An Introduction to Statistical Learning

Resumo - Capítulos 1 à 8

Ana Laíse Nascimento & Miguel Feliciano Mota Alves

Agosto, 2024

Sumário

1 INTRODUÇÃO

1.1 UMA VISÃO GERAL DO APRENDIZADO ESTATÍSTICO

O Aprendizado Estatístico refere-se a uma gama de ferramentas utilizadas para entender dados. Essas ferramentas podem ser classificadas como supervisionadas ou não supervisionadas. Em termos gerais, o aprendizado supervisionado envolve a construção de um modelo estatístico para predição ou estimação de uma saída baseada em uma ou mais entradas.

Em contrapartida - no aprendizado não supervisionado - há entradas, porém não há saídas supervisionadas. Ainda assim, podemos aprender sobre os relacionamentos e estruturas dos dados.

1.2 UMA BREVE HISTÓRIA DO APRENDIZADO ESTATÍSTICO

Apesar do termo aprendizado estatístico ser relativamente novo, muitos conceitos foram desenvolvidos há muito tempo. No começo do século XIX, por exemplo, o método do mínimos quadrados[1] foi desenvolvido, implementando a forma mais inicial do que hoje é conhecido como regressão linear[2].

A regressão linear é utilizada para predição de valores quantitativos, como o salário de um indivíduo. Para prever valores qualitativos, a exemplo do aumento ou diminuição do mercado de ações, a análise de discriminante[3] foi proposta em 1936. Na década de 1940, um método alternativo foi proposto: regressão logística[4]. No começo da década de 1970, o modelo linear generalizado[5] foi desenvolvido para descrever tanto regressão linear quanto logística.

Por causa do gargalo computacional, a grande maioria dos métodos inventados nesta década eram lineares. Contudo, na década de 1980, esse gargalo foi diminuído a tal ponto que modelos não lineares foram permitidos computacionalmente. No meio dos anos 80, árvores de classificação e regressão[6] foram desenvolvidas, seguidos pelo modelo aditivo generalizado[7]. Redes neurais[8] ganharam popularidade na década de 80 e máquinas de

[1] Também conhecido como Mínimos Quadrados Ordinários (MQO) ou, em inglês, *least squares* ou, ainda, *Ordinary Least Squares* (OLS).
[2] Em inglês, *linear regression.*
[3] Em inglês, *linear discriminant analysis* (LDA).
[4] Em inglês, *logistic regression.*
[5] Conhecido como *generalized linear model* (GLM) em inglês.
[6] *Classification tree* e *regression tree* em inglês.
[7] Também conhecido como *generalized additive models* (GAM).
[8] *Neural networks* em inglês.

vetores de suporte[9] em 90.

Desde então, o aprendizado estatístico vem emergindo como um novo ramo da estatística. Nos dias de hoje, o progresso desse ramo vem sendo marcado pelo aumento da disponibilidade de ferramentas poderosas e de usabilidade amigável, como a linguagem de programação Python.

1.3 NOTAÇÃO

Para representar o número de dados distintos ou observações na nossa amostra, usaremos n. Usaremos p para denotar o número de variáveis disponíveis para fazer predições ou, simplesmente, variáveis de entrada. Em alguns casos, p pode ser muito grande: na ordem de milhares ou até mesmo milhões.

Em termos gerais, x_{ij} vai representar o valor da observação i da variável j, onde $i = 1, 2, ..., n$ e $j = 1, 2, ..., p$. Durante esse resumo, i será usado para indexar as amostras ou observações (de 1 até n) e j será utilizado para indexar as variáveis (de 1 até p). X irá denotar a matriz $n \times p$ cujo (i, j) elemento é x_{ij}. Matricialmente, isso é:

$$X = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}$$

Para representar as linhas de X, é usado o vetor de tamanho p, que contém os p valores da observações i:

$$x_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{ip} \end{pmatrix} \tag{1}$$

Os vetores são por por padrão representados por colunas.

Para representar as colunas de X, é usado o vetor de tamanho n, que contém os n valores da variável p:

$$x_j = \begin{pmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{nj} \end{pmatrix}$$

[9] Em inglês, *Support Vector Machine* (SVM).

A notação ^{T} denota a transposição da matriz ou vetor:

$$X^T = \begin{pmatrix} x_{11} & x_{21} & \dots & x_{n1} \\ x_{12} & x_{22} & \dots & x_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1p} & x_{2p} & \dots & x_{np} \end{pmatrix}$$

Usamos y_i para representar a observação i da variável a qual queremos fazer predições. Em forma de vetor:

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

Então, os dados consistem de $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$, onde cada x_i é um vetor de tamanho p (Se $p = 1$, então x_i é um escalar).

2 APRENDIZADO ESTATÍSTICO

2.1 O QUE É APRENDIZADO ESTATÍSTICO

Em termos gerais, suponha que observamos uma resposta[10] quantitativa Y e p diferentes preditores[11], $X_1, X_2, ..., X_p$. Nós assumimos que há um certo tipo de relacionamento entre Y e $X = (X_1, X_2, ..., X_p)$, o qual pode ser escrito na forma geral

$$Y = f(X) + \epsilon \tag{2}$$

Aqui f é uma função fixada, mas desconhecida, e ϵ é um termo de erro randômico, que é independente de Xe tem média zero. A função f representa a informação sistemática que X fornece sobre Y.

Em essência, o aprendizado estatístico refere-se a um conjunto de métodos para estimar f.

[10] Outros nomes que a saída do modelo recebe: variável resposta, variável dependente, variável predita. Em inglês, a saída é chamada output, recebendo os seguintes nomes: *variable response, dependent variable, output variable.*

[11] Além de variável preditora, a entrada recebe estes nomes: variável independente, variável de entrada, entrada. Em inglês, *predictor, independent variable, feature, input variable.*

2.1.1 Por que estimar f

Há duas razões principais para o desejo de estimar f: predição e inferência.

2.1.1.1 Predição

Em muitas situações, um conjunto de entrada X está prontamente disponível, porém a saída Y não pode ser obtida facilmente. Nessa situação, uma vez que a média do termo de erro tende a zero, podemos predizer Y usando

$$\hat{Y} = \hat{f}(X), \tag{3}$$

onde $\hat{f}$ representa a nossa estimação para f, e $\hat{Y}$ representa a predição resultante de Y. Nesse contexto, f é usualmente uma caixa preta no sentido de sua forma exata não ter uma importância tão grande para o estatístico, desde que f apresente predições acuradas de Y.

A acurácia de Y como uma predição depende de duas parcelas, as quais chamaremos de erro redutível e irredutível. Como, em geral, f não vai ser uma predição perfeita, algum erro naturalmente vai ser gerado.

O erro é redutível porque pode-se melhorar potencialmente a acurácia de f ao utilizar a melhor técnica de aprendizado estatístico. Em compensação, o erro é irredutível porque - por definição - Y também está em função de ϵ, que não pode ser predito por X. O erro irredutível é maior que zero, uma vez que a parcela pode conter variáveis imensuráveis que são úteis na predição de Y.

Considere uma estimativa fixa f e um conjunto de preditores fixos X, o que produz uma predição $\hat{Y} = \hat{f}(X)$ e uma variabilidade vinda somente de ϵ. É fácil mostrar que

$$\begin{aligned} E(Y - \hat{Y})^2 &= E|f(X) + \epsilon - \hat{f}(X)|^2 \\ &= \underbrace{|f(X) - \hat{f}(X)|^2}_{\text{Redutível}} + \underbrace{Var(\epsilon)}_{\text{Irredutível}}, \end{aligned} \tag{4}$$

onde $E(Y - \widehat{Y})^2$ representa a média ou valor esperado da diferença ao quadrado entre o valor predito e o verdadeiro valor de Y, e $Var(\epsilon)$ representa a variância associada ao valor de ϵ.

2.1.1.2 Inferência

Muitas vezes, estamos interessados em entender a associação entre Y e $X_1, \ldots, X_p$. Nesse caso, temos o desejo de estimar f, mas nosso objetivo não é necessariamente fazer predições para Y. Agora $\hat{f}$ não pode ser tratada como uma caixa preta, já que queremos saber sua forma exata. Nesse contexto, as seguintes respostas para as suas respectivas perguntas podem ser interessantes:

- Que preditores são associados à variável resposta?
  Identificar os poucos preditores importantes dentro um grande conjunto de X pode ser extremamente útil.

- Qual o relacionamento entre a resposta e cada preditor?
  Algumas varíaveis de entrada podem ter um relacionamento positivo com a saída. Já outros podem ter o oposto. Pode-se, ainda, ter uma dependência entre o valor de outras variáveis e o relacionamento da saída e cada entrada.

- O relacionamento entre Y e cada preditor pode ser sumarizado adequadamente utilizando uma equação linear ou o relacionamento é mais complicado? Historicamente, a maioria dos métodos para estimar f assumiram uma abordagem linear. Entretanto, isso nem sempre pode ser verdade: um modelo linear pode não fornecer uma representação acurada do relacionamento.

Dependendo do objetivo final, seja predição, inferência ou uma combinação dos dois, diferentes métodos podem ser apropriados.

Por exemplo, regressões lineares permitem uma relativamente simples e interpretável inferência, mas podem não resultar em predições acuradas. Redes neurais, em contrapartida, oferecem predições mais acertadas em detrimento de interpretabilidade.

2.1.2 Como estimar f?

Nosso objetivo é aplicar um método de aprendizagem estatística nos dados de treinamento[12] a fim de estimar uma função desconhecida f: $Y \approx \hat{f}(X)$, que funciona para qualquer (X, Y). A maioria desses métodos de aprendizagem podem ser classificados como paramétricos ou não paramétricos.

2.1.2.1 Métodos paramétricos

Métodos paramétricos envolvem uma abordagem com duas etapas:

1. Primeiro, fazemos uma suposição sobre a forma funcional de f. Podemos, por

[12] São observações usadas para treinar ou ensinar como estimar f.

exemplo, supor que f é linear em X:

$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p + \tag{5}$$

Nesse caso, só precisamos estimar os valores dos $p + 1$ coeficientes $\beta_0, \beta_1, ..., \beta_p$.

2. Depois que o modelo foi selecionado, precisamos de um procedimento que utiliza os dados de treinamento para ajustar ou treinar o modelo. No caso do modelo linear , precisamos estimar os valores dos $p + 1$ coeficientes $\beta_0, \beta_1, ..., \beta_p$.

Assumir uma forma paramétrica simplifica o problema, uma vez que é muito mais fácil estimar um conjunto de parâmetros do que descobrir uma função arbitrária desconhecida f. A desvantagem é que quase nunca o modelo escolhido vai corresponder exatamente o f desconhecido.

Para tentar corresponder melhor, pode-se escolher um modelo mais flexível, contudo isso pode levar a um fenômeno conhecido como sobreajuste[13] ou, em inglês, *overfitting*.

2.1.2.2 Métodos não paramétricos

Métodos não paramétricos não fazem suposições explícita da forma funcional de f. Em vez disso, é procurado uma estimativa de f que aproxima-se dos dados o máximo possível sem que haja uma distorção tão abrupta.

Uma grande vantagem desse método é o potencial de ajustar com acurácia uma vasta faixa de formas para f. Em compensação, como não é feito suposição, um grande número de observações é necessário para uma estimação acurada de f.

2.1.3 O paradigma entre predição acurada e interpretabilidade do modelo

Alguém pode se perguntar: por que escolher um método mais restritivo em vez de uma abordagem mais flexível? Há muitas razões, se estamos interessados em interpretabilidade, modelos mais restritivos normalmente são melhores.

Devemos, então, utilizar modelos mais flexíveis para predição? A resposta é: nem sempre! Esse fenômeno, que pode parecer contraintuitivo a primeira vista, tem a ver com o potencial de *overfitting* das abordagens mais flexíveis.

[13] O sobreajuste significa que o modelo segue os erros ou ruídos de forma muito próxima.

2.1.4 Supervisionado X Não supervisionado

A maioria dos problemas de aprendizagem estatística estão nestas duas categorias: supervisionado ou não supervisionado. No aprendizado supervisionado - para cada observação do preditor x_i, $i = 1, ..., n$, observamos uma variável resposta associada. Grande parte dos métodos estatísticos clássicos - a exemplo de regressão linear e logística, modelo aditivo generalizado (GAM), *boosting* e máquinas de vetores de suporte (SVM) - operam no domínio de aprendizado supervisionado.

Em contraste, aprendizado não supervisionado descreve de certa forma uma situação mais desafiadora na qual, para cada observação $i = 1, 2, ..., n$, observamos um vetor de medida x_i, mas nenhuma resposta associada y_i. Não é possível, por exemplo, ajustar um modelo de regressão linear, já que não há variável resposta para predizer.

Sendo assim, que tipo de análise estatística podemos usar? Podemos buscar entender os relacionamentos entre as variáveis ou, até mesmo, entre as observações. Um exemplo de ferramenta estatística que pode-se usar é a análise de *clustering*, a qual - em poucas palavras - busca verificar se observações pertencem a grupos relativamente distintos.

Apesar da maioria dos problemas tenderem às categorias de supervisionado ou não supervisionado, nem sempre a classificação é tão clara. Por exemplo, suponha que temos um conjunto de dados com n observações. Para m dessas observações, onde $m < n$, temos tanto as medidas de preditores quanto as de resposta. Para as $n - m$ observações restantes, temos somente as medidas de variáveis de entrada. Esse tipo de cenário chama-se de aprendizado semi-supervisionado e, como abordagem, podemos usar um método que incorpore tanto as m observações, que têm uma resposta associada, quanto as $n - m$ observações, as quais nâo têm variável de saída. Embora seja um problema bem interessante, não será abordado neste resumo.

2.1.5 Regressão X Classificação

As variáveis podem ser classificadas como quantitativas ou qualitativas. As quantitativas possuem valores numéricos, como a idade de uma pessoa, a altura e valor de uma casa. Em contrapartida, as variáveis qualitativas assumem valores de uma das K diferentes classes ou categorias, a exemplo do estado civil (solteiro, casado, divorciado ou viúvo) e a marca do produto comprado (marca A, B ou C).

Geralmente, os problemas com resposta qualitativa são abordados como problemas de classificação, enquanto os com resposta quantitativa são abordados como problemas de regressão. Entretanto, essa distinção nem sempre é tão clara.

Tende-se a selecionar métodos com base no tipo de resposta (qualitativa ou quan-

titativa), ou seja, podemos usar regressão linear para quantitativo e regressão logística quando é qualitativo. Contudo, se os preditores são qualitativos ou quantitativos, não tem muita importância, uma vez que grande parte dos métodos estatísticos podem ser utilizados com qualquer tipo de variável desde que ela seja codificada propriamente.

2.2 AVALIANDO A ACURÁCIA DO MODELO

Um dos objetivos mais importantes desse resumo é introduzir ao leitor uma vasta gama de métodos de aprendizado estatísticos. Isso é necessário porque não há um método que seja melhor que todos os outros para todos os *data sets*. Portanto, é uma tarefa essencial e desafiadora decidir qual ferramenta produz os melhores resultados. Nessa subseção, discute-se os conceitos mais importantes que surgem nessa seleção

2.2.1 Medindo a qualidade do ajuste

A fim de avaliar a perfomance de método estatístico em conjunto de dados, precisamos de alguma forma de mensurar o quão bem as predições correspondem a saída dos dados.

No cenário de regressão, a medida mais comum utilizada é o erro quadrático médio (MSE)[14], dado por

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2, \tag{6}$$

onde $\hat{f}(x_i)$ é a predição que $\hat{f}$ dá para a observação i. O MSE será pequeno se as respostas preditas estão perto das respostas verdadeiras será grande se, para algumas observações, o valor predito e a verdadeira resposta diferirem substancialmente.

O MSE em (6) é computado usando os dados de treino, então deve ser referenciado como MSE de treinamento. Entretanto, estamos interessados na acurácia das predições que obtemos quando aplicamos o modelo para dados de teste previamente não vistos.

Então, queremos um método que dê o menor valor de MSE de teste. Em outras palavras, se tivéssemos um grande número de observações de teste, computaríamos

$$Ave(y_0 - \hat{f}(x_0))^2, \tag{7}$$

a média quadrática do erro de predição[15] para as observações de teste (x_0, y_0).

[14] *Em inglês, mean squared error*
[15] *Average squared prediction error em inglês.*

Como escolher o método que minimiza o MSE de teste? Em alguns cenários, temos o *data set* de teste disponível e, então, simplesmente utilizar a equação (7) e selecionar o método no qual o MSE de teste é o menor.

Em caso de não termos o conjunto de teste disponível, uma solução lógica seria escolher a ferramenta com menor MSE de treinamento. Isso, porém, não garante que o método com menor MSE de treino terá o menor MSE de teste.

Uma propriedade fundamental do aprendizado de máquina, funciona independente do conjunto de dados, é que, à medida que flexibilidade do modelo aumenta, o MSE de treinamento diminuirá - mas MSE de teste pode não diminuir, causando o efeito chamado *overfitting*.

O sobreajuste acontece quando a ferramenta estatística está trabalhando muito para descobrir padrões no *training data* e acaba pegando padrões causados por um evento aleatório. Note que não obstante o *overfitting* ocorreu, espera-se que o MSE de treino seja menor que o MSE de teste, tendo em vista que a maioria dos modelos tendem a minimizar, direta ou indiretamente, o *training* (treinamento) MSE. O sobreajuste refere-se especificamente ao caso em que o modelo mais flexível gera menores valores de *test* (teste) MSE.

Um método muito importante para estimar o MSE de teste é a validação cruzada[16] (Capítulo 5).

2.2.2 O paradigma do Viés e Variância

Apesar da prova matemática ser além do escopo do livro e do resumo, é possível mostrar que o valor esperado do MSE de teste, para um dado valor x_0, pode ser sempre decomposto na soma de três parcelas fundamentais: a variância[17] de $\hat{f}(x_0)$, o viés[18] ao quadrado de $\hat{f}(x_0)$ e a variância do termo de erro ϵ. Isso é:

$$E(y_0 - \hat{f}(x_0))^2 = Var(\hat{f}(x_0)) + [Bias(\hat{f}(x_0))]^2 + Var(\epsilon). \tag{8}$$

Aqui a notação $E(y_0 - \hat{f}(x_0))^2$ define o MSE de teste esperado para x_0 e refere-se ao MSE de teste médio que obteríamos se repetidamente estimássemos f usando um grande número de *training sets* (conjuntos de treinamento) e testando para cada x_0.

A equação (8) nos diz que a fim de atingir o erro esperado mínimo, devemos selecionar um método de aprendizado de máquina que gera simultaneamente baixa variância e baixo

[16] Em inglês, *cross-validation*.
[17] Popularmente conhecida como *variance*.
[18] Em inglês, *bias*.

viés.

Variance refere-se a quantidade de $\hat{f}$ que iria mudar caso a estimássemos usando um diferente conjunto de treinamento. Desde que um conjunto de dados é usado para ajustar um modelo, um *dataset* diferente vai resultar em um f diferente, mas, idealmente, essa mudança deve ser pequena.

Se um método tem uma alta variância, porém, pequenas mudanças no conjunto de treino resultarão em grande mudanças em $\hat{f}$. Em geral, ferramentas mais flexíveis tem uma maior variância.

Por outro lado, o viés refere-se ao erro que é introduzido ao aproximar um problema real, o qual pode ser muito complicado, por um modelo simples. Por exemplo, a regressão linear assume que há um relacionamento linear entre as variáveis, contudo é improvável que isso seja realmente verdade para problemas da vida real. Então, isso vai indubitavelmente gerar *bias* na estimativa de f. Em média, métodos mais flexíveis resultam em menos viés.

Como regra geral, à medida que usamos ferramentas mais flexíveis, a variância vai aumentar e o viés vai diminuir. Logo, a taxa de mudança dessas duas parcelas determina se o MSE de teste aumenta ou diminui. Conforme a flexibilidade do método aumenta, o viés tende a inicialmente diminuir mais rápido do que a variância aumenta, diminuindo o MSE de teste. Em compensação, em algum ponto, o aumento da flexibilidade não afeta significativamente o viés, mas aumenta rapidamente a variância, aumentando o MSE.

Esse fenômeno supracitado e exemplificado na Figura (1) é o que chamamos paradigma do viés e variância[19]. Na vida real, o tipo de decomposição feito em (8) geralmente não é possível, entretanto deve-se sempre ter em mente esse *trade-off*.

2.2.3 O cenário de Classificação

Até o momento, a discussão de acurácia de modelo focou bastante no plano de regressão. Muitos dos conceitos, contudo, podem ser transferidos para o cenário da classificação com apenas algumas modificações por causa do fato de y_i não ser mais quantitativo.

A abordagem mais comum para quantificar a acurácia da estimação é pela taxa de erro do treinamento, a proporção de erros que são feitos se aplicarmos a estimativa $\hat{f}$ no conjunto de observações.

$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i). \tag{9}$$

[19]Mais popularmente conhecido como *bias-variance trade-off*.

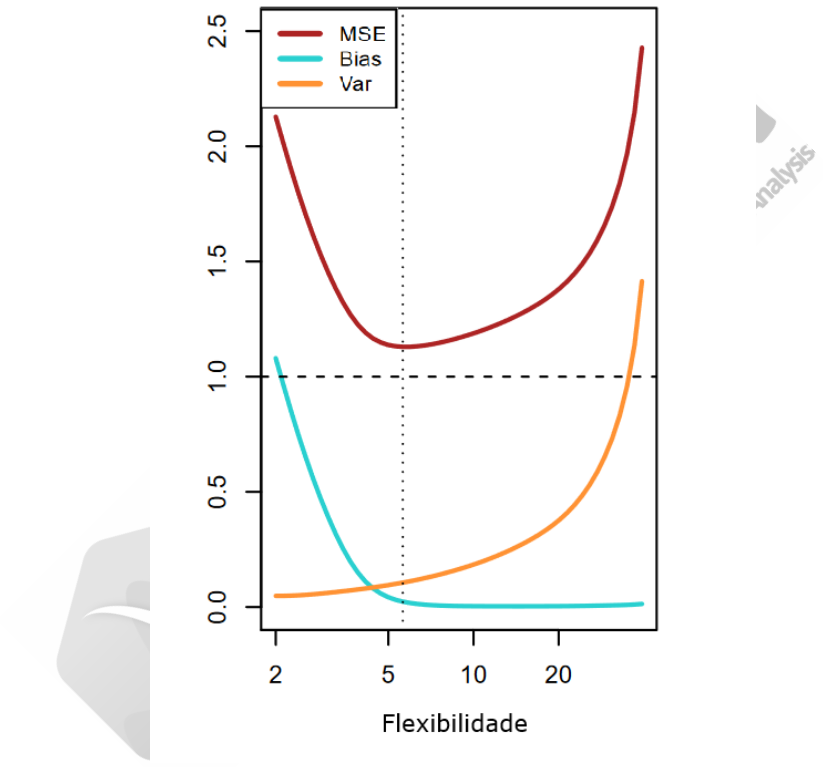

Figura 1: Viés (curva azul), variância (curva laranja) e MSE de teste (curva vermelha) exemplificando o fenômeno. A linha pontilhada vertical representa o nível de flexibilidade do menor MSE de teste.

Aqui $\hat{y}_i$ é classe predita para a observação i usando $\hat{f}$, e I é uma variável indicadora, que é igual a 1 se $y_i \neq \hat{y}_i$ e 0 se $y_i = \hat{y}_i$.

A taxa de erro de teste é dada por

$$Ave(I(y_0 \neq \hat{y}_0)), \tag{10}$$

onde $\hat{y}_0$ é classe predita resultante da aplicação do modelo em x_0. Um bom modelo classificador é que possui o menor teste de erro (10).

2.2.3.1 O classificador de Bayes

É possível mostrar que a taxa de erro de teste é minimizada, em média, por um simples classificador que atribui a cada observação sua classe mais provável dado o valor do preditor:

$$Pr(Y = j|X = x_0) \tag{11}$$

Note que (11) é uma probabildade condicional: é a probabilidade de $Y = j$ dado um vetor preditor observado x_0. Isso é o chamado classificador de Bayes ou, em inglês, *Bayes classifier*.

O classificador de Bayes produz a menor taxa de erro de teste possível, chamada taxa de erro de Bayes[20]. Em geral, a taxa de erro geral de Bayes é dada por

$$1 - E(\max_j Pr(Y = j|X), \tag{12}$$

onde a expectativa calcula a média da probabilidade sobre todos os valores possíveis de X. A taxa de erro de Bayes é similar ao erro irredutível, discutido anteriormente.

2.2.3.2 K-Vizinhos mais Próximos

Teoricamente, nós sempre temos a meta de utilizar o classificador de Bayes para predição. Para dados reais, entretanto, nós não temos a distribuição condicional de Y dado X. Por conseguinte, não é possível usar o *Bayes classifier*.

Uma forma de classificar uma observação com a mais alta probabilidade estimada é o K-vizinhos mais próximos (KNN)[21]. Dado um inteiro K e uma observação de teste x_0, o KNN primeiro identifica os K pontos no conjunto de treinamento que são mais próximos

[20] Em inglês, *Bayes error rate*.
[21] Mais comumente chamado de *K-nearest neighbors* (KNN).

de x_0, representado por $\mathcal{N}_0$. Depois, é estimada a probabilidade condicional da classe j para a fração de pontos em $\mathcal{N}_0$ cuja resposta equivale j:

$$Pr(Y = j|X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} I(y_i = j). \tag{13}$$

Finalmente, KNN classifica a observação de teste x_0 na classe com a maior probabilidade de (13). Apesar do fato de ser uma abordagem bem mais simples, o KNN costuma produzir classificadores surpreendemente próximos do classificador Bayes ótimo.

A escolha do K tem um efeito drástico no *KNN classifier* obtido. Quanto menor o K, menor o viés, maior a variância e maior a flexibilidade do modelo. Por outro lado, quanto maior o K, maior o viés, menor a variância e menor a flexibilidade do modelo.

Análogo ao cenário de regressão, não há uma conexão tão forte entre a taxa de erro de treinamento e de teste. Portanto, escolher o nível de flexibilidade do modelo é crítico e o conceito de *bias-variance Trade-off* continua com uma importância vital.

3 REGRESSÃO LINEAR

3.1 REGRESSÃO LINEAR SIMPLES

A regressão linear simples modela a relação entre uma variável dependente Y e uma variável independente X usando uma linha reta. Este modelo é construído baseando-se em dois parâmetros principais: o intercepto α e o coeficiente de inclinação β que são estimados a partir dos dados. Os métodos para estimar esses coeficientes e avaliar a precisão das estimativas são discutidos fornecendo uma base para a previsão e interpretação das relações entre variáveis. A relação pode ser aproximadamente representada pela equação linear abaixo:

$$Y \approx \beta_0 + \beta_1 X$$

Por exemplo, X pode representar quantidade de anúncios de **TV** e Y pode representar as **vendas**. Podemos, assim, fazer a regressão de vendas a partir de TV ao ajustar o modelo

$$\textbf{vendas} \approx \beta_0 + \beta_1 \textbf{TV}$$

3.1.1 Fundamentos Matemáticos para o Cálculo de β_0 e β_1

A regressão linear simples usa a equação da linha reta:

$$Y = \beta_0 + \beta_1 X + \epsilon$$

onde:

- Y é a variável dependente (resposta)
- X é a variável independente (preditor)
- β_0 é o intercepto da linha no eixo Y
- β_1 é a inclinação da linha indicando a mudança em Y para uma mudança de uma unidade em X
- ϵ é o termo de erro representando a diferença entre os valores observados e os valores ajustados pelo modelo.

O objetivo é encontrar os valores de β_0 e β_1 que minimizem a soma dos quadrados dos resíduos (RSS), onde o resíduo de cada ponto é a diferença entre o valor observado de Y e o valor previsto por nosso modelo.

3.1.2 Minimizando o RSS

Para minimizar o RSS em relação a β_0 e β_1, utilizamos o cálculo diferencial, tomando a derivada parcial do RSS com respeito a cada coeficiente e igualando a zero. Isso nos dá um sistema de equações, frequentemente chamado de equações normais:

$$\frac{\partial RSS}{\partial \beta_0} = -2 \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i) = 0$$

$$\frac{\partial RSS}{\partial \beta_1} = -2 \sum_{i=1}^{n} x_i (y_i - \beta_0 - \beta_1 x_i) = 0$$

Resolvendo este sistema para β_0 e β_1, obtemos as fórmulas para calcular os estimadores de mínimos quadrados para os coeficientes, indicando a relação média entre X e Y enquanto o intercepto β_0 representa o valor esperado de Y quando $X = 0$.

Essa abordagem matemática para a estimativa dos coeficientes de regressão não só permite a interpretação das relações entre variáveis, mas também fundamenta muitas das propriedades estatísticas dos estimadores, incluindo sua variância e distribuição, facilitando testes de hipóteses e a construção de intervalos de confiança.

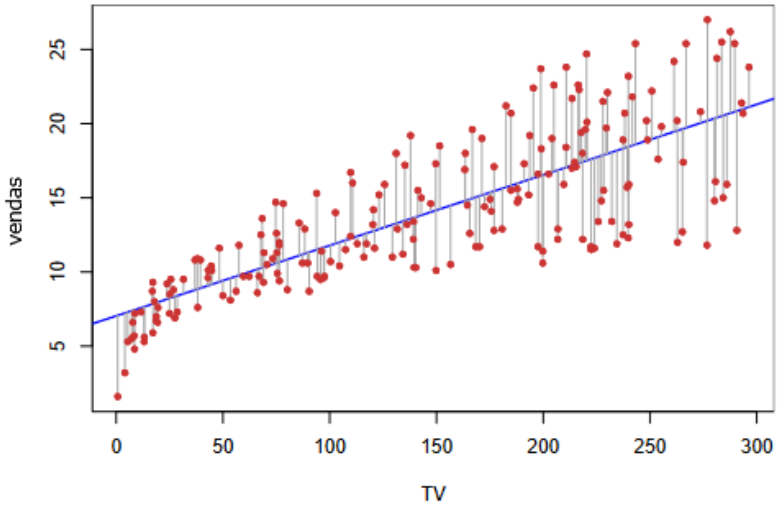

Figura 2: Para os dados de **anúncio de TV**, o ajuste de mínimos quadrados para uma regressão de **vendas** a partir de **TV** é mostrado. O ajuste é encontrado ao minimizar o RSS. Cada linha cinza representa um resíduo.

3.1.3 Precisão das Estimativas dos Coeficientes

Tópico crucial ao avaliar a qualidade e a confiabilidade de um modelo de regressão linear. Este conceito refere-se à avaliação de quão próximas as estimativas dos coeficientes de regressão, β_0 (intercepto) e β_1 (inclinação), estão dos verdadeiros valores dos parâmetros na população. A precisão dessas estimativas é fundamental para fazer previsões confiáveis e para inferir a natureza da relação entre as variáveis.

3.1.3.1 Erro Padrão das Estimativas

Para medir a precisão das estimativas dos coeficientes, usam o erro padrão, que fornece uma medida de dispersão das estimativas de coeficientes em torno dos valores verdadeiros dos parâmetros. O erro padrão de β_0 e β_1 é calculado a partir da variabilidade dos resíduos e do número de observações, bem como da variabilidade dos valores da variável independente X

O erro padrão de uma estimativa estatística é uma medida da dispersão ou variabilidade dessa estimativa em relação ao verdadeiro valor do parâmetro na população. No contexto da regressão linear, quando nos referimos ao erro padrão dos coeficientes de regressão (por exemplo, o erro padrão do coeficiente de inclinação β_1 ,estamos falando sobre o quão precisamente podemos estimar a inclinação da relação entre a variável dependente e a variável independente a partir de nossa amostra Fornece também uma medida de dispersão das estimativas de coeficientes em torno dos valores verdadeiros dos parâmetros, essencial para fazer previsões confiáveis e inferir a natureza da relação entre as variáveis.

3.1.3.2 Erro Padrão do Coeficiente de Inclinação e Intercepto

Para a regressão linear simples, o erro padrão do coeficiente de inclinação (β_1) é calculado pela fórmula: β_1

$$SE_{\beta_1} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 / \sum_{i=1}^{n}(x_i - \bar{x})^2}$$

onde:

- y_i são os valores observados da variável dependente,
- $\widehat{y_i}$ são os valores ajustados previstos pelo modelo de regressão,
- x_i são os valores da variável independente,
- $\bar{x}$ é a média dos valores de x_i,
- n é o número total de observações.

A expressão $\sum_{i=1}^{n}\left(y_i - \widehat{y_i}\right)^2$ representa a soma dos quadrados dos resíduos (RSS), e $\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2$ é a soma dos quadrados das diferenças entre os valores de x e a média de x, que mede a variabilidade de x. Já para o erro Padrão do Intercepto β_0 a fórmula é a seguinte:

$$SE_{\beta_0} = SE_{\beta_1} \times \sqrt{\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}}$$

3.1.3.3 Intervalos de Confiança e Testes de Hipóteses

Intervalos de confiança e testes de hipóteses são usados para avaliar a significância estatística dos coeficientes, crucial para determinar a existência de uma relação significativa entre as variáveis. Com base no erro padrão, construímos intervalos de confiança para os coeficientes, que fornecem uma gama de valores plausíveis para os verdadeiros valores dos parâmetros na população. Um intervalo de confiança de 95% para β_1, por exemplo, dá-nos um intervalo dentro do qual podemos estar 95% confiantes de que o verdadeiro valor de β_1 se encontra.

O erro padrão também é usado em testes de hipóteses para avaliar se os coeficientes são significativamente diferentes de zero (ou de outro valor de referência), o que indicaria uma relação significativa entre as variáveis. O teste mais comum é o teste t, onde a estatística de teste é calculada como a estimativa do coeficiente dividida pelo seu erro padrão. Comparamos essa estatística de teste a uma distribuição t para determinar se a relação observada poderia ser devida ao acaso.

3.1.4 Precisão do Modelo de Regressão

3.1.4.1 Coeficiente de Determinação - R^2

O R^2, ou coeficiente de determinação, indica quão bem os valores observados podem ser previstos pelos valores ajustados do modelo, variando de 0 a 1. Onde um valor de 1 indica que o modelo ajusta perfeitamente os dados (todos os pontos estão na linha de regressão), e um valor de 0 indica que o modelo não explica nenhuma variação nos dados. - Cálculo: O R^2 é calculado como a proporção da variância explicada pelo modelo pela variância total dos dados. Matematicamente, é expresso como:

$$R^2 = 1 - \frac{RSS}{TSS}$$

Onde RSS é a soma dos quadrados dos resíduos (a variância não explicada pelo modelo) e TSS é a soma total dos quadrados (a variância total nos dados).

3.1.4.2 Limitações do R^2

- **Sensibilidade ao Número de Preditores:** Uma limitação importante do R^2 é que ele pode aumentar simplesmente adicionando mais variáveis ao modelo, independentemente de essas variáveis terem significância estatística ou não. Isso pode levar a modelos superajustados, onde o modelo se ajusta muito bem aos dados de treino, mas pode ter um desempenho ruim em novos dados.

- Não Indica Precisão Absoluta: O R^2 não fornece uma medida absoluta da "boa"ou "má"qualidade do ajuste do modelo; em vez disso, é uma medida relativa de quão bem o modelo se compara a um modelo nulo que não usa nenhuma variável independente para prever os valores de (Y).

3.1.4.3 R^2 ajustado

Para corrigir isso, usa-se o R^2 ajustado que penaliza o acréscimo de variáveis não significativas, fornecendo uma medida mais precisa da qualidade do ajuste.

Para abordar a limitação do R^2 em contextos com múltiplas variáveis preditoras, o R^2 ajustado é frequentemente utilizado. O R^2 ajustado modifica o cálculo do R^2 para levar em conta o número de preditores no modelo, penalizando a adição de variáveis que não melhoram substancialmente o modelo.O R^2 ajustado é calculado como:

$$1 - \frac{(1 - R^2)\,(n - 1)}{n - p - 1}$$

onde (n) é o número de observações e (p) é o número de preditores. Diferente do R^2, o R^2 ajustado pode diminuir se uma variável que não melhora o modelo for adicionada, fornecendo assim uma medida mais precisa da qualidade do ajuste, especialmente útil em modelos com muitas variáveis

3.2 REGRESSÃO LINEAR MÚLTIPLA

A regressão linear múltipla estende o conceito para múltiplos preditores, permitindo modelar relações mais complexas.

3.2.1 ESTIMAÇÃO DOS COEFICIENTES DE REGRESSÃO

Na regressão linear múltipla, o modelo é expresso como:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

onde (Y) é a variável dependente,$X_1, X_2, \ldots, X_p$ são as variáveis independentes (preditores), $\beta_0, \beta_1, \ldots, \beta_p$ são os coeficientes a serem estimados, e (ϵ) é o termo de erro.

- Método dos Mínimos Quadrados: Para estimar os coeficientes β do modelo, utiliza-se o método dos mínimos quadrados, que busca minimizar a soma dos quadrados dos resíduos (diferença entre os valores observados e os previstos pelo modelo). A solução resultante fornece as estimativas dos coeficientes que melhor se ajustam aos dados no sentido de minimizar a soma dos quadrados dos erros.

- Interpretação dos Coeficientes: Cada coeficiente β_j ($j = 1, 2, \ldots, p$) representa o efeito marginal da variável preditora correspondente X_j sobre a variável resposta (Y), mantendo todas as outras variáveis constantes. Isso significa que β_j quantifica a mudança esperada em (Y) para uma unidade de mudança em X_j, assumindo que os valores das outras variáveis preditoras no modelo permanecem inalterados.

Trazendo os dados de **anúncio de TV**, por exemplo, pode-se montar a seguinte regressão:

$$\textbf{vendas} = \beta_0 + \beta_1 \times \textbf{TV} + \beta_2 \times \textbf{rádio} + \beta_3 \times \textbf{jornal} + \epsilon$$

3.2.1.1 Possibilidade de Relação Não-linear

O modelo de regressão linear múltipla pressupõe uma relação linear entre as variáveis preditoras e a resposta. No entanto, em muitos casos, essa relação pode ser não-linear. Para lidar com a não-linearidade, podem-se incluir termos polinomiais ou interações entre preditores no modelo.

3.2.1.2 Adição ou Remoção de Variáveis do Modelo

A escolha das variáveis a serem incluídas no modelo é crítica. Incluir variáveis irrelevantes pode levar a um modelo superajustado, enquanto excluir variáveis importantes pode resultar em um modelo subajustado. Métodos de seleção de variáveis, como stepwise forward selection, backward elimination, e best subset selection, são abordagens comuns para encontrar o modelo ótimo. Esses métodos de seleção são detalhados no Capítulo (5).

- Stepwise Forward Selection:

  - Descrição: Este método começa com um modelo sem nenhum preditor e, em seguida, adiciona variáveis preditoras uma a uma. Em cada etapa, a variável que proporciona a maior melhoria significativa ao modelo é incluída, até que nenhum outro preditor resulte em uma melhoria significativa.

  - Vantagens: É computacionalmente eficiente e útil quando há um grande número de variáveis.

  - Limitações: Pode não encontrar o melhor modelo se a inclusão de uma variável só se tornar significativa após a inclusão de outra.

- Backward Elimination:

  - Descrição: Este método começa com um modelo que inclui todas as variáveis preditoras possíveis. Em seguida, remove a variável menos significativa (aquela com o maior p-valor) em cada etapa, continuando até que todas as variáveis restantes no modelo sejam significativas.

  - Vantagens: Relativamente simples e eficaz, garantindo que as interações entre as variáveis sejam consideradas desde o início.

  - Limitações: Pode ser computacionalmente intensivo com um grande número de variáveis e pode reter variáveis desnecessárias se a eliminação precoce de outras variáveis reduzir a aparente significância de um preditor.

- Best Subset Selection

  - Descrição: Este método considera todos os possíveis subconjuntos de variáveis preditoras e seleciona o subconjunto que resulta no melhor ajuste do modelo, conforme avaliado por um critério de seleção pré-especificado, como o menor Critério de Informação de Akaike (AIC), o menor Critério de Informação Bayesiano (BIC), ou o maior R^2 ajustado.

  - Vantagens: É o método mais completo, capaz de identificar o modelo ótimo entre

todos os subconjuntos possíveis de preditores.

- Limitações: É computacionalmente inviável para muitas variáveis preditoras devido ao crescimento exponencial do número de modelos possíveis para avaliar.

- Multicolinearidade:

  A multicolinearidade surge quando duas ou mais variáveis preditoras estão altamente correlacionadas, podendo ser detectada pelo fator de inflação da variância (VIF) e tratada pela remoção de variáveis redundantes ou pela regressão de crista.

3.3 OUTRAS CONSIDERAÇÕES NO MODELO DE REGRESSÃO

Incluir o uso de variáveis qualitativas como preditores e diagnósticos de modelo para identificar e resolver problemas como multicolinearidade e heterocedasticidade, essenciais para melhorar a qualidade do modelo de regressão.

3.3.1 Variáveis Qualitativas

Explica como incorporar preditores categóricos no modelo de regressão através de variáveis *dummy*.

3.3.2 Extensões do Modelo Linear

Aborda métodos para modelar relações não-lineares e interações entre variáveis preditoras dentro do framework da regressão linear.

3.3.3 Problemas Potenciais

Discute diagnósticos de regressão e potenciais problemas no ajuste do modelo, incluindo alta alavancagem, pontos influentes e colinearidade, oferecendo estratégias para resolvê-los.

4 CLASSIFICAÇÃO

4.1 UMA VISÃO GERAL DE CLASSIFICAÇÃO

A classificação é um método de aprendizado supervisionado que é fundamental para prever respostas categóricas a partir de um ou mais preditores. Essa técnica tem ampla aplicabilidade, sendo usada em várias áreas práticas, como diagnóstico médico, reconhecimento de imagem e previsão de inadimplência de crédito, o que demonstra sua importância em contextos diversificados.

No processo de classificação, uma função f é formulada para mapear um conjunto de preditores $X = (X_1, X_2, \ldots, X_p)$ em uma resposta Y. Especificamente, em casos de classificação binária, como a determinação de inadimplência, f pode modelar a probabilidade de um evento ser classificado como '1' (inadimplente), o que é matematicamente representado por $P(Y = 1|X) = f(X)$.

Para desenvolver um modelo de classificação eficaz, os dados são geralmente divididos entre um conjunto de treinamento, onde o modelo é criado, e um conjunto de teste, usado para avaliar o modelo. A precisão do modelo, geralmente medida pela proporção de previsões corretas, é o critério principal para avaliar seu desempenho, embora outras métricas, como a área sob a curva ROC, também sejam relevantes para análises mais detalhadas.

A curva ROC (Receiver Operating Characteristic) é uma ferramenta gráfica usada para avaliar o desempenho de modelos de classificação binária. Ela é particularmente útil para visualizar a capacidade de um modelo em distinguir entre duas classes alvo. A curva ROC é construída ao plotar a Taxa de Verdadeiro Positivo (TPR - True Positive Rate) contra a Taxa de Falso Positivo (FPR - False Positive Rate) em diferentes limiares de classificação.

4.1.1 Métricas de Avaliação

4.1.1.1 Taxa de Verdadeiro Positivo (TPR)

Também conhecida como sensibilidade ou recall, a TPR mede a proporção de positivos reais que são corretamente identificados pelo modelo. Matematicamente, ela é definida como:

$$\text{TPR} = \frac{TP}{TP + FN} \tag{14}$$

onde: - TP são os Verdadeiros Positivos: o número de positivos que foram corretamente identificados pelo modelo. - FN são os Falsos Negativos: o número de positivos que foram incorretamente identificados como negativos pelo modelo.

4.1.1.2 Taxa de Falso Positivo (FPR)

A FPR mede a proporção de negativos reais que são incorretamente identificados como positivos. É calculada como:

$$\text{FPR} = \frac{FP}{FP + TN} \tag{15}$$

onde: - FP são os Falsos Positivos: o número de negativos que foram incorretamente

identificados como positivos pelo modelo. - TN são os Verdadeiros Negativos: o número de negativos que foram corretamente identificados como negativos pelo modelo.

4.1.1.3 Construção da Curva ROC

Para construir a curva ROC, calcula-se TPR e FPR para vários limiares de decisão do modelo. Um limiar de decisão é um ponto de corte que separa as previsões de classe positiva das previsões de classe negativa. Ao variar esse limiar, obtêm-se diferentes valores de TPR e FPR, que são então plotados em um gráfico com FPR no eixo horizontal e TPR no eixo vertical.

A área sob a curva ROC (em inglês, AUC - *Area Under the Curve*) é uma medida única que resume o desempenho do modelo, independentemente de qualquer limiar específico. Um modelo perfeito terá uma AUC de 1.0, indicando uma capacidade perfeita de separar as duas classes, enquanto um modelo que faz previsões aleatórias terá uma AUC de 0.5.

A curva ROC é especialmente valorizada porque fornece uma medida compreensiva de desempenho em diferentes níveis de sensibilidade e especificidade, ajudando a escolher o melhor ponto de equilíbrio para a aplicação específica do modelo.

4.1.2 POR QUE NÃO USAR REGRESSÃO

A regressão linear pode não ser adequada para problemas de classificação que envolvem a previsão de uma resposta categórica. Em contextos de classificação, as respostas são tipicamente categóricas, como "sim/não"ou "aprovado/reprovado", diferindo substancialmente das variáveis contínuas para as quais a regressão linear é projetada.

4.1.2.1 Problemas com a Regressão Linear em Classificação

4.1.2.2 Modelagem Inadequada de Probabilidades

A regressão linear pode, tecnicamente, ser aplicada para prever valores categóricos ao codificar categorias como números (por exemplo, 0 e 1 para duas classes). No entanto, este método modela diretamente a resposta como uma função linear dos preditores, o que pode resultar em previsões fora do intervalo [0,1], valores que não são probabilisticamente válidos para categorias.

4.1.2.3 Extrapolando Fora dos Limites Adequados

Quando se usa regressão linear para prever categorias binárias, o modelo pode prever valores abaixo de 0 ou acima de 1. Isso ocorre porque a regressão linear não restringe a faixa de saída, levando a previsões que não têm interpretação prática como probabilidades.

4.1.2.4 Consequências do Uso Incorreto

A utilização da regressão linear em dados categóricos pode levar a interpretações errôneas e a uma confiança equivocada na capacidade do modelo de discriminar entre as classes.

4.1.2.5 Alternativas à Regressão Linear para Classificação

Diante dessas limitações, outros modelos como a regressão logística e a análise discriminante são recomendados para classificação. A regressão logística, em particular, modela a probabilidade de a resposta pertencer a uma classe particular, fornecendo saídas entre 0 e 1, que são interpretações válidas de probabilidades. A equação da regressão logística é:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p \tag{16}$$

onde p é a probabilidade de a observação pertencer à classe 1.

Esta análise ilustra a importância de escolher o modelo correto de acordo com a natureza dos dados e o tipo de resposta envolvida, ressaltando que, para dados categóricos, modelos que operam dentro dos limites de probabilidades válidas são essenciais para resultados confiáveis e interpretações corretas.

4.2 REGRESSÃO LOGÍSTICA

A Regressão Logística,é um método estatístico que é amplamente utilizado para modelar a probabilidade de uma variável resposta binária com base em uma ou mais variáveis preditoras. Diferente da regressão linear, a regressão logística é ideal para situações onde a resposta é categórica, fornecendo um framework robusto para estimar as probabilidades de resultados de classificação.

4.2.1 Modelo Logístico

O modelo logístico é uma técnica estatística fpara prever a ocorrência de um evento com base em uma série de preditores. O modelo logístico é particularmente adequado para situações onde a variável resposta é binária, como "sim"ou "não", "sucesso"ou "falha". Este modelo é uma forma de regressão logística, que permite analisar como diferentes variáveis preditoras influenciam a probabilidade de um evento específico.

O modelo logístico é formulado como:

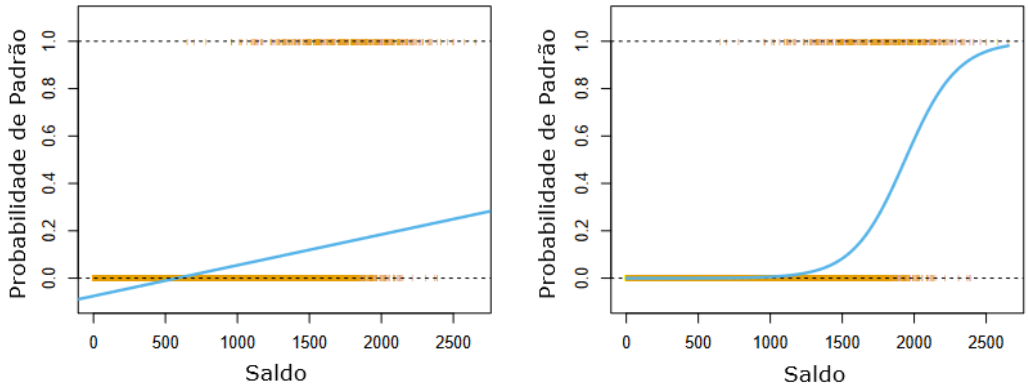

Figura 3: Classificação usando os dados **Padrão**. Esquerda: probabilidade estimada de **padrão** usando regressão linear. Note que algumas probabilidades estimadas são negativas! Direita: probabilidades preditas de **padrão** usando regressão logística. Perceba que todas as probabilidades ficam entre 0 e 1.

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 \tag{17}$$

onde p representa a probabilidade de ocorrência do evento de interesse. O lado esquerdo da equação, conhecido como logito, é o logaritmo natural do odds (chance) de p ocorrer versus não ocorrer. Os β_i são coeficientes que representam a influência de cada variável preditora X_i na probabilidade p.

Cada coeficiente no modelo logístico quantifica o efeito de mudar uma unidade na variável preditora correspondente, mantendo todas as outras constantes, sobre o logaritmo do odds de ocorrência do evento. Um coeficiente positivo β_i indica que conforme X_i aumenta, a probabilidade p de ocorrência do evento também aumenta, mantendo todas as outras variáveis fixas. Inversamente, um coeficiente negativo sugere uma diminuição na probabilidade com o aumento de X_i.

A probabilidade de ocorrência do evento, p, pode ser expressa de volta na escala de [0,1] utilizando a transformação logística:

$$p = \frac{e^{\beta_0+\beta_1 X_1}}{1+e^{\beta_0+\beta_1 X_1}}$$

Essa fórmula transforma a linearidade dos preditores e seus coeficientes em uma probabilidade limitada entre 0 e 1, tornando-se assim adequada para modelar respostas binárias.

Como exemplo, pode-se usar os dados **Padrão**. Nesse caso, a regressão logística modela a probabilidade de **padrão**:

4.2.1.1 Aplicações Práticas do Modelo Logístico

A aplicabilidade do modelo logístico é vasta e abrange desde previsões médicas, como a probabilidade de um paciente desenvolver uma doença específica com base em características como idade, peso e histórico familiar, até aplicações em finanças, como determinar a probabilidade de inadimplência em empréstimos. O modelo é valorizado por sua flexibilidade e capacidade de fornecer insights quantitativos sobre a influência dos preditores.

4.2.1.2 Estimativa dos Coeficientes

A estimativa dos coeficientes no modelo de regressão logística é um processo fundamental para determinar como as variáveis preditoras influenciam a probabilidade de um evento. Os coeficientes são estimados usando o método de máxima verossimilhança, que busca os valores de coeficientes que maximizam a probabilidade de observar os dados dados esses coeficientes.

4.2.1.3 Método de Máxima Verossimilhança

O método de máxima verossimilhança envolve a construção de uma função de verossimilhança, que expressa a probabilidade dos dados observados sob diferentes valores para os coeficientes do modelo. A função de verossimilhança para a regressão logística é baseada na probabilidade condicional dos dados, dada uma especificação dos parâmetros do modelo. A função é dada por:

$$L(\beta, \beta_0) = \prod_{i:y_i=1} p_i(x_i) \prod_{i:y_i=1} (1 - p_i(x_i)) \tag{18}$$

onde p_i é a probabilidade modelada de ocorrência do evento para a observação i, y_i é o valor observado da variável resposta (0 ou 1), e β são os coeficientes a serem estimados. Esta função é então maximizada em relação aos coeficientes. A maximização é geralmente realizada usando métodos numéricos, como o algoritmo de Newton-Raphson, porque as expressões analíticas diretas para os coeficientes máximos verossímeis raramente são viáveis devido à complexidade do modelo.

4.2.1.4 Significado dos Coeficientes Estimados

Os coeficientes obtidos por este método quantificam a mudança no logaritmo do *odds* do evento de interesse associado a uma unidade de mudança na variável preditora correspondente, mantendo todas as outras constantes. Estes coeficientes fornecem insights valiosos sobre a relação entre as variáveis preditoras e a probabilidade do evento, permitindo interpretações significativas no contexto da aplicação em questão.

4.2.2 REGRESSÃO LOGÍSTICA MÚLTIPLA

A regressão logística múltipla é uma extensão da regressão logística simples que permite incluir múltiplos preditores, tornando-a uma ferramenta poderosa para analisar a influência de várias variáveis independentes sobre uma variável resposta binária. Este método é amplamente utilizado em campos como medicina, economia e ciências sociais, onde é necessário avaliar o impacto simultâneo de várias variáveis.

Modelo de Regressão Logística Múltipla: A regressão logística múltipla pode ser express pela equação logit, que relaciona os logaritmos dos odds da probabilidade de ocorrência do evento a um conjunto de preditores:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p \tag{19}$$

Aqui, p representa a probabilidade de ocorrência do evento de interesse (por exemplo, sucesso ou falha), β_0 é o intercepto, β_i são os coeficientes para cada variável preditora X_i, e p é o número total de preditores no modelo.

Interpretação dos Coeficientes: Os coeficientes β_i representam a mudança no logaritmo dos odds de p em resposta a uma unidade de mudança na variável preditora X_i, mantendo todas as outras variáveis constantes. Um valor positivo de β_i indica que à medida que X_i aumenta, a probabilidade de p ocorrer também aumenta, enquanto um valor negativo indica o oposto.

Estimação dos Coeficientes: Os coeficientes são geralmente estimados através do método de máxima verossimilhança, que busca valores para β que maximizam a probabilidade de observar os dados dados esses parâmetros. Pr é a probabilidade modelada de ocorrência do evento, calculada como:

$$\log\left(\frac{\Pr(Y = k \mid X = x)}{\Pr(Y = K \mid X = x)}\right) = \beta_{k0} + \beta_{k1} x_1 + \cdots + \beta_{kp} x_p$$

Aplicabilidade do Modelo: A regressão logística múltipla é particularmente útil quando se deseja entender o impacto relativo de várias variáveis em uma resposta binária. Ela pode ajudar a identificar quais fatores são os mais significativos na previsão do evento de interesse e quão forte é a associação de cada fator com a probabilidade de ocorrência desse evento.

Este modelo é essencial para a construção de um entendimento robusto e detalhado das relações entre múltiplos fatores e uma resposta binária, permitindo decisões informadas

baseadas em análises complexas de dados multidimensionais.

4.2.3 REGRESSÃO LOGÍSTICA POLINOMIAL

A regressão logística multinomial é uma extensão do modelo de regressão logística que é aplicável quando a variável resposta tem mais de duas categorias. Diferente da regressão logística binária, que é usada para respostas com duas categorias (como "sim"ou "não"), a regressão logística multinomial permite que se analise respostas categóricas com múltiplos níveis, como "baixo", "médio", e "alto".

Modelo de Regressão Logística Multinomial: No modelo de regressão logística multinomial, a probabilidade de cada categoria da resposta depende linearmente das variáveis independentes. A fórmula para este modelo é uma generalização da equação logística e pode ser expressa como:

$$\log\left(\frac{P(Y=k)}{P(Y=K)}\right) = \beta_{k0} + \beta_{k1}X_1 + \beta_{k2}X_2 + \cdots + \beta_{kp}X_p \tag{20}$$

onde: - $P(Y = k)$ é a probabilidade de a resposta pertencer à categoria k. - K é a categoria de referência contra a qual todas as outras categorias são comparadas. - β_{ki} são os coeficientes a serem estimados para cada categoria k (exceto para a categoria de referência).

Estimação dos Coeficientes: Assim como na regressão logística binária, os coeficientes na regressão logística multinomial são geralmente estimados usando o método de máxima verossimilhança. Este método maximiza a função de verossimilhança, que é a probabilidade de observar os dados dados os parâmetros do modelo. A solução geralmente requer o uso de software estatístico, pois não há soluções analíticas simples e é necessário resolver um problema de otimização.

Interpretação dos Resultados: A interpretação dos coeficientes de uma regressão logística multinomial é similar à de uma regressão logística binária, mas com uma nuance adicional devido à presença de múltiplas categorias. Cada coeficiente indica como a mudança em uma variável preditora influencia a log-chance de uma categoria específica em relação à categoria de referência. Um coeficiente positivo sugere que um aumento na variável preditora está associado a uma maior chance da categoria em questão comparada à categoria de referência, enquanto um coeficiente negativo indica o contrário.

Aplicações Práticas A regressão logística multinomial é utilizada em diversos campos, incluindo pesquisa de mercado, onde pode ajudar a determinar fatores que influen-

ciam a escolha do consumidor entre várias marcas ou produtos; em saúde pública para analisar fatores de risco para diferentes categorias de doenças; e em ciência política para examinar as preferências eleitorais entre múltiplos candidatos ou partidos.

Este modelo oferece uma forma flexível e poderosa de modelar respostas categóricas com várias categorias, permitindo aos pesquisadores e analistas explorar complexidades nos dados que vão além das análises binárias simples.

4.3 MODELOS GENERATIVOS PARA CLASSIFICAÇÃO

Os Modelos Gerativos para Classificação são uma classe de algoritmos usados para modelar como os dados são gerados ao identificar distribuições que caracterizam cada classe. Esses modelos não só preveem a classe de uma nova observação, mas tentam entender a distribuição dos dados dentro de cada classe para fazer a previsão. Este tópico é dividido em quatro subseções, detalhando a Análise Discriminante Linear (LDA) para um e múltiplos preditores, a Análise Discriminante Quadrática (QDA) e o Naive Bayes.

4.3.1 Análise Discriminante Linear para $p = 1$

A Análise Discriminante Linear [22] para um único preditor é um método que assume que as observações de cada classe são distribuídas normalmente com uma média específica da classe e uma variância comum a todas as classes. A LDA estima a média e a variância dessas distribuições normais usando os dados de treinamento. O modelo é usado para determinar um limiar que melhor separa as classes, minimizando a probabilidade de classificação incorreta. A formulação matemática é dada por:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k) \tag{21}$$

onde μ_k é a média da classe k, σ^2 é a variância comum, e π_k é a probabilidade prévia da classe k.

4.3.2 Análise Discriminante Linear para $p > 1$

Quando há mais de um preditor, a LDA é estendida para acomodar vetores de médias para cada classe e uma matriz de covariância comum. Isso permite que o modelo avalie a relação linear multivariada entre preditores e classes. A função discriminante para múltiplos preditores é:

$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log(\pi_k) \tag{22}$$

[22] Em inglês, *Linear Discriminant Analysis* (LDA)

onde Σ é a matriz de covariância comum entre as classes.

4.3.3 Análise Discriminante Quadrática

A QDA [23] relaxa a suposição da LDA de que as variâncias são iguais em todas as classes, permitindo que cada classe tenha sua própria matriz de covariância. Isso proporciona flexibilidade para capturar diferentes formas e orientações das distribuições de classes nos dados. A função discriminante para a QDA é:

$$\delta_k(x) = -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log(\pi_k) \tag{23}$$

onde Σ_k é a matriz de covariância específica da classe k.

4.3.4 *Naive Bayes*

modelo que assume independência condicional entre cada par de características dentro de uma classe. Apesar de sua simplicidade, o *Naive Bayes* pode ser surpreendentemente eficaz e é especialmente útil quando a dimensionalidade dos dados é alta. A probabilidade de uma observação pertencer a uma classe é dada por:

$$P(y=k|x) \propto P(x|y=k)P(y=k)$$

onde $P(x|y=k)$ é o produto das probabilidades individuais de cada característica, assumindo independência.

4.3.5 Limitações dos Modelos Generativos de Classificação

Ao explorar os métodos de classificação como Naive Bayes, Análise Discriminante Linear (LDA) e Análise Discriminante Quadrática (QDA), é essencial entender suas limitações específicas, que podem afetar o desempenho do modelo em diferentes cenários de dados.

4.3.5.1 Limitações do Naive Bayes

- Independência das características: Naive Bayes opera sob a suposição de que todas as características são independentes umas das outras, dado o resultado da classificação. Essa suposição simplista pode não ser válida em muitos casos práticos onde as características são interdependentes, como em cenários de processamento de linguagem natural, afetando negativamente a precisão do modelo.

[23] em inglês, *quadratic Discriminant Analysis* (QDA)

- Distribuição de probabilidade: Este método também assume distribuições específicas para as características (geralmente Gaussiana, multinomial ou Bernoulli). Se a distribuição real dos dados for diferente, o desempenho do modelo pode ser comprometido.

4.3.5.2 Limitações da LDA

- Homogeneidade de variâncias: LDA pressupõe que todas as classes compartilham a mesma matriz de covariância. Quando as variâncias são significativamente diferentes entre as classes, este pressuposto é violado, e o modelo pode não realizar classificações precisas, pois não consegue capturar a estrutura verdadeira dos dados.

- Distribuições normais: A eficácia da LDA também depende da suposição de que as características são normalmente distribuídas dentro de cada classe. Dados com distribuições assimétricas ou não normais podem levar a uma má performance do modelo.

4.3.5.3 Limitações da QDA

- Complexidade do modelo: A flexibilidade da QDA para permitir uma matriz de covariância distinta para cada classe aumenta a complexidade do modelo. Embora isso possa ajudar a modelar melhor as características dos dados, também aumenta o risco de overfitting, especialmente em conjuntos de dados com alta dimensionalidade ou com um número limitado de amostras.

- Requisitos de dados: A necessidade da QDA de estimar uma matriz de covariância separada para cada classe exige um volume maior de dados. Em conjuntos de dados menores, a variabilidade nas estimativas de covariância pode ser alta, resultando em uma classificação imprecisa.

4.4 COMPARAÇÃO DE MÉTODOS DE CLASSIFICAÇÃO

A comparação de métodos de classificação envolve a análise de diferentes algoritmos de aprendizado de máquina para determinar qual é mais eficaz para um problema específico ou conjunto de dados. Esta análise considera diversos fatores, como a complexidade do modelo, a capacidade de manejar grandes volumes de dados, a sensibilidade a outliers, e o desempenho prático em aplicações reais. Métodos comuns como a regressão logística, a análise discriminante e o Naive Bayes são frequentemente avaliados sob suas suposições fundamentais:

Regressão Logística: É eficaz quando as relações entre as variáveis preditoras e a resposta são aproximadamente lineares e os dados são linearmente separáveis.

Análise Discriminante Linear (LDA) e Quadrática (QDA): Estes métodos supõem distribuições normais dos dados por classe. A LDA é mais eficiente quando as classes compartilham variâncias semelhantes, enquanto a QDA pode acomodar classes com covariâncias distintas.

Naive Bayes: Apesar de sua suposição simplista de independência entre os recursos, este método pode ser altamente eficaz, especialmente em tarefas de classificação de texto, devido à sua capacidade de manejar um grande número de recursos

4.4.1 Comparação Analítica

Na comparação analítica de métodos de classificação, a ênfase é colocada em entender as diferenças teóricas e práticas entre diversos algoritmos disponíveis para classificação, o que é fundamental para identificar o método mais apropriado para uma aplicação específica, levando em conta as características dos dados e os objetivos da análise.

Análise Teórica: A análise teórica explora as bases matemáticas e estatísticas de cada método. Por exemplo, a regressão logística é valorizada por fornecer estimativas probabilísticas diretas, úteis em contextos onde as probabilidades das classificações são importantes. Por outro lado, métodos como máquinas de vetores de suporte (SVM) são destacados pela sua capacidade de maximizar a margem entre as classes, ideal para dados com possibilidade de separação clara.

Comparação de Suposições: Cada algoritmo carrega suposições específicas sobre a distribuição dos dados. A Análise Discriminante Linear (LDA) e a Quadrática (QDA) pressupõem distribuições normais das classes, mas a QDA adiciona a flexibilidade de permitir matrizes de covariância distintas para cada classe. O Naive Bayes, por sua vez, assume independência entre as características, o que pode ser uma simplificação excessiva, mas contribui para a eficiência computacional do modelo.

Performance em Dados Reais: A eficácia dos métodos é geralmente testada em conjuntos de dados reais, utilizando métricas como precisão, recall e F1-score. As métricas de precisão, recall e F1-score são amplamente utilizadas para avaliar o desempenho de modelos de classificação. Cada uma dessas métricas fornece insights distintos sobre a qualidade das previsões feitas por um modelo, ajudando a entender como ele se comporta em diferentes aspectos da classificação.

Precisão

A precisão é uma métrica que mede a proporção de identificações positivas feitas pelo modelo que foram realmente corretas. Ela é calculada pela fórmula:

$$\text{Precisão} = \frac{\text{Verdadeiros Positivos (TP)}}{\text{Verdadeiros Positivos (TP)} + \text{Falsos Positivos (FP)}} \tag{24}$$

onde:

- Verdadeiros Positivos (TP): são os casos em que o modelo corretamente prevê a classe positiva. - Falsos Positivos (FP): são os casos em que o modelo incorretamente prevê a classe positiva.

Essa métrica é particularmente importante em situações onde o custo de um falso positivo é alto, como no diagnóstico médico, onde um diagnóstico falso de uma doença pode levar a tratamentos desnecessários e estresse para o paciente.

Recall

O recall, também conhecido como sensibilidade, mede a capacidade do modelo de identificar todas as instâncias relevantes de uma classe específica. É calculado por:

$$\text{Recall} = \frac{\text{Verdadeiros Positivos (TP)}}{\text{Verdadeiros Positivos (TP)} + \text{Falsos Negativos (FN)}} \tag{25}$$

onde:

- Falsos Negativos (FN) são os casos em que o modelo falha em identificar a classe positiva.

O recall é crucial em contextos onde é essencial capturar todas as possíveis ocorrências da classe de interesse, como na detecção de fraudes ou em condições médicas graves onde falhar em identificar um caso pode ter consequências sérias.

F1-Score

O F1-score é a média harmônica entre precisão e recall, proporcionando um único indicador que leva em conta tanto a precisão quanto o recall. O F1-score é particularmente útil quando é necessário encontrar um equilíbrio entre precisão e recall, e é calculado por:

$$\text{F1-Score} = 2 \cdot \frac{\text{Precisão} \times \text{Recall}}{\text{Precisão} + \text{Recall}} \tag{26}$$

Esse score é especialmente valioso em situações onde classes desbalanceadas podem fazer com que outras métricas apresentem uma visão distorcida do desempenho do modelo. O F1-score ajuda a balancear os efeitos de classes desproporcionais, oferecendo uma visão mais realista da eficácia do modelo em classificar corretamente os dados.

Utilizando essas métricas juntas, é possível obter uma visão abrangente da performance de um modelo de classificação, permitindo ajustes e otimizações baseados em evidências claras de como o modelo está performando em termos de captura e precisão da classificação desejada.

Essas comparações empíricas são cruciais para ilustrar como cada método se comporta em condições variadas, como desbalanceamento de classes (quando as frequências das classes em um conjunto de dados não são aproximadamente iguais, resultando em uma predominância de uma ou mais classes em detrimento de outras.), presença de ruído e outliers.

Análise de Sensibilidade A sensibilidade dos métodos a variações nos dados, como outliers ou escalonamento inadequado, também é analisada. Alguns métodos podem ser mais vulneráveis a tais perturbações, enquanto outros mostram maior robustez. Essa análise é vital para escolher um método adequado, especialmente em ambientes práticos onde essas condições são comuns.

4.4.2 Comparação Empírica

A comparação empírica dos métodos de classificação é realizada através de estudos de caso e análises práticas, aplicando-se diversos métodos a conjuntos de dados reais para avaliar seu desempenho. Esta abordagem permite uma avaliação direta de como diferentes algoritmos de classificação operam sob condições variadas e com dados que refletem problemas reais, proporcionando insights valiosos sobre a aplicabilidade prática de diferentes métodos de classificação.

Os aspectos considerados na comparação empírica incluem:

- Desempenho do Modelo: Avaliado em termos de acurácia, precisão, recall, F1-score, e área sob a curva ROC (AUC).

- Robustez: Capacidade dos modelos de manter um desempenho consistente em diferentes conjuntos de dados ou frente a perturbações dos dados.

- Sensibilidade a Parâmetros: Impacto dos ajustes nos parâmetros no desempenho dos modelos. **- Velocidade de Treinamento e Predição:** Relevante em aplicações em tempo real ou ao lidar com grandes volumes de dados. **- Capacidade de Lidar com Desbalanceamento de Classes:** Crucial em contextos onde algumas classes são menos representadas do que outras.

Para medir e comparar esses fatores, utilizam-se métodos estatísticos rigorosos como testes de hipóteses, análises de variância (ANOVA) e técnicas de correlação, que ajudam

a quantificar a precisão e a confiabilidade dos resultados obtidos. Além disso, a validação cruzada é empregada como uma técnica fundamental para avaliar a generalidade dos modelos de classificação. Neste processo, o conjunto de dados é dividido em "k"subconjuntos ou "folds", com o modelo sendo treinado em "k-1"desses folds e testado no fold restante, processo que se repete "k"vezes com cada fold servindo uma vez como conjunto de teste. A média do desempenho do modelo em todos os folds é usada como a estimativa de seu desempenho, fornecendo uma medida robusta que também ajuda a prevenir o overfitting.

5 MÉTODOS DE REAMOSTRAGEM

Métodos de reamostragem são uma ferramenta indispensável na estatística moderna, os quais envolvem a repetida tomada de amostras de um conjunto de treinamento e ajuste do modelo para cada amostra a fim de obter informações adicionais sobre o modelo.

Abordagens de reamostragem podem ser computacionalmente custosas, porque elas envolvem ajustar o mesmo método estatístico várias vezes utilizando diferentes *subsets* do conjunto de treinamento. Isso, entretanto, não é um problema geralmente dado os recentes avanços do poder computacional.

Nesse capítulo, iremos discutir dois dos mais comumente usados métodos de reamostragem: validação cruzada[24] e *bootstrap*[25]

5.1 VALIDAÇÃO CRUZADA

No capítulo 2, nós discutimos a distinção entre a taxa de erro de teste e a taxa de erro de treinamento. O erro de teste é a média do erro que resulta da utilização de um método estatístico para predizer a resposta em uma nova observação. Em contraste, o erro de treino pode ser facilmente calculado aplicando o método de aprendizado de máquina nas observações usadas no treinamento.

Na ausência de um grande conjunto de teste, algumas técnicas podem ser utilizadas para estimar o valor da taxa usando os dados de treino disponíveis. Nessa seção, consideramos uma classe de métodos que estimam a taxa de erro de teste ao separar um subconjunto das observações de treinamento do processo de ajuste, e então aplicar um método de aprendizado de máquina para essas observações separadas.

[24]É mais usual falar *cross-validation*.
[25]Não há uma tradução oficial para esse tipo de método, então será utilizado o termo em inglês.

5.1.1 Abordagem de conjunto de validação

Suponha que queremos estimar o erro de teste associado com o ajuste de um modelo particular dado um conjunto de observações. A abordagem de conjunto de validação é uma simples estratégia para essa tarefa.

Essa estratégia consiste em aleatoriamente dividir as observações disponíveis em duas partes, o conjunto de treinamento e o *validation set* ou *hold-out set*. O modelo é ajustado com conjunto de treino e, então, o modelo é usado para predizer as respostas para as observações no conjunto de validação. A resultante taxa de erro do conjunto de validação, tipicamente calculada por MSE nas respostas quantitativas, fornece uma estimativa da taxa de erro de teste.

A abordagem de *validation set* é conceitualmente simples e de fácil implementação, porém ela tem duas potenciais desvantagens:

1. A estimativa da taxa de teste de erro a partir de um conjunto de validação pode ser altamente variável, dependendo precisamente de quais observações estão incluídas no conjunto de treino e de validação.

2. Na abordagem de validação, apenas um subconjunto das observações é utilizado para ajustar o modelo. Uma vez que os métodos estatísticos tendem a performar pior quando treinados em menores conjuntos de observações, isso sugere que a taxa de erro de validação pode tender a superestimar a taxa de erro de teste para um modelo com o todas as observações.

Nas próximas subseções, apresentaremos a validação cruzada, um refinamento da abordagem de conjunto de validação.

5.1.2 *Leave-One-Out Cross-Validation*

Leave-one-out[26] *cross-validation* (LOOCV) é intimamente relacionado com a abordagem de conjunto de validação, mas ele tenta tratar as desvantagens.

Como a abordagem anterior, LOOCV envolve dividir o conjunto de observações em duas partes. Entretanto, em vez de criar dois subconjuntos de tamanho semelhante, uma única observação (x_1, y_1) é usada para o conjunto de validação, e as observações restantes $\{(x_2, y_2), \ldots, (x_n, y_n)\}$ fazem o conjunto de treinamento. O modelo é ajustado com as $n - 1$ observações, e a predição $\hat{y}_1$ é feita para a observação separada, usando o valor x_1.

Como (x_1, y_1) não foi usado para o treinamento, $MSE_1 = (y_1 - \hat{y}_1)^2$ fornece uma estimativa aproximadamente não enviesada para o erro de teste. Entretanto, é uma

[26]Não há uma tradução oficial para esse termo, mas ele significa literalmente "deixar um de fora"

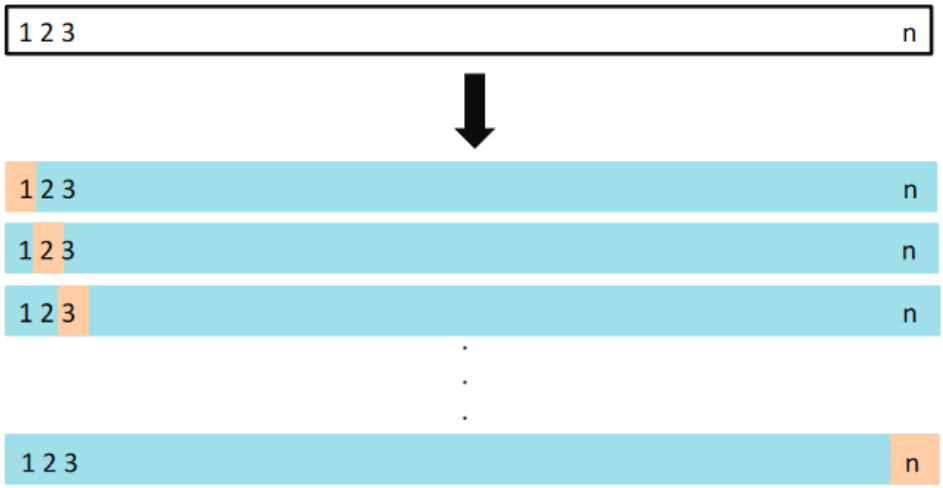

Figura 4: Uma ilustração do *LOOCV*. Um conjunto de n dados é repetidamente dividido entre conjunto de treinamento (em azul), que contém todas menos uma observação, e o conjunto de validação (em bege), que contém somente uma observação. O erro de teste é, então, estimado ao fazer a média dos n MSEs. O primeiro conjunto de treinamento contém todas menos a observação 1, o segundo conjunto contém todas menos a observação 2, e assim sucessivamente.

estimativa insatisfatória, uma vez que é altamente variável.

Nesse sentido, nós repetimos o procedimento escolhendo a observação (x_2, y_2) para a validação. Repetindo esse processo n vezes produz n erros quadráticos: $MSE_1, \ldots, MSE_n$. A estimativa LOOCV para MSE de teste é a média das n estimativas de erro de teste:

$$CV_{(n)} = \frac{1}{n} \sum_{i=1}^{n} MSE_i. \tag{27}$$

LOOCV tem algumas vantagens em relação à abordagem de conjunto de validação:

1. Tem bem menos viés. Ao ajustar repetidamente o modelo, o LOOCV tende a não superestimar a taxa de erro de teste tanto quanto a abordagem *validation set* faz.

2. Em contraste com a abordagem anterior - a qual gerava resultados diferentes dado a aleatoriedade das divisões de treino e validação - utilizar LOOCV múltiplas vezes vai resultar nos mesmos resultados.

LOOCV tem o potencial de ser custoso para implementar[27], já que o modelo tem que ser treinado n vezes. Contudo, é um método bem geral e pode ser utilizado em qualquer

[27] O livro traz fórmula mágica que faz com que o LOOCV tenha o mesmo custo computacional de um modelo convecional. Entretanto, ele só funciona para mínimos quadrados lineares ou regressões polinomiais.

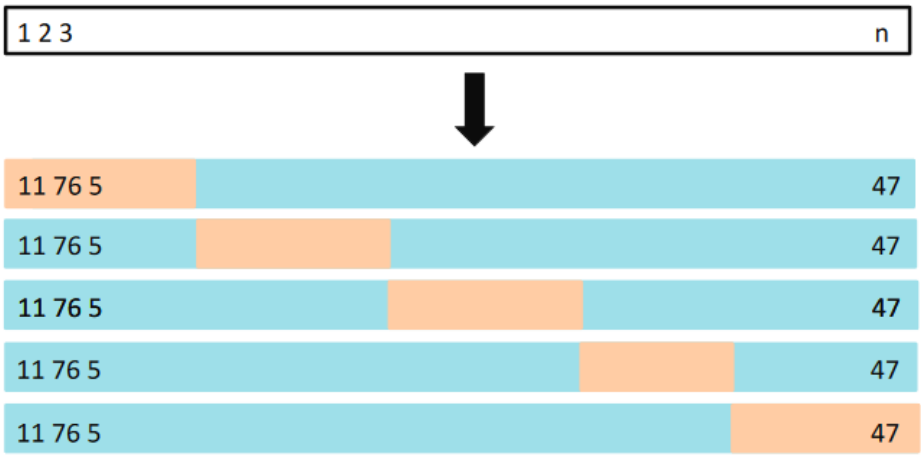

Figura 5: Uma ilustração de 5-*Fold CV*. Um conjunto de n dados é aleatoriamente dividido em 5 grupos não sobrepostos. Cada um desses 1/5 age como um conjunto de validação (em bege), e os 4/5 remanescentes como o conjunto de treinamento (em azul). O erro de teste é estimado ao fazer a média das cinco estimativas MSE resultantes.

modelo preditivo.

5.1.3 *k-Fold Cross-Validation*

Uma alternativa para o LOOCV é o *k-fold CV*[28]. Essa abordagem envolve dividir aleatoriamente um conjunto de observações em k grupos, camadas ou *folds* de tamanho aproximadamente igual. A primeira camada é tratada como *validation set*, e o método é ajustado com as $k-1$ camadas. O MSE_1 é então computado com as observações do conjunto de validação. Com um conjunto de validação diferente, esse processo é repetido k vezes, produzindo k MSE. Por conseguinte, a estimativa *k-fold CV* é calculada pela média desses valores,

$$CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} MSE_i. \tag{28}$$

Não é difícil de ver que o LOOCV é um caso especial de *k-fold CV* no qual k é igual a n. Na prática, é comum utilizar $k = 5$ ou $k = 10$. A maior vantagem aparente disso é o custo computacional: caso o valor de n seja muito grande, o LOOCV vai demandar muito poder do computador. Em contrapartida, ajustar o modelo 5 ou 10 vezes não representa um custo muito grande.

[28]Novamente, não há uma tradução oficial desse termo, mas o significado literal é validação cruzada com k camadas/dobras.

5.1.4 O paradigma do Viés e Variância para *k-Fold Cross Validation*

Foi mencionado na subseção anterior que uma vantagem do *k-fold CV* é o custo computacional, entretanto existe uma vantagem menos óbvia e potencialmente mais importante que isso: o *k-fold CV* resulta frequentemente em estimativas mais acuradas da taxa de erro de teste do que o LOOCV.

A abordagem de conjunto de validação da subseção 5.1.1 menciona como tal abordagem pode superestimar a taxa de erro de teste. Não é difícil ver, em contrapartida, que o LOOCV nos dá estimativas aproximadamente não enviesadas. Performando um *k-fold CV* com, digamos, $k = 5$ ou $k = 10$, vai resultar em um nível intermediário de *bias*. Portanto, da perspectiva de redução de viés, fica claro que o LOOCV é o método mais adequado.

Contudo, sabemos que o viés não é a única fonte de preocupação: devemos considerar também a variância. E acontece que o LOOCV tem uma maior variância que o *k-fold CV*, uma vez que os resultados dos modelos ajustados do LOOCV possuem alta correlação entre si.

Resumindo, há um *bias-variance trade-off* associado com a escolha do k no *k-fold cross validation*. Usualmente, é escolhido $k = 5$ ou $k = 10$ para performar o *k-fold CV*, visto que empiricamente esse valores geram uma estimativa de taxa de erro de teste nem com viés extremamente alto nem com variância muito alta.

5.1.5 *Cross Validation* em problemas de Classificação

Até o momento, ilustramos o uso de validação cruzada no cenário de regressão onde o Y é quantitativo, e assim usamos o MSE para quantificar o erro de teste. Mas validação cruzada pode ser muito útil para uma abordagem em classificação onde Y é qualitativo.

Nesse cenário, *cross-validation* funciona da mesma forma discutida anteriormente, com a exceção de que - em vez de usar MSE para quantificar o erro de teste - usamos o número de observações classificadas incorretamente. Por exemplo, em classificação, a taxa de erro do LOOCV toma a seguinte forma:

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} Err_i, \tag{29}$$

onde $Err_i = I(y_i \neq \hat{y}_i)$. As taxas de erro do *k-fold CV* e do conjunto de validação são definidas analogamente.

5.2 *BOOTSTRAP*

O *bootstrap* é uma ferramenta extremamente poderosa e amplamente aplicável que pode ser usada para quantificar a incerteza associada a um estimador ou método de aprendizado estatístico. Como um simples exemplo. o *bootstrap* pode ser usado para estimar o erro padrão dos coeficientes de uma regressão linear.

A abordagem *bootstrap* permite que usemos o computador para emular o processo de obtenção de novos conjuntos amostrais de modo que possamos estimar a variabilidade de estimador qualquer $\hat{\alpha}$ sem adquirir amostrar adicionais. Ou seja, em vez de obter distintos *data sets* da população repetidas vezes, obtemos conjuntos de dados ao amostrar repetidamente o *data set* original.

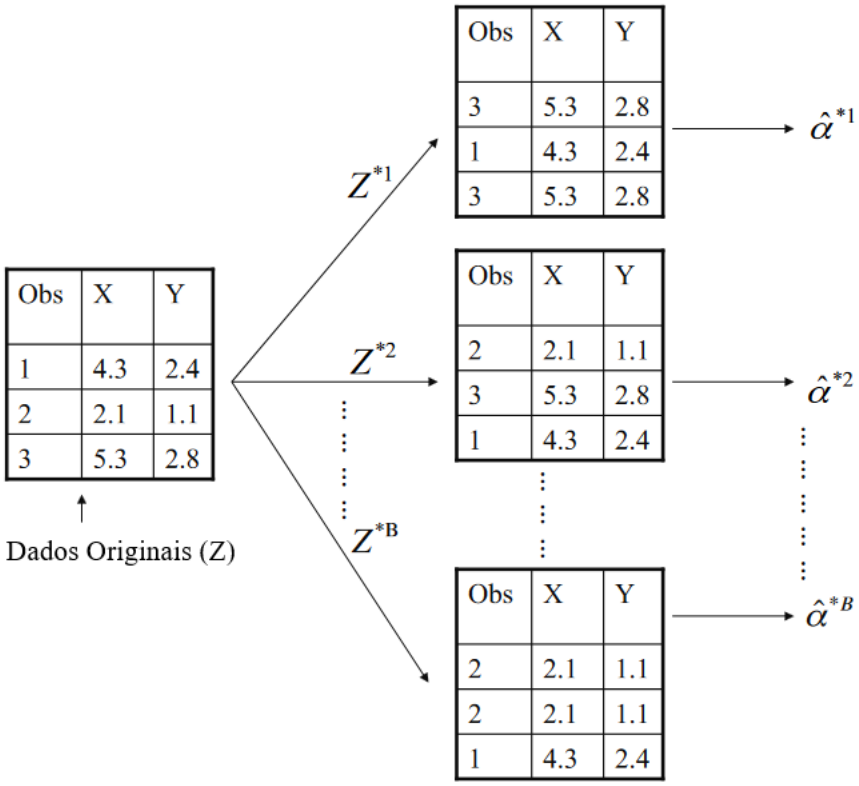

Figura 6: Uma ilustração gráfica da abordagem *bootstrap* em uma pequena amostra contendo $n = 3$ observações. Cada conjunto *bootstrap* contém n observações, selecionadas por amostragem com reposição do *data set* original. Cada conjunto bootstrap é usado para obter uma estimativa de α

Suponha um conjunto de dados com n observações. Randomicamente, selecionamos n observações do *data set* a fim de produzir o conjunto *bootstrap*, Z^{*1}. A amostragem é feita com reposição, o que significa que a mesma observação pode ocorrer mais de uma vez no *boostrap data set*. Note que se uma observação está contida em Z^{*1}, então tanto X quanto Y são incluídos.

Podemos, assim, usar Z^{*1} para produzir uma nova estimativa *bootstrap* para α, chamada $\hat{\alpha}^{*1}$. Esse procedimento é repetido B vezes para um valor grande de B, a fim de produzir B diferentes *boostrap data sets*, $Z^{*1}, Z^{*2}, \ldots, Z^{*B}$ e B estimadores α correspondentes, $\hat{\alpha}^{*1}, \hat{\alpha}^{*2}, \ldots, \hat{\alpha}^{*B}$. Por fim, estimamos o erro padrão desses estimadores *boostrap* com a fórmula

$$SE_B(\hat{\alpha}) = \sqrt{\frac{1}{B-1}\sum_{r=1}^{B}\left(\hat{\alpha}^{*r} - \frac{1}{B}\sum_{r'=1}^{B}\hat{\alpha}^{*r'}\right)^2}. \tag{30}$$

Isso serve como uma estimativa do erro padrão de $\hat{\alpha}$ do conjunto de dados original.

A Figura (6) representa o processo de *bootstrap*.

6 SELEÇÃO DE MODELO LINEAR E REGULARIZAÇÃO

No cenário de regressão, o modelo linear padrão

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \epsilon \tag{31}$$

é comumente usado para descrever o relacionamento entre uma resposta Y e um conjunto de variáveis $X_1, X_2, \ldots, X_p$. Vimos no capítulo 3 que geralmente ajustamos o modelo usando mínimos quadrados.

Nos capítulos seguintes (7, 8 e 10), algumas abordagens não-lineares são apresentadas como forma de expandir o *framework* de modelos lineares. Mas, antes de seguir com o mundo não-linear, discutimos neste capítulo (6) maneiras de melhorar o simples modelo linear, uma vez que - em termos de inferência - o modelo linear se sobressai e - em termos de predição - se mostra bem competitivo quando comparado com outros modelos.

Por que procurar outro procedimento de ajuste em vez do mínimos quadrados?

- Acurácia de Predição: Ao restringir ou encolher os coeficientes estimados, muitas vezes consegue-se reduzir substancialmente a variância em troca de um aumento insignificante do viés.

- Interpretabilidade do Modelo: Muitas vezes algumas ou muitas das variáveis usados numa regressão não estão de fato associadas com a resposta. Ao remover esses atributos, isto é, colocando os respectivos coeficientes em zero, obtém-se um modelo

mais facilmente interpretável.

Quais são as alternativas para o procedimento de ajuste então? Nesse capítulo, discutimos as três mais importantes classes de métodos:

- *Subset Selection*: essa abordagem envolve identificar um subconjunto de p preditores associados à resposta.
- *Shrinkage*: essa abordagem envolve ajustar um modelo com todos os p preditores. Entretanto, os coeficientes estimados são encolhidos, tendendo a zero. O *shrinkage* também é conhecido como Regularização.
- *Redução de Dimensão*: essa abordagem envolve projetar os p preditores em um subespaço dimensional M, onde $M < p$.

6.1 *SUBSET SELECTION*

Nesta seção consideramos alguns métodos para selecionar subconjuntos de preditores. Isso inclui procedimentos de *best subset*[29] e *stepwise model selection*[30].

6.1.1 *Best Subset Selection*

Para performar o *best subset selection*, ajustamos uma regressão linear com mínimos quadrados para cada possível combinação dos p preditores. Então, comparamos os modelos resultantes, identificando qual é o melhor.

O problema de selecionar o melhor subconjunto entre as 2^p possibilidades não é trivial. Esse procedimento é normalmente dividido em duas partes, como descrito no algoritmo 6.1.1 abaixo:

[29]Em português, seleção do melhor subconjunto.
[30]Em português, seleção do modelo passo a passo.

Algoritmo 6.1.1 *Best Subset Selection*

1. Deixamos $\mathcal{M}_0$ denotar o modelo nulo, que contém 0 preditores. Esse modelo simplesmente prediz a média da amostra para cada observação.

2. Para $k = 1, 2, \ldots, p$:
   (a) Ajuste todos os $\binom{p}{k}$ modelos que contém exatamente k preditores.
   (b) Escolha o melhor entre esses $\binom{p}{k}$ modelos e chame-o de $\mathcal{M}_k$. Entenda melhor como tendo o menor RSS ou - equivalentemente - o maior R^2.

3. Escolha um único melhor modelo entre $\mathcal{M}_0, \ldots, \mathcal{M}_p$ usando o erro de predição no conjunto de validação, C_p (AIC), BIC, o R^2 ou o método de validação cruzada.

Ainda que *best subset selection* seja uma simples e conceitualmente atrante abordagem, ela sofre de limitações computacionais. Em geral, há 2^p modelos que envolvem os subconjuntos de p preditores. Então, se $p = 10$, há aproximadamente 1.000 modelos possíveis para serem considerados e - se $p = 20$ - há mais de 1 milhão de possiblidades! Consequentemente, essa abordagem torna-se computacionalmente impraticável para valores de p maiores que aproximadamente 40, mesmo com computadores modernos.

6.1.2 *Stepwise Selection*

Por razões computacionais e pelo fato de - quanto maior o espaço de procura de modelo - maior a chance de encontrar um modelo que performe bem para aquele *dataset* mas não represente o problema real, discutiremos o *stepwise selection*.

6.1.2.1 *Forward Stepwise Selection*

Enquanto o melhor procedimento de seleção de subconjunto considera todos os 2^p modelos possíveis contendo subconjuntos dos p preditores, o *forward stepwise* considera um conjunto muito menor de modelos.

Forward stepwise começa com um modelo que não contém preditores e, em seguida, adiciona preditores ao modelo - um de cada vez - até que todos os preditores estejam no modelo. Em particular, em cada passo, a variável que proporciona a maior melhoria adicional ao ajuste é adicionada ao modelo. Mais formalmente, o procedimento de seleção forward stepwise é dado no Algoritmo 6.1.2.1:

Algoritmo 6.1.2.1 *Forward Stepwise Selection*

1. Deixamos $\mathcal{M}_0$ denotar o modelo nulo, que contém 0 preditores.

2. Para $k = 1, 2, \ldots, p-1$:
   (a) Considere todos os $p-k$ modelos que aumentam os preditores em $\mathcal{M}_k$ em um preditor.
   (b) Escolha o melhor entre esses $p-k$ modelos e chame-o de $\mathcal{M}_{k+1}$. Entenda melhor como tendo o menor RSS ou o maior R^2.

3. Escolha um único melhor modelo entre $\mathcal{M}_0, \ldots, \mathcal{M}_p$ usando o erro de predição no conjunto de validação, C_p (AIC), BIC, o R^2 ou o método de validação cruzada.

A vantagem computacional dessa abordagem é clara. Porém, embora ela tenda a performar bem na prática, não é garantido que esse método encontre o melhor modelo possível entre os 2^p subconjuntos.

Suponha, por exemplo, que temos um *dataset* com 3 atributos X_1, X_2, X_3 e queremos achar o melhor modelo com 2 variáveis. Ainda, utilizando o método de *best subset selection*, achamos que esse modelo é composto pelos atributos X_2 e X_3. Se - por algum motivo - o modelo de 1 variável com maior desempenho seja o X_1, a abordagem *forward stepwise* não vai conseguir encontrar esse melhor modelo de 2 atributos. Esse exemplo está ilustrado na tabela 6.1.2.1.

# Variáveis	*Best subset*	*Forward stepwise*
1	X_1	X_1
2	X_2, X_3	X_1, X_2

Essa abordagem pode ser aplicada mesmo em cenários de muitas dimensões onde $n < p$. Nesse caso, entretanto, só é possível construir modelos $\mathcal{M}_0, \ldots, \mathcal{M}_{n-1}$, uma vez que o modelo não iria resultar em soluções únicas se $p \geq n$.

6.1.2.2 Backward Stepwise Selection

Ao contrário da seleção *forward stepwise*, ela começa com o modelo de mínimos quadrados completo contendo todos os preditores p e, em seguida, remove iterativamente o preditor menos útil, um de cada vez. Detalhes são fornecidos no Algoritmo 6.1.2.2.

Algoritmo 6.1.2.2 *Backward Stepwise Selection*

1. Deixamos $\mathcal{M}_p$ denotar o modelo completo, que contém todos os p preditores.

2. Para $k = p, p-1, \ldots, 1$:
   (a) Considere todos os k modelos que contêm todos menos um preditores em $\mathcal{M}_k$, para um total de $k-1$ preditores.
   (b) Escolha o melhor entre esses k modelos e chame-o de $\mathcal{M}_{k-1}$. Entenda melhor como tendo o menor RSS ou o maior R^2.

3. Escolha um único melhor modelo entre $\mathcal{M}_0, \ldots, \mathcal{M}_p$ usando o erro de predição no conjunto de validação, C_p (AIC), BIC, o R^2 ou o método de validação cruzada.

Assim como o *forward stepwise selection*, essa abordagem não garante que encontraremos o melhor modelo que contém um subconjunto dos p preditores.

Backward selection requer que o número de observações n seja maior que o número de variáveis p, diferentemente do *forward stepwise selection*.

6.1.3 Escolhendo o modelo ótimo

Para selecionar melhor modelo em relação ao erro de teste, precisamos estimar esse erro. Há duas maneira mais comuns:

1. Podemos indiretamente estimar o erro de teste ao ajustar o erro de treinamento levando em consideração o viés.

2. Podemos diretamente estimar o erro de teste usando a abordagem de conjunto de validação ou a abordagem de validação cruzada.

6.2 MÉTODOS *SHRINKAGE*

Os método descritos na seção 6.1 envolvem usar mínimos quadrados para ajustar o modelo linear que contém um subconjunto dos preditores. Como alternativa para isso, pode-se ajustar um modelo que contém todos os p preditores usando uma técnica que restringe ou regulariza os coeficientes estimados, ou equivalentemente, que encolhe[31] os coeficientes a zero.

As duas técnicas mais conhecidas para o encolhimento dos coeficientes são:

[31] A palavra usada em inglês é *shrink*, que significa encolher.

1. *Ridge Regression*;

2. *Lasso*.

6.2.1 *Ridge Regression*

O procedimento de mínimos quadrados estima $\beta_0, \beta_1, \ldots, \beta_p$ usando valores que minimizam

$$RSS = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 .$$

A regressão Ridge é bem similar, com exceção de que os coeficientes são estimados ao minimizar uma quantia ligeiramente diferente:

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = RSS + \lambda \sum_{j=1}^{p} \beta_j^2, \tag{32}$$

onde $\lambda \geq 0$ é um parâmetro de afinação, determinado separadamente.

Assim como mínimos quadrados, *ridge regression* busca coeficientes que façam um RSS pequeno (primeiro termo da equação 32). Entretanto, o segundo termo, $\lambda \sum_{j=1}^{p} \beta_j^2$ - chamado de penalidade *shrinkage* - é pequeno quando $\beta_1, \ldots, \beta_p$ são próximos de zero e, por consequência, tem o efeito de encolher as estimativas de β_j para zero.

O parâmetro de afinação λ serve para controlar o impacto relativo dos dois termos na estimação dos coeficientes de regressão. Quando $\lambda = 0$, o termo de penalidade não tem efeito, ou seja, é igual os coeficientes de mínimos quadrados. Contudo, conforme $\lambda \to \infty$, o impacto da penalidade *shrinkage* aumenta, e os coeficientes da *ridge regression* aproximam-se de zero.

Note que, em (32), a penalidade não afetou o intercepto β_0, e sim os coeficientes $\beta_1, \beta_2, \ldots, \beta_p$

6.2.1.1 Por que *Ridge Regression* em vez de Mínimos Quadrados?

Em geral, em situações onde o relacionamento entre resposta e os preditores é aproximadamente linear, as estimativas com mínimos quadrados vão ter baixo viés mas podem ter alta variância.

A vantagem da regressão ridge está enraizada no *bias-variance trade-off*. Conforme λ cresce, a flexibilidade do ajuste regressão ridge diminui, diminuindo a variância mas

aumentando o viés. Portanto, *ridge regression* funciona melhor em situações onde mínimos quadrados produzem estimativas com alta variância.

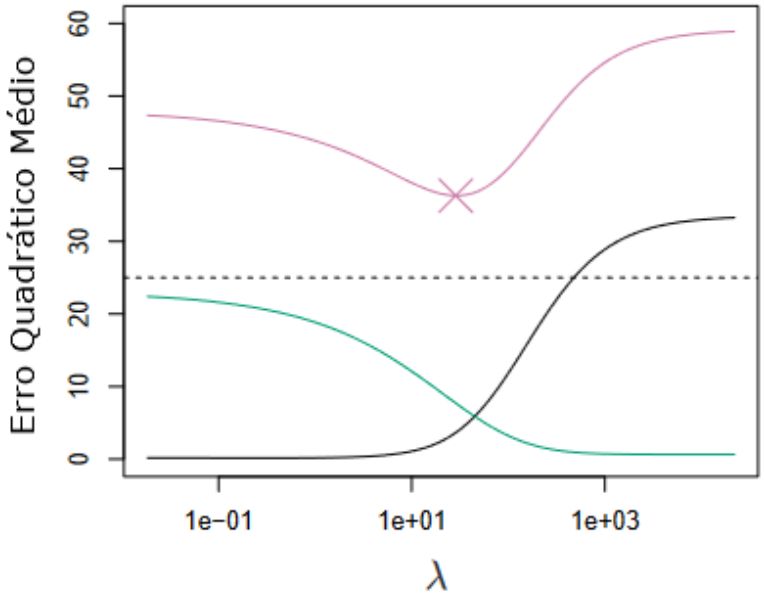

Figura 7: Viés ao quadrado (preto), variância (verde) e erro quadrático médio de teste (lilás) para *ridge regression* em um conjunto de dados simulados, em função de λ. As linhas horizontais tracejadas indicam o mínimo possível MSE. A cruz lilás indica o modelo de *ridge regression* em que o MSE é o menor possível.

6.2.2 Lasso

A regressão ridge tem uma desvantagem óbvia: diferentemente da abordagem de *subset selection*, *ridge regression* vai incluir todos os p preditores no modelo final. A penalidade *shrinkage* vai encolher todos os coeficientes em direção a zero, mas nenhum deles vai ser exatamente zero (a não ser que $\lambda = \infty$).

O lasso é uma alternativa relativamente recente que supera essa desvantagem. Os coeficientes lasso, β_λ^L, minimizam a quantia:

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| = RSS + \lambda\sum_{j=1}^{p}|\beta_j|, \tag{33}$$

Comparando (32) e (33), pode-se ver que lasso e ridge tem formulações parecidas. A única diferença é que o termo β_j^2 da regressão ridge (32) foi trocado por $|\beta_j|$ na penalidade lasso (33).

Assim como na regressão ridge, o lasso encolhe os coeficientes para zero. Entretanto, a penalidade lasso tem o efeito de forçar alguns coeficientes para serem exatamente zero

quando o parâmetro de afinação é suficientemente grande. Assim, o lasso performa uma seleção variáveis e, consequentemente, é muito mais fácil interpretar os modelos gerados com lasso do que os gerados por *ridge regression*.

6.2.2.1 Comparando Lasso e *Ridge Regression*

É claro que o lasso tem uma vantagem em relação à regressão ridge: interpretabilidade. Em contrapartida, qual dos dois produz uma melhor acurácia de predição? Depende, nem sempre um vai dominar o outro.

Em geral, o lasso tende a performar melhor em um cenário onde somente um relativamente pequeno número de preditores tem coeficientes substancial. Já a regressão ridge vai performar melhor quando a resposta é uma função de vários preditores, todos com coeficientes de tamanho parecido.

Técnicas tal qual a validação cruzada podem ser usadas para determinar qual abordagem é melhor num determinado *data set*.

6.2.3 Selecionando o Parâmetro de Afinação

Assim como as abordagens *subset*, a implementação de *ridge regression* e do lasso requerem um método para selecionar um valor para o parâmetro de afinação λ em (32) e (33). *Cross-validation* é uma simples maneira de resolver esse problema.

Escolhemos uma gama de valores para λ e calculamos o erro de validação cruzada para cada valor de λ. Selecionamos, assim, o valor de parâmetro de afinação cujo o erro do *cross-validation* é o menor.

6.3 MÉTODOS DE REDUÇÃO DE DIMENSIONALIDADE

Os métodos que vimos até o momento nesse capítulo são definidos usando os preditores originais, $X_1, X_2, \ldots, X_p$. Vamos explorar agora uma classe de abordagens que transformam os preditores e, em seguida, ajustam o modelo de mínimos quadrados usando essas variáveis transformadas. Essas técnicas chamam-se métodos de redução de dimensionalidade.

Assuma que $Z_1, Z_2, \ldots, Z_M$ representam $M < p$ combinações lineares dos nossos p preditores originais. Isso é:

$$Z_m = \sum_{j=1}^{p} \phi_{jm} X_j \tag{34}$$

para algumas constantes $\phi_{1m}, \phi_{2m}, \ldots, \phi_{pm}$, $m = 1, \ldots, M$. Podemos, assim, ajustar o modelo de regressão linear

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i, \quad i = 1, \ldots, n. \tag{35}$$

usando mínimos quadrados. Note que em (35), os coeficientes de regressão são dados por $\theta_0, \theta_1, \ldots, \theta_M$.

O termo redução de dimensionalidade vem do fato que essa abordagem reduz o problema de estimar os $p + 1$ coeficientes $\beta_0, \beta_1, \ldots, \beta_p$ para o simples problema de estimar os $M + 1$ coeficientes $\theta_0, \theta_1, \ldots, \theta_M$, onde $M < p$. Isto é, a dimensão do problema foi reduzida de $p + 1$ para $M + 1$.

Todos os métodos de redução de dimensionalidade funcionam em dois passos. Primeiro, encontrar os preditores transformados $Z_1, Z_2, \ldots, Z_M$. Segundo, o modelo é ajustado usando esses M preditores.

Nesse capítulo, veremos duas abordagens para essa tarefa:

1. *Principal Components*;
2. *Partial Least Squares*.

6.3.1 *Principal Components Regression*

Principal components analysis[32] (PCA) é uma abordagem popular por derivar um conjunto de baixa dimensionalidade de um conjunto de alta dimensionalidade.

6.3.1.1 Uma visão geral da Análise de Componentes Principais

PCA é uma técnica de reduzir a dimensão de uma matriz X com dados em $n \times p$. A direção da primeira componente principal Z_1 é aquela na qual as observações variam mais. Outra interpretação seria: o vetor da primeira componente principal define a linha que é a mais próxima possível dos dados.

Em geral, podemos construir até p distintas componentes principais. A segunda componente principal Z_2 é uma combinação linear das variáveis que não são correlacionadas com Z_1.

[32] Em português, análise de componentes principais.

6.3.1.2 A abordagem de Regressão de Componentes Principais

A abordagem de regressão de componentes principais (PCR) envolve construir os primeiros M componentes principais $Z_1, \ldots, Z_M$, e - em seguida - usar esses componentes como preditores em um modelo de regressão linear com mínimos quadrados. A ideia chave é que usualmente um pequeno número de componentes principais é suficiente para explicar tanto a maior parte da variabilidade dos dados quanto o relacionamento com a resposta.

Se a suposição supracitada for válida, então ajustar um modelo de mínimos quadrados a $Z_1, \ldots, Z_M$ levará a melhores resultados do que ajustar um modelo de mínimos quadrados a $X_1, \ldots, X_p$, uma vez que a maior parte ou todas as informações dos dados relacionadas à resposta estão contidas em $Z_1, \ldots, Z_M$, e estimando apenas os coeficientes $M \ll p$[33] podemos mitigar o *overfitting*

Observe que - embora o PCR forneça uma maneira simples de realizar regressão usando $M < p$ preditores - ele não é um método de seleção de variáveis. Isso ocorre porque cada um dos M componentes principais usados na regressão é uma combinação linear de todos os p atributos originais.

No PCR, o número de componentes principais, M, é normalmente escolhido por validação cruzada.

Quando performando o PCR, geralmente recomendamos a padronização/normalização de cada preditor antes de gerar os componentes principais. A normalização garante que todas as variáveis estejam na mesma escala. Na ausência da padronização, as variáveis que tem alta variância afetam mais as componentes principais e, por conseguinte, afetam o resultado do modelo. Se todas as variáveis tiverem a mesma escala, contudo, pode-se escolher não padronizá-las.

6.3.2 *Partial Least Squares*

A abordagem PCR tem uma desvantagem: não há garantia que as direções que melhor explicam os preditores vão ser também as melhores direções para predizer a resposta. Isso ocorre porque a abordagem PCR é não supervisionada em um primeiro momento (na construção das componentes principais).

Apresentamos, assim, o *partial least squares* (PLS), uma alternativa supervisionada para o PCR.

Assim como o PCR, o PLS primeiro identifica um novo conjunto de *features* $Z_1, \ldots, Z_M$ que são uma combinação linear dos atributos originais, e - em seguida - ajusta um modelo

[33] O símbolo $\ll$ representa que M é muito menor que p.

linear por mínimos quadrados nesse novo conjunto.

Diferentemente do PCR, entretanto, PLS identifica esses novos atributos de uma forma supervisionada, fazendo uso da resposta Y a fim identificar novas variáveis que - além de aproximar bem os antigos atributos - são relacionados com a resposta.

Descrição de como a primeira direção PLS é calculada: depois de padronizar os p preditores, PLS calcula a primeira direção Z_1 ao colocar ϕ_{j1} em (34) igual ao coeficiente de uma regressão linear simples de Y em X_j. Portanto, ao calcular $Z_1 = \sum_{j=1}^{p} \phi_{j1} X_j$, PLS coloca o maior peso nas variáveis que são mais fortemente relacionadas com a resposta.

Para encontrar a segunda direção do PLS, ajustamos cada variável para Z_1, ao regressar cada variável em Z_1 e tomar os resíduos. Esses resíduos podem ser interpretados como a informação remanescente que não foi explicada pela primeira direção PLS. Calculamos então Z_2 utilizando esses dados ortogonalizados da mesma forma que Z_1 foi calculado baseado nos dados originais. Essa abordagem pode ser reutilizada M vezes para achar as múltiplas componentes PLS $Z_1, Z_2, \ldots, Z_M$.

Assim como no PCR, o número M de direções usadas no PLS é um parâmetro de afinação tipicamente escolhido com validação cruzada.

6.4 CONSIDERAÇÕES EM ALTAS DIMENSÕES

6.4.1 Dados *High-Dimensional*

A maioria das técnicas estatísticas para regressão e classificação são para o cenário de baixa dimensionalidade[34], no qual n, o número de observações, é muito maior que p, o número de atributos. Isso decorre muito do fato de que - ao longo da história - os problemas que requeriam o uso de estatística eram de baixa dimensões.

Nos últimos 20 anos, novas tecnologias têm mudado a forma com os dados são coletados. Agora, é comum coletar um número quase ilimitado de variáveis (p muito grande). Enquanto p pode ser muito grande, o número de observações n é comumente limitado devido ao custo, à disponibilidade amostral e a outras coisas.

Data sets que contém mais atributos que observações são usualmente chamados de dados *high-dimensional* ou, em português, dados de altas dimensões. As abordagens clássicas, a exemplo do modelo de mínimos quadrados, não são apropriadas para esse cenário.

Definimos o cenário de alta dimensão como o caso em que o número de características p é maior que o número de observações n. Mas as considerações que discutiremos agora

[34]Quando se fala de dimensão, trata-se do número de variáveis p.

certamente também se aplicam se p for ligeiramente menor que n, e é melhor mantê-las sempre em mente ao realizar o aprendizado supervisionado.

6.4.2 O que dá errado em Altas Dimensões?

Para ilustrar a necessidade de cuidado extra e de técnicas especializadas para regressão e classificação quando $p > n$, começamos examinando o que pode dar errado se aplicarmos uma técnica estatística não destinada ao cenário de alta dimensão. Para este propósito, examinamos a regressão de mínimos quadrados, mas os mesmos conceitos se aplicam à regressão logística, à análise discriminante linear e a outras abordagens estatísticas clássicas.

Quando o número de características p é tão grande ou maior que o número de observações n, os mínimos quadrados descritos no Capítulo 3 não podem (ou melhor, não deveriam) ser realizados. A razão é simples: independentemente de existir ou não realmente uma relação entre as características e a resposta, os mínimos quadrados produzirão um conjunto de estimativas de coeficientes que resultam num ajuste perfeito aos dados, de modo que os resíduos sejam zero.

6.4.3 Regressão em Altas Dimensões

Acontece que muitos dos métodos vistos nesse capítulo, como *forward stepwise selection*, *ridge regression*, lasso, PCR e PLS são particularmente úteis para performar regressão em altas dimensões. Essencialmente, essas abordagens evitam o *overfitting* ao usar um método menos flexível do que mínimos quadrados.

Três pontos importantes em regressão em *high dimensions*:

1. Regularização ou *shrinkage* tem um papel chave em problemas de alta dimensionalidade;

2. A escolha do parâmetro de afinação é essencial para uma boa performance preditiva;

3. O erro de teste tende a aumentar conforme a dimensionalidade do problema aumenta.

O terceiro ponto é conhecido como a *maldição da dimensionalidade* ou, em inglês, *curse of dimensionality*. Em geral, adicionar atributos que são verdadeiramente relacionados com a resposta vão melhorar o modelo, diminuindo o erro de teste. Entretanto, adicionar variáveis que não tem relação com a resposta vai causar um deterioramento no modelo ajustado e, consequentemente, um aumento no erro de teste.

Assim, vemos que novas tecnologias que permitem a coleta de medições para milhares

ou milhões de características são uma faca de dois gumes: podem levar a modelos preditivos melhorados se essas variáveis forem de fato relevantes para o problema em questão, mas levarão a resultados piores se os atributos não forem relevantes. Mesmo que sejam relevantes, a variância incorrida no ajuste dos seus coeficientes pode pesar mais que a redução do enviesamento que trazem.

6.4.4 Interpretando Resultados em Altas Dimensões

Quando performamos o lasso, a regressão ridge ou outros procedimentos de regressão em um cenário de alta dimensionalidade, devemos ter cuidado na forma com que vamos relatar os resultados obtidos. No capítulo 3, aprendemos sobre a multicolinearidade, que se refere às variáveis numa regressão que podem ser correlacionadas entre si. No cenário de alta dimensionalidade, o problema da multicolinearidade é extremo: qualquer variável pode ser escrita como uma combinação linear de todas as outras variáveis do modelo.

Essencialmente, isso significa dizer que nunca podemos saber exatamente quais variáveis (caso existam) são verdadeiramente preditivas da resposta e que nunca podemos identificar quais os melhores coeficientes para usar numa regressão. No máximo, podemos esperar atribuir grandes coeficientes de regressão a variáveis que estejam correlacionadas com as variáveis que realmente são preditivas do resultado.

É importante, ademais, ser particularmente atento ao relatar erros e medidas de um ajuste de modelo em um cenário de alta dimensionalidade. Vimos que quando $p > n$, é fácil obter um modelo inútil que tem zero resíduo. Portanto, nunca devemos usar MSE, p-valor, R^2 ou outros métodos tradicionais nos dados de treinamento como evidência de um bom modelo num cenário desse.

Em contrapartida, devemos relatar os resultados com base num *data set* de teste independente ou nos erros de *cross-validation*. Por exemplo, o MSE ou R^2 de um conjunto de dados de teste independente são medidas válidas, porém o MSE ou R^2 de dados de treinamento certamente não são.

7 INDO ALÉM DA LINEARIDADE

7.1 REGRESSÃO POLINOMIAL

o foco deste capítulo é a extensão dos modelos de regressão linear para capturar relações não lineares entre variáveis, as quais serão detalhadas nos próximos tópicos

7.1.1 Conceitos Básicos

A regressão polinomial é uma forma de regressão onde o modelo inclui termos polinomiais das variáveis preditoras. Isso permite que o modelo ajuste relações não lineares entre a variável dependente e uma ou mais variáveis independentes. O modelo é descrito pela equação:

$$Y = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \ldots + \beta_d x_i^d + \epsilon_i \tag{36}$$

Aqui, Y é a variável dependente, x é a variável independente, $\beta_0, \beta_1, ..., \beta_d$ são os coeficientes do modelo, d é o grau do polinômio, e ϵ é o termo de erro.

7.1.2 Estimação dos Coeficientes

- Método de Estimação: Os coeficientes são geralmente estimados usando o método dos mínimos quadrados. Esse método busca minimizar a soma dos quadrados dos resíduos (diferenças entre os valores observados e os valores previstos pelo modelo).

- Interpretação dos Coeficientes: Cada coeficiente no modelo polinomial pode ser interpretado como o efeito da variável independente no nível correspondente do termo polinomial, ajustando-se pelos outros termos no modelo.

7.1.3 Considerações Práticas

- Escolha do Grau do Polinômio: A escolha do grau do polinômio é crucial. Um grau muito baixo pode não capturar toda a complexidade dos dados (subajuste), enquanto um grau muito alto pode levar a um modelo que se ajusta demais aos dados de treinamento (sobreajuste).

- Diagnóstico de Ajuste: Gráficos de resíduos e outras métricas de diagnóstico são usados para avaliar a adequação do modelo aos dados. Isso ajuda a identificar problemas como heterocedasticidade ou a presença de outliers.

- Validação Cruzada: Métodos de validação cruzada podem ser utilizados para ajudar na seleção do grau do polinômio. Esses métodos permitem avaliar o desempenho do modelo em conjuntos de dados que não foram usados durante o treinamento.

7.1.4 Aplicações

- Aplicações Práticas: A regressão polinomial é frequentemente aplicada em contextos onde as relações entre variáveis são conhecidas por serem não lineares, como em estudos de crescimento populacional, decaimento de processos químicos, ou comportamentos de

mercado financeiro.

7.2 FUNÇÕES DEGRAU

As funções degrau são uma técnica em modelos estatísticos usada para evitar impor uma estrutura global quando se trabalha com variáveis contínuas. Ao dividir a variável em diferentes intervalos, ou "bins", cada intervalo é associado a um valor constante distinto, transformando a variável contínua em uma variável categórica ordenada.

Isso é feito por meio de pontos de corte específicos. Essa abordagem é útil para modelar efeitos não lineares de maneira simples e é amplamente usada em áreas como biostatística e epidemiologia, apesar de algumas limitações, como a possível perda de tendências contínuas nos dados.

Funções degrau transformam uma variável contínua X em uma variável categórica ordenada dividindo seu alcance em intervalos fixos, ou "bins". Cada bin é associado a uma constante, utilizando os pontos de corte $c_1, c_2, ..., c_K$. Isso é representado matematicamente como:

$$C_j(X) = I(c_j \leq X < c_{j+1}) \tag{37}$$

onde $I(\cdot)$ é a função indicadora. Em um modelo linear, as variáveis resultantes $C_j(X)$ são usadas como preditores, e o modelo é dado por:

$$y_i = \beta_0 + \beta_1 C_1(x_i) + \beta_2 C_2(x_i) + \cdots + \beta_K C_K(x_i) + \epsilon_i \tag{38}$$

Este método é particularmente útil quando os dados apresentam comportamentos distintos em diferentes intervalos de X.

7.3 FUNÇÕES DE BASE

Modelos de regressão utilizando funções base abordam relações complexas de uma maneira estruturada, transformando a variável X em um conjunto de funções $b_1(X), b_2(X), \ldots, b_K(X)$. Funções base são transformações específicas aplicadas aos dados para capturar padrões e tendências, possibilitando o uso de modelos lineares sobre essas transformações. Além dos polinômios e funções peça-constante, métodos como splines e séries de Fourier também são comuns. Essa abordagem amplia a flexibilidade do modelo, permitindo análises robustas com ferramentas de inferência de modelos lineares. A aplicação de funções base em modelos de regressão possibilita uma modelagem mais flexível e profunda dos dados. Estas são algumas das transformações comuns e suas respectivas representações matemáticas:

7.4 SPLINES DE REGRESSÃO

Regressão spline é um método avançado que ajusta um modelo polinomial por partes ao longo do intervalo de uma variável X, dividido por pontos chamados de "nós"ou "knots". Esses nós são essenciais porque permitem que o modelo mude sua forma em diferentes segmentos do domínio de X, proporcionando uma grande flexibilidade para modelar relações não-lineares complexas entre variáveis.

Na prática, a escolha da localização e do número de nós é crucial, pois influencia diretamente a capacidade do modelo de capturar a variação dos dados sem sobreajustar. Métodos comuns para definir esses nós incluem a colocação uniforme ou com base nos quantis dos dados.

As splines podem ser de vários tipos, como B-splines ou splines cúbicas, que são populares devido à sua suavidade. As B-splines, por exemplo, oferecem uma forma eficiente de controlar a suavidade da função ajustada, permitindo ajustes finos através de penalidades na variação das derivadas da função spline.

Além disso, os modelos de spline são geralmente ajustados usando métodos de mínimos quadrados ou mínimos quadrados penalizados, proporcionando uma forma robusta de estimar os parâmetros do modelo enquanto controla o sobreajuste, o que é crucial para garantir que o modelo generalize bem para novos dados.

7.4.1 Polinômios por Partes

A regressão polinomial por partes é uma abordagem estatística que divide o intervalo de uma variável independente X em diferentes regiões, ajustando polinômios separados em cada região. Esta técnica é útil quando diferentes partes do domínio de X exibem diferentes comportamentos que um único modelo polinomial global não conseguiria capturar eficientemente. Os pontos onde a mudança de modelo ocorre são chamados de "nós". Aqui estão os principais pontos sobre esta técnica:

1. Flexibilidade: Permite a adaptação do modelo às variações locais dos dados, aumentando a flexibilidade do modelo em comparação com um polinômio global de alto grau.

2. Nós: Os pontos de mudança nos coeficientes do modelo são definidos como nós. O número e a posição dos nós podem afetar significativamente o ajuste do modelo.

3. Descontinuidades: Um desafio comum com este modelo é a possibilidade de descontinuidades nos nós, o que pode tornar a função ajustada visualmente desagradável ou analiticamente problemática.

4. Grau do Polinômio: O grau dos polinômios pode variar, não sendo necessário utilizar sempre polinômios de alto grau, o que confere certa versatilidade ao modelo.

Suponhamos que queremos modelar a relação entre anos de experiência X e salário Y para um conjunto de dados. Observamos que o aumento salarial não é constante e parece acelerar após 5 anos de experiência e novamente após 10 anos. Usamos uma regressão spline cúbica com nós em 5 e 10 anos para modelar essa relação.

Definição do Modelo

O modelo seria definido por três polinômios cúbicos:

$$Y_i = \begin{cases} \beta_{01} + \beta_{11}X_i + \beta_{21}X_i^2 + \beta_{31}X_i^3 + \epsilon_i, & \text{para } X_i < 5 \\ \beta_{02} + \beta_{12}(X_i - 5) + \beta_{22}(X_i - 5)^2 + \beta_{32}(X_i - 5)^3 + \epsilon_i, & \text{para } 5 \leq X_i < 10 \\ \beta_{03} + \beta_{13}(X_i - 10) + \beta_{23}(X_i - 10)^2 + \beta_{33}(X_i - 10)^3 + \epsilon_i, & \text{para } X_i \geq 10 \end{cases} \tag{39}$$

Ajuste do Modelo

Para garantir a continuidade e a suavidade nas junções (nós em 5 e 10 anos), impomos condições de continuidade nos nós, assegurando que o valor e as derivadas até a segunda ordem dos polinômios sejam iguais nos nós.

Inferência Estatística

Após ajustar os coeficientes usando mínimos quadrados, podemos realizar testes de hipóteses para avaliar a significância dos coeficientes, examinar o ajuste do modelo e realizar previsões para novos dados dentro do domínio de X.

7.4.2 Restrições e Splines

7.4.3 Representação da Base de Splines

Ao modelar tendências não-lineares em dados, as splines cúbicas são ferramentas poderosas por sua flexibilidade e suavidade. Elas são definidas matematicamente como uma combinação de polinômios cúbicos básicos e funções de potência truncada que são ajustadas para cada nó. Este modelo pode ser expresso pela equação:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \sum_{j=1}^{K} \beta_{3+j} h(x_i, c_j) + \epsilon_i \tag{40}$$

onde $h(x, c) = (x - c)_+^3$ é a função de potência truncada que contribui para a forma da spline acima do nó c, sendo $(x - c)_+^3 = (x - c)^3$ se $x > c$, e 0 caso contrário.

A continuidade da spline e suas primeiras duas derivadas nos nós é garantida, permitindo que apenas a terceira derivada apresente descontinuidade. Essa característica faz com que a spline seja visualmente suave e contínua em toda sua extensão, exceto na terceira derivada.

Para estabilizar o modelo nas extremidades, as splines naturais são empregadas, adicionando restrições de linearidade nos pontos fora dos nós extremos. Essa abordagem é formulada como:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \sum_{j=1}^{K} \beta_{3+j} h(x_i, c_j) + \delta \begin{cases} (x_i - c_1)^3, & \text{se } x_i < c_1 \\ (x_i - c_K)^3, & \text{se } x_i > c_K \end{cases} + \epsilon_i \quad (41)$$

A implementação dessas restrições assegura que a função seja linear onde X é menor que o nó mais baixo ou maior que o nó mais alto, proporcionando um ajuste mais estável e reduzindo a variabilidade nas estimativas de borda.

Essas técnicas são essenciais para modelar dados com complexidades não-lineares, permitindo ajustes precisos e interpretações robustas dos resultados modelados.

7.4.4 Escolhendo o Número e Localizações dos Nós

Escolher o número e as localizações dos nós em uma regressão spline é crucial para capturar adequadamente as características dos dados. Os nós aumentam a flexibilidade da spline nas regiões em que são colocados, permitindo que os coeficientes polinomiais se ajustem rapidamente para acomodar mudanças na função subjacente. Assim, uma abordagem é colocar mais nós em áreas onde a função parece variar mais rapidamente e menos nós onde a função é mais estável.

Embora essa estratégia possa ser eficaz, na prática é comum distribuir os nós de maneira uniforme ao longo dos quantis dos dados. Por exemplo, é possível definir um número desejado de graus de liberdade para o modelo e, em seguida, usar o software para posicionar automaticamente o número correspondente de nós nos quantis uniformes dos dados, como os percentis 25º, 50º e 75º.

A determinação do número ideal de nós envolve um equilíbrio entre flexibilidade e complexidade do modelo. Utilizar muitos nós pode levar ao sobreajuste, especialmente em regiões com dados escassos, enquanto poucos nós podem não capturar suficientemente

a complexidade dos dados. Uma técnica objetiva para selecionar o número de nós é o uso de validação cruzada, onde diferentes configurações são testadas e a configuração com o menor erro quadrático médio (RSS) validado cruzadamente é escolhida.

Ao analisar dados de salários utilizando regressão spline, a definição dos nós é um passo crítico que influencia diretamente a flexibilidade e a adequação do modelo. Geralmente, o número de graus de liberdade determina quantos nós ou polinômios serão utilizados no modelo. A escolha de distribuir os nós uniformemente ao longo dos quantis dos dados visa garantir que cada segmento da spline contenha uma quantidade suficiente de dados para um ajuste robusto, particularmente em áreas de alta variabilidade.

Por exemplo, ao definir nós nos percentis 25^{o}, 50^{o} e 75^{o}, buscamos equilibrar a necessidade de flexibilidade do modelo em responder às mudanças nos dados, enquanto mantemos a simplicidade e evitamos o sobreajuste. Cada nó adiciona uma função de potência truncada ao modelo polinomial básico, permitindo ajustes locais na curva de regressão:

$$h(x, c) = (x - c)_{+}^{3} = \max(0, x - c)^{3} \tag{42}$$

Essas funções são integradas ao modelo de polinômio cúbico, aumentando sua capacidade de capturar variações sutis nos dados em torno de cada nó. O modelo final é ajustado usando mínimos quadrados, otimizando os coeficientes para minimizar o erro.

Além do ajuste, a validade do modelo é frequentemente testada por validação cruzada, uma técnica que envolve ajustar o modelo a subconjuntos dos dados e testar sua capacidade de prever novas observações. O modelo com o menor erro quadrático médio, indicado por RSS durante a validação cruzada, é geralmente selecionado.

7.5 SPLINES DE SUAVIZAÇÃO

7.5.1 Visão Geral dos Splines de Suavização

A abordagem das splines de suavização é uma técnica sofisticada para ajustar uma curva suave aos dados, visando equilibrar a precisão do ajuste e a suavidade da curva resultante. Essa metodologia é essencialmente uma extensão dos métodos de regressão regularizados, como a regressão ridge e o lasso, e utiliza um conceito semelhante de "perda mais penalidade".

No caso das splines de suavização, o objetivo é minimizar uma função objetivo que compreende dois termos principais: um termo de perda, que é a soma dos quadrados dos resíduos (RSS) entre os valores observados e os valores ajustados pela função $g(x)$; e um

termo de penalidade, que impõe uma suavidade à função $g(x)$ ao penalizar a soma dos quadrados das segundas derivadas de g ao longo de seu domínio. Matematicamente, isso é representado pela equação:

$$\text{RSS} = \sum_{i=1}^{n}(y_i - g(x_i))^2 + \lambda \int (g''(t))^2 \, dt. \tag{43}$$

Aqui, λ é um parâmetro de ajuste não negativo que controla o equilíbrio entre a aderência aos dados e a suavidade da curva. Quando λ é zero, a spline se ajusta exatamente aos dados, resultando em um modelo altamente flexível que pode sobreajustar. Com um λ muito alto, a função $g(x)$ tende a ser muito suave, aproximando-se de uma linha reta.

As splines de suavização são, em essência, polinômios cúbicos por partes com nós nos valores únicos $x_1, \ldots, x_n$ dos dados, e são contínuas até a segunda derivada em cada nó. Curiosamente, fora do intervalo dos nós extremos, a função $g(x)$ torna-se linear, comportando-se como uma spline natural cúbica. No entanto, a escolha de λ influencia diretamente a extensão do "encolhimento"aplicado à spline, moderando sua flexibilidade para evitar o sobreajuste.

Este modelo é particularmente útil quando se busca uma representação que seja ao mesmo tempo fiel aos dados e que exiba uma variação suave, evitando as oscilações bruscas que modelos mais flexíveis podem apresentar. A escolha adequada do parâmetro λ é crucial, podendo ser guiada por técnicas como validação cruzada para encontrar o equilíbrio ideal entre viés e variância.

7.5.2 Escolha do Parâmetro de Suavização

Na análise de splines de suavização, a escolha do parâmetro λ é fundamental para controlar a suavidade da curva ajustada. Este parâmetro atua como um regulador entre a aderência aos dados e a suavidade da curva, impactando diretamente os graus de liberdade efetivos da spline. Com λ igual a zero, a spline se ajusta exatamente aos pontos de dados, resultando em um modelo extremamente flexível, mas potencialmente sobreajustado. À medida que λ aumenta, a curva se torna progressivamente mais suave, até chegar ao extremo de uma linha reta quando λ é suficientemente grande.

Para determinar o valor ótimo de λ, a validação cruzada, especialmente a validação cruzada de exclusão de um (LOOCV), é uma ferramenta eficaz. O processo de LOOCV para splines de suavização é computacionalmente eficiente, permitindo que se ajuste o modelo várias vezes, excluindo uma observação por vez e calculando o erro quadrático médio residual para cada configuração de λ. O objetivo é encontrar o λ que minimiza

esse erro, equilibrando assim a suavidade e a capacidade de ajuste do modelo.

A fórmula utilizada para calcular o RSS de validação cruzada com LOOCV para splines de suavização considera as previsões ajustadas e a influência de cada ponto de dados no ajuste final, representada pelos elementos diagonais da matriz de suavização S_λ. Este processo não apenas seleciona o λ mais apropriado, mas também proporciona insights sobre como a suavidade influencia a interpretação dos dados e a generalização do modelo. Ao final, a escolha de λ define o equilíbrio entre viés e variância, crucial para garantir que o modelo seja robusto e confiável.

7.6 REGRESSÃO LOCAL

A regressão local é uma abordagem flexível e não linear para ajustar modelos a dados, enfocando na estimativa localizada em torno de um ponto específico x_0, usando apenas observações próximas a esse ponto. Esse método ajusta um modelo de mínimos quadrados ponderados localmente para cada novo ponto de predição x_0, onde os pesos são especificados para cada observação com base em sua proximidade ao ponto x_0.

Algoritmo 7.6 *Regressão Local em* $X = x_0$

1. Reúna a fração $s = k/n$ de pontos de treinamento cujos x_i estão mais próximos de x_0.

2. Os pesos K_{i0} são atribuídos às observações baseados em uma função de ponderação K, que diminui conforme a distância entre a observação e o ponto x_0 aumenta. A escolha dessa função é crucial e pode variar dependendo do contexto específico dos dados.

3. Para cada ponto de interesse x_0, um novo modelo de mínimos quadrados ponderados é ajustado minimizando a soma dos quadrados dos resíduos ponderados:

$$\sum_{i=1}^{n} K_{i0}(y_i - \beta_0 - \beta_1 x_i)^2 \tag{44}$$

4. O valor ajustado em x_0 é dado por $\hat{f}(x_0) = \hat{\beta} + \hat{\beta}_1 x_0$.

7.6.1 Processo Analítico da Regressão Local

1. Definição dos Pesos: Os pesos K_{i0} são atribuídos às observações baseados em uma função de ponderação K, que diminui conforme a distância entre a observação e o ponto x_0 aumenta. A escolha dessa função é crucial e pode variar dependendo do contexto específico dos dados.

2. Ajuste de Mínimos Quadrados Ponderados: Para cada ponto de interesse x_0, um novo modelo de mínimos quadrados ponderados é ajustado minimizando a soma dos quadrados dos resíduos ponderados:

$$\text{minimizar} \quad \sum_{i=1}^{n} K_{i0}(y_i - g(x_i))^2 \tag{45}$$

onde y_i são os valores observados e $g(x_i)$ são os valores ajustados pelo modelo.

3. Escolha do Span s: O span s controla a proporção de pontos usados para calcular a regressão local em x_0. Um valor pequeno de s resulta em um ajuste mais local e flexível, enquanto um valor grande de s leva a um ajuste mais global e suave. A seleção de s é análoga à escolha do parâmetro de suavização λ em splines de suavização.

4. Validação Cruzada: O span s pode ser otimizado através de validação cruzada para determinar qual valor proporciona o melhor equilíbrio entre viés e variância, minimizando assim o erro de predição.

7.6.2 Aplicações e Generalizações

A regressão local pode ser adaptada para múltiplas variáveis ou para ajustes que são globais em algumas variáveis e locais em outras. Esses modelos com coeficientes variáveis são particularmente úteis quando se deseja adaptar um modelo com base nos dados mais recentes. Além disso, a abordagem pode ser estendida para ajustes bivariados ou multivariados, usando vizinhanças de dimensões superiores para ajustar modelos lineares às observações próximas de cada ponto alvo no espaço multidimensional.

Esta metodologia, contudo, pode enfrentar dificuldades em dimensões altas devido à escassez de dados próximos a x_0, um problema conhecido como a "maldição da dimensionalidade", que também afeta métodos como a regressão dos vizinhos mais próximos (*Nearest Neighbor Regression*).

8 MÉTODOS BASEADOS EM ÁRVORE

Nesse capítulo, descreveremos métodos baseados em árvore para regressão e classificação. Estes envolvem estratificar ou segmentar o espaço preditor em regiões simples. Uma vez que as regras usadas para segmentar o espaço preditor podem ser sumarizados numa árvore, esses tipos de abordagens são conhecidos como métodos de árvore de decisão.

Estes métodos são simples e úteis para interpretação, embora não sejam tipicamente competitivos com as melhores abordagens de aprendizado supervisionado.

8.1 O BÁSICO DE ÁRVORES DE DECISÃO

As árvores de decisão podem ser aplicadas a problemas de regressão e classificação. Primeiro, consideramos os problemas de regressão e depois passamos para a classificação.

8.1.1 Árvores de Regressão

8.1.1.1 Predição via Estratificação do Espaço de Atributos

Discutiremos o processo de construção de uma árvore de regressão. A grosso modo, existem duas etapas:

1. Dividimos o espaço preditor − isto é, o conjunto de possíveis valores para $X_1, X_2, \ldots, X_p$ − em J distintas e não sobrepostas regiões, $R_1, R_2, \ldots, R_j$.

2. Para cada observação que é segmentada na região R_j, fazemos a mesma predição, que é simplesmente a média dos valores de resposta para as observações de treinamento em R_j.

Como encontramos as regiões $R_1, \ldots, R_J$? Na teoria, as regiões podem assumir qualquer formato. Entretanto, escolhemos dividir esse espaço em retângulos de alta dimensionalidade, ou *caixas*, para simplicidade de interpretação dos resultados do modelo preditivo. O objetivo é encontrar $R_1, \ldots, R_J$ que minimizam o RSS, dado por:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2, \tag{46}$$

onde $\hat{y}_{R_j}$ é média da resposta das observações de treinamento dentro da j-ésima caixa. Infelizmente, é computacionalmente inviável considerar cada possível partição do espaço

de atributos em J caixas. Por isso, tomamos a abordagem *top-down*[35] e *greedy*[36] de *divisão binária recursiva*[37].

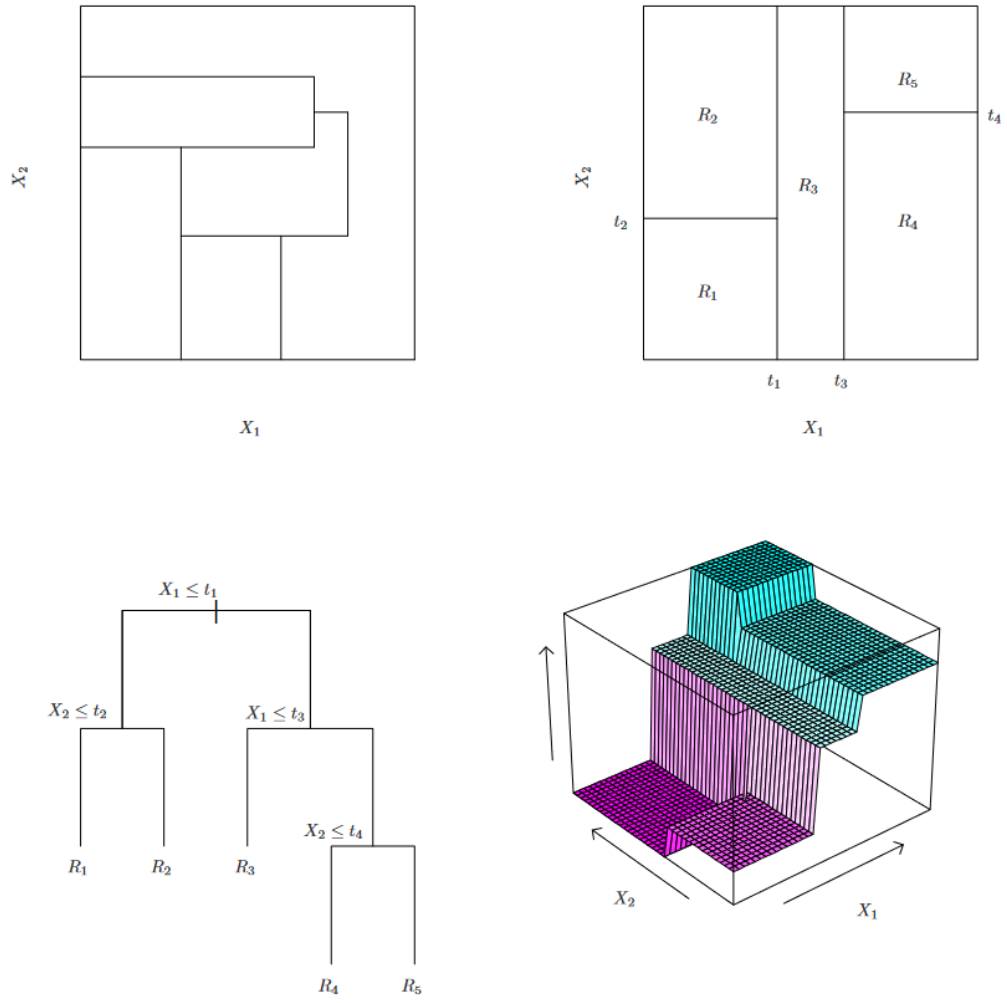

Figura 8: Canto superior esquerdo: uma partição de um espaço bidimensional que não poderia ser resultado de uma divisão binária recursiva. Canto superior direito: a saída de uma divisão binária recursiva em um exemplo bidimensional. Canto inferior esquerdo: a árvore correspondente da partição do gráfico do canto superior direito. Canto inferior direito: um gráfico em perspectiva da superfície de previsão correspondente àquela árvore.

Para performar a divisão binária recursiva, selecionamos, primeiro, o preditor X_j e o *cutpoint* s tal que a divisão do espaço de atributos em regiões $\{X|X_j < s\}$[38] e $\{X|X_j \geq$

[35]Isso traduz: de cima para baixo. A abordagem é *top-down* porque começa do topo da árvore, onde todas as observações pertencem a uma mesma região, e então o espaço de variáveis é dividido sucessivamente.

[36]A tradução literal de *greedy* é ganancioso. É *greedy* porque, a cada passo da construção do modelo, a melhor divisão daquele passo específico é escolhida em vez de escolher uma divisão que resultará num melhor modelo em passsos seguintes.

[37]Em inglês, recursive binary splitting.

[38]A notação $\{X|X_j < s\}$ significa a região do espaço preditor em que X_j assume um valor menor que s.

$s\}$ tende a maior redução possível de RSS. Ou seja, consideramos todos os preditores $X_1, \ldots, X_p$ e todos os possíveis valores de *cutpoint* s para cada um dos preditores e, assim, escolhemos o preditor e o *cutpoint* que produz árvore com o menor RSS. Em maior detalhes, para qualquer j e s, definimos o par de meio-planos

$$R_1(j,s) = \{X|X_j < s\} \quad e \quad R_2(j,s) = \{X|X_j \geq s\}, \tag{47}$$

e procuramos o valor de j e s que minimizam a equação

$$\sum_{i:\, x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:\, x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \tag{48}$$

onde $\hat{y}_{R_1}$ é a resposta média para as observações de treinamento em $R_1(j,s)$, e $\hat{y}_{R_2}$ é a resposta média para as observações de treinamento em $R_2(j,s)$.

Em seguida, repetimos o processo: encontrar o melhor preditor e o melhor *cutpoint* de forma a minimizar o RSS de cada uma das regiões resultantes. Dessa vez, em contrapartida, em vez de dividir o espaço de preditores inteiro, dividiremos somente uma das duas regiões identificadas anteriormente. Esse processo continua até um critério de parada ser alcançado.

A Figura (8) representa esse processo.

8.1.1.2 Poda da Árvore

O processo supracitado pode produzir boas predições no conjunto de treinamento, entretanto ele tende a realizar um *overfitting* dos dados, já que a árvore resultante pode ser muito complexa.

Nesse sentido, uma árvore menor com poucas regiões $R_1, \ldots, R_J$ pode gerar uma menor variância e melhor interpretabilidade com o custo sendo o aumento pequeno do viés.

Uma forma de fazer isso é escolher um valor limite de diminuição de RSS, o qual será responsável por parar o processo de recursão binária da árvore quando uma divisão de região diminuir o RSS menor que o valor limite. Essa maneira pode gerar árvores menores, contudo ela pode impedir de encontrar uma divisão que diminui bastante o RSS, mas que está logo em seguida de uma divisão insignificante.

Cost complexity pruning, também conhecido como *weakest link pruning*, é uma forma de utilizarmos validação cruzada com um pequeno conjunto de *subtrees* a fim de "podarmos"a árvore. Em vez de considerarmos cada possível *subtree*, consideramos uma sequência de

árvores indexadas por um parâmetro de afinação não-negativo α. Para cada valor de α, há uma subárvore correspondente $T \subset T_0$ tal que

$$\sum_{m=1}^{|T|} \sum_{i:\, x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha |T| \tag{49}$$

a equação (49) é a menor possível. Aqui $|T|$ indica o número de nós terminais[39] da árvore T, R_m é o retângulo que corresponde ao m-ésimo nó terminal, e $\hat{y}_{R_m}$ é a resposta predita associada à R_m. A equação (49) é parecida com o método lasso, no qual uma formulação similar é usada para controlar a complexidade do modelo linear.

O parâmetro α controla o *trade-off* entre a complexidade e o ajuste do modelo. Quando $\alpha = 0$, a subárvore T vai ser igual à T_0. Conforme α aumenta, por outro lado, a quantidade de nós folhas tende a diminuir.

O Algoritmo 8.1.1.2 sumariza esse processo.

Algoritmo 8.1.1.2 *Construindo uma Árvore de Regressão*

---

1. Use divisão binária recursiva para construir uma árvore grande com os dados de treinamento, parando somente quando cada nó terminal tiver menos que o número mínimo de observações.

2. Aplique *cost complexity pruning* a essa árvore grande a fim de obter uma sequência de melhores *subtrees*, em função de α.

3. Use *K-fold cross-validation* para escolher α. Isto é, divida as observações de treinamento em K camadas. Para cada $k = 1, \ldots, K$:
   (a) Repita os Passos 1 e 2 em todos menos na k-ésima camada dos dados de treino.
   (b) Avalie o erro quadrático médio (MSE) da predição nos dados deixados de fora da k-ésima camada, em função de α.

   Faça a média dos resultados para cada valor de α, e escolha um α que minimize o erro médio.

4. Escolha a subárvore do Passo 2 que corresponde ao valor escolhido de α.

---

[39] Os nós terminais de uma árvore também podem ser chamados de nós folhas.

8.1.2 Árvores de Classificação

Uma árvore de classificação é muito similar com uma árvore de regressão, diferenciando-se pelo fato de que é usada para respostas qualitativas. Além disso, a árvore de classificação utiliza - em vez da resposta média das observações - a classe mais comum da região a qual as observações pertencem.

Assim como no cenário de regressão, usa-se a divisão binária recursiva para criar a árvore de classificação. Entretanto, no cenário de classificação, o RSS não pode ser usado como critério das divisões binárias. Nesse contexto, a alternativa natural para o RSS é a taxa de erro de classificação:

$$E = 1 - \max_k(\hat{p}_{mk}), \tag{50}$$

onde $\hat{p}_{mk}$ representa a proporção das observações de treinamento da m-ésima região que é da k-ésima classe. Entretanto, esse erro de classificação não é suficientemente sensível para a construção de árvores, e - na prática - duas outras medidas são preferíveis:

1. O *índice de Gini* é definido por

$$G = \sum_{k=1}^{K} \hat{p}_{km}(1 - \hat{p}_{mk}), \tag{51}$$

   uma medida da variância total entre as K classes. Não é difícil de ver que o *Gini index* assume um valor pequeno se todos os $\hat{p}_{mk}$ são próximos de zero ou um. Por essa razão, o índice de Gini é referido como uma medida de pureza do nó: um valor pequeno indica que o nó contém predominantemente observações de uma classe única.

2. Uma alternativa para o índice de Gini é a *entropia*, dada por

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}. \tag{52}$$

   Uma vez que $0 \leq \hat{p}_{mk} \leq 1$, sabe-se que $0 \leq -\hat{p}_{mk} \log \hat{p}_{mk}$. A entropia vai assumir um valor próximo de zero se todos os $\hat{p}_{mk}$ são próximos de zero ou um. Portanto, a entropia tem um valor pequeno quando o m-ésimo nó é puro. O índice de Gini e a entropia são bem próximos numericamente.

Qualquer uma dessas três abordagens podem ser usadas para a poda de uma árvore,

mas é preferível utilizar a taxa de erro de classificação se a acurácia de predição de uma árvore podada é o objetivo.

8.1.3 Árvores X Modelos Lineares

Qual modelo é melhor? Depende do problema: se o relacionamento entre os atributos e a resposta é bem aproximado por um modelo linear, então uma abordagem como regressão linear vai funcionar bem e vai superar em performance um método como árvore de regressão, que não explora essa estrutura linear. Se, contudo, há um relacionamento não-linear e complexo entre as variáveis independentes e a resposta, árvores de decisão podem ter melhores performances do que as abordagens clássicas.

Por outro lado, outras considerações podem estar em jogo: por exemplo, a predição usando uma árvore pode ser preferível pela interpretabilidade e visualização.

8.1.4 Vantagens e Desvantagens das Árvores

Vantagens:

- ▲ Algumas pessoas acreditam que as árvores de decisão assemelha-se à tomada de decisão humana mais do que as outras abordagens.
- ▲ Árvores podem ser vistas graficamente e são mais facilmente interpretadas.
- ▲ Árvores podem lidar facilmente com preditores qualitativos sem a necessidade de criar variáveis *dummy*.

Desvantagens:

- ▼ Árvores geralmente não tem o mesmo nível de acurácia preditiva que outras abordagens.
- ▼ Árvores podem ser muito não-robustas, ou seja, uma pequena mudança nos dados podem causar uma grande mudança na árvore final estimada.

8.2 *BAGGING, RANDOM FORESTS, BOOSTING* E *BAYESIAN ADDITIVE REGRESSION TREES*

Um método *ensemble* é uma abordagem que combina muitos "blocos de construção", ou melhor, modelos simples a fim de obter um único e potencialmente poderoso modelo. Esses modelos simples são conhecidos - por vezes - como fracos aprendizes, uma vez que eles podem levar à predições medíocres quando sozinhos.

Discutiremos agora *bagging, random forests, boosting*, e *Bayesian additive regression*

trees. Esses são métodos *ensemble* para os quais o bloco de construção simples é uma regressão ou uma árvore de classificação.

8.2.1 *Bagging*

O bootstrap, introduzido no Capítulo 5, é uma ideia extremamente poderosa: pode ser usada em muitas situações nas quais é difícil computar diretamente o desvio padrão de uma quantia de interesse. No caso do bagging, veremos que o bootstrap pode ser usado para melhorar métodos de aprendizado estatístico.

Bootstrap aggregation, ou *bagging*, é um procedimento de uso geral para reduzir a variância de um método de aprendizagem estatística.

Nessa abordagem, geramos B diferentes *bootstrapped data sets* de treinamento. Em seguida, treinamos nosso método no b-ésimo *bootstrapped data set* de treinamento para encontrarmos $\hat{f}^{*b}(x)$ e - por fim - fazemos a média de todas as predições, encontrando

$$\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x). \tag{53}$$

.

Enquanto bagging pode melhorar as predições para muitos métodos de regressão, ele é particularmente útil para as árvores de decisão.

Para aplicar bagging em árvores de regressão, simplesmente utilizamos a abordagem supracitada. As árvores das amostras bootstrap não são podadas, levando - portanto - à alta variância, mas baixo viés. Calculando a média dessas B árvores reduz a variância, gerando - potencialmente - melhorias impressionantes.

Para utilizar bagging em árvores de classificação, podemos gravar - dada uma observação de teste - a classe predita por cada uma das B árvores, e, então, tomar uma votação majoritária: a predição geral é a classe que mais ocorre entre as B predições.

8.2.1.1 Out-of-Bag Error Estimation

Pode-se demonstrar que, em média, cada árvore bagging usa aproximadamente dois terços das observações. O um terço restante não usado para ajustar uma *bagged tree* é chamada de observações *out-of-bag*[40] (OOB).

Podemos predizer a resposta para a i-ésima observação usando cada uma das árvores nas quais aquela observação é OOB. Calculando a média das predições (regressão) ou

[40] O significado desse termo é fora da sacola, tradução que faz sentido dado o contexto.

o voto majoritário (classificação), obtemos uma única predição OOB para a i-ésima observação. Fazendo isso para as n observações, podemos conseguir o MSE OOB geral ou o erro de classificação, que são estimativas válidas do erro de teste.

8.2.1.2 Medidas de Importância de Variável

Não é difícil de ver que bagging melhora a acurácia de predição em detrimento da interpretabilidade. Em contrapartida, o bagging permite obter um sumário geral da importância de cada preditor usando o RSS (regressão) ou o índice de Gini (classificação). Dada a quantidade total que o RSS, ou o índice de Gini, é diminuído por causa das divisões de um preditor em todas as B árvores, podemos saber a importância de um preditor. Quanto maior a diminuição, maior a importância; e, quanto menor a diminuição, menor a importância.

8.2.2 *Random Forests*

Assim como no bagging, construímos um número de árvores de decisão com amostras *bootstrapped* de treinamento. Mas - na construção dessas árvores de decisão - a cada vez que a divisão da árvore é considerada, uma *amostra aleatória* dos m preditores é escolhida como os candidatos dessa divisão. Ou seja, a divisão só pode escolher um entre os m preditores, tipicamente sendo $m \approx \sqrt{p}$.

Isso parece algo sem sentido, mas tem uma lógica inteligente. Suponha que existe um preditor forte no conjunto de dados, seguido por um número de outros preditores moderadamente fortes. Na coleção de árvores bagging, a maioria vai utilizar esse forte preditor na divisão primária e, consequentemente, vão ser bem similares entre si. Portanto, as predições dessas árvores vão ser altamente correlacionadas, levando a uma redução menor da variância do que se as predições fosse não-correlacionadas.

Random Forests superam esse problema ao forçar cada divisão a considerar somente um *subset* dos preditores. Podemos pensar nesse processo como o ato de *descorrelacionar* as árvores.

A maior diferença entre bagging e *random forests* é a escolha do subconjunto de preditores m. Se $m = p$ em uma *random forest*, então esse método torna-se o bagging.

8.2.3 *Boosting*

Boosting funciona de uma maneira similar ao bagging, com a exceção de que as árvores são construídas sequencialmente: cada árvore "cresce" usando a informação de árvores anteriores. Boosting não envolve amostragem bootstrap: em vez disso, cada árvore é ajustada com uma versão modificada do *data set* original.

O método boosting é sumarizado no Algoritmo 8.2.3.

Algoritmo 8.2.3 *Boosting para Árvores de Regressão*

---

1. Coloque $\hat{f}(x) = 0$ e $r_i = y_i$ para todo i do conjunto de treinamento.

2. Para $b = 1, 2, \ldots, B$, repita:
   (a) Ajuste a árvore $\hat{f}^b$ de d divisões ($d+1$ nós terminais) com os dados (X, r).
   (b) Atualize $\hat{f}$ ao adicionar uma versão encolhida da nova árvore:

$$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x). \tag{54}$$

   (c) Atualize os resíduos,

$$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i). \tag{55}$$

3. Produza o modelo otimizado,

$$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x). \tag{56}$$

Boosting tem três parâmetros de afinação:

1. O número B de árvores. Usamos validação cruzada para selecionar B.

2. O parâmetro de encolhimento λ, um pequeno número positivo, que controla a taxa de aprendizado do boosting. Valores típicos são 0.01 ou 0.001, e a escolha depende do problema.

3. O número d de divisões de cada árvore, que controla a complexidade do *boosted ensemble*. Geralmente, d é a profundidade da interação e controla a ordem de interação do modelo boosting, uma vez que d divisões podem envolver no máximo d variáveis.

8.2.4 *Bayesian Additive Regression Trees*

Por simplicidade, apresentaremos *Bayesian Additive Regression Trees* (BART) para regressão. Antes de introduzirmos o algoritmo BART, definimos algumas notações.

Deixe K denotar o número de árvores de regressão, e B o número de iterações que o algoritmo BART vai funcionar.

A notação $\hat{f}_k^b(x)$ representa a predição com x para a k-ésima árvore de regressão usada na b-ésima iteração. No fim de cada iteração, as K árvores da iteração serão somadas,

ou seja, $\hat{f}^b(x) = \sum_{k=1}^{K} \hat{f}_k^b(x)$ para $b = 1, \ldots, B$.

Na primeira iteração do algoritmo BART, todas as árvores são inicializadas para ter um único nó raiz, com $\hat{f}_k^1(x) = \frac{1}{nK} \sum_{i=1}^{n} y_i$, a resposta média dividida pelo número total de árvores. Portanto, $\hat{f}^1(x) = \sum_{k=1}^{K} \hat{f}_k^1(x) = \frac{1}{n} \sum_{i=1}^{n} y_i$

Nas iterações seguintes, BART atualiza cada uma das K árvores, uma de cada vez. Na b-ésima iteração, para atualizar a k-ésima árvore, subtraímos de cada resposta média as predições de todas menos da k-ésima árvore, a fim de obter o resíduo parcial

$$r_i = y_i - \sum_{k'<k} \hat{f}_{k'}^b(x_i) - \sum_{k'<k} \hat{f}_{k'}^{b-1}(x_i)$$

para a i-ésima observação $i = 1, \ldots, n$.

O produto do BART é a coleção dos modelos preditos,

$$\hat{f}^b(x) = \sum_{k=1}^{K} \hat{f}_k^b(x), \text{ para } b = 1, 2, \ldots, B.$$

Normalmente, descartamos os primeiros desses modelos de previsão, uma vez que os modelos obtidos nas iterações anteriores – conhecidos como período *burn-in* – tendem a não fornecer resultados muito bons. Deixe L denotar o número de iterações de *burn-in*. Por exemplo, podemos considerar $L = 200$. Então, para obter uma única previsão, simplesmente calculamos a média após as iterações de *burn-in*, $\hat{f}^b(x) = \sum_{k=1}^{K} \hat{f}_k^b(x)$.

O Algoritmo 8.2.4 sumariza o processo.

Algoritmo 8.2.4 *Bayesian Additive Regression Trees*

1. Deixe $\hat{f}_1^1(x) = \hat{f}_2^1 = \cdots = \hat{f}_K^1 = \frac{1}{nK}\sum_{i=1}^n y_i$.

2. Calcule $\hat{f}^1(x) = \sum_{k=1}^K \hat{f}_k^1(x) = \frac{1}{n}\sum_{i=1}^n y_i$.

3. Para $b = 2, \ldots, B$:
   (a) Para $k = 1, 2, \ldots, K$:
      i. Para $i = 1, \ldots, n$, calculamos o resíduo parcial atual

$$r_i = y_i - \sum_{k'<k} \hat{f}_{k'}^b(x_i) - \sum_{k'<k} \hat{f}_{k'}^{b-1}(x_i)$$

      ii. Ajustamos a nova árvore, $\hat{f}_k^b(x)$, para r_i, ao aleatoriamente perturbar a k-ésima árvore da iteração anterior, $\hat{f}_k^{b-1}(x)$. Perturbações que melhoram o modelo são favorecidas.
   (b) Calcule $\hat{f}^b(x) = \sum_{k=1}^K \hat{f}_k^b(x)$.
4. Calcule a média após L amostras *burn-in*,

$$\hat{f}(x) = \frac{1}{B-L}\sum_{b=L+1}^{B} \hat{f}^b(x).$$

8.2.5 Sumário dos Métodos *Ensemble* de Árvores

Vimos quatro abordagens *ensemble* de árvores: *bagging*, *random forests*, *boosting* e BART.

- Em bagging, as árvores são construídas independentemente a partir de amostras randômicas de observações. Consequentemente, as árvores tendem a serem similares entre si. Portanto, bagging pode falhar em explorar completamente o espaço do modelo.

- Em *random forests*, as árvores são novamente construídas a partir de amostras aleatórias de observações. Entretanto, cada divisão de cada árvore é performada usando um *subset* randômico dos atributos, descorrelacionando as árvores e levando a uma exploração mais completa do espaço do modelo.

- Em boosting, usamos somente os dados originais e não construímos novas amostras aleatórias. As árvores são construídas sucessivamente, usando uma abordagem de aprendizado "lento": cada nova árvore é ajustada com o sinal que é deixado pelas árvores anteriores e encolhida antes de ser usada.

- Em BART, novamente utilizamos somente os dados originais e construímos as árvores sucessivamente. No entanto, cada árvore é perturbada para evitar mínimos locais e conseguir uma exploração mais completa do espaço do modelo.