intercepto β_0 não é penalizado
- A solução é feita em duas partes:

Regressão Ridge

- A solução é feita em duas partes
- Primeiro estimamos

$$\beta_0 = \bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$$

- Para a segunda parte, centralizamos as entradas, i.e. cada x_{ij} é substituído por $x_{ij} - \bar{x}_j$ e montamos a matriz $\mathbf{X}$, sem a coluna extra igual a 1, para minimizar o critério

$$S(\beta, \lambda) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta^T\beta$$

Regressão Ridge

- A solução da regressão ridge é então dada por

$$\hat{\beta}^{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$$

- Onde $\mathbf{I}$ é uma matriz identidade $p \times p$
- Note que a solução adiciona uma **constante positiva** λ à diagonal de $\mathbf{X}^T\mathbf{X}$ antes da inversão
- Essa foi a motivação inicial por trás do método, para evitar o problema da singularidade

Regressão Lasso

- Na regressão lasso, a soma dos mínimos quadrados penalizada é dada por

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \left\{ \frac{1}{2} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

- Essa penalização é também chamada de regularização L_1
- Note a similaridade com regressão ridge
- Um valor de λ suficientemente pequeno fará com que alguns coeficientes sejam exatamente 0

Regressão Lasso

- A nova restrição em cima dos valores de β_j faz com que **não exista expressão de forma fechada** para solucionar o problema
  - Temos um problema de programação quadrática
- Assim como na ridge, a solução é feita em duas partes: estimamos $\beta_0 = \bar{y}$, depois centralizamos os x_{ij} normalizados antes de minimizar o critério

Regressão Elastic Net

- A regressão lasso tem algumas limitações
  - Em problemas com p grande e N pequeno, a regressão lasso selecionará no máximo N variáveis
  - Além disso, em um grupo de variáveis muito correlacionadas, a lasso tende a selecionar apenas uma variável do grupo
- Para solucionar essas limitações a regressão elastic net adiciona a penalização quadrática da ridge à penalização da lasso

Regressão Elastic Net

- O problema então torna-se:

$$\hat{\beta} = \arg\min_{\beta} \left\{ \frac{1}{2} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda_2 \sum_{j=1}^{p} \beta_j^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| \right\}$$

- O termo quadrático faz com que exista uma solução única para o problema
- Com $\lambda_1 = 0$, temos ridge, com $\lambda_2 = 0$, temos lasso e com $\lambda_1 = \lambda_2 = 0$, temos regressão linear sem regularização

Softwares Disponíveis

Softwares Disponíveis

- Python
  - Statsmodels
    - Faz as avaliações de significância dos coeficientes
  - Sklearn
- R
  - glm (modelos lineares generalizados)
  - glmnet: elastic net
- Javascript
  - Vai no Google
- Julia
  - Roberts

FIM

Aprendizagem de Máquina

Modelos Lineares de Regressão

Telmo de Menezes e Silva Filho
tmfilho@gmail.com/telmo@de.ufpb.br
www.de.ufpb.br

UFPB Departamento de ESTATÍSTICA