**Marcel Chacon Gonçalves**

# Tópicos em aprendizagem estatística de máquinas com aplicações em finanças

Niterói - RJ, Brasil

15 de dezembro de 2020

Universidade Federal Fluminense

**Marcel Chacon Gonçalves**

# Tópicos em aprendizagem estatística de máquinas com aplicações em finanças

**Trabalho de Conclusão de Curso**

Monografia apresentada para obtenção do grau de Bacharel em Estatística pela Universidade Federal Fluminense.

Orientador(a): Prof. Dr. Márcio Watanabe Alves de Souza

Niterói - RJ, Brasil

15 de dezembro de 2020

Universidade Federal Fluminense

**Marcel Chacon Gonçalves**

# Tópicos em aprendizagem estatística de máquinas com aplicações em finanças

Monografia de Projeto Final de Graduação sob o título *"Tópicos em aprendizagem estatística de máquinas com aplicações em finanças"*, defendida por Marcel Chacon Gonçalves e aprovada em 10 de dezembro de 2020, na cidade de Niterói,no Estado do Rio de Janeiro, pela banca examinadora constituída pelos professores:

---

**Prof. Dr. Márcio Watanabe Alves de Souza**
Departamento de Estatística – UFF

---

**Prof. Dr. Jessica Quintanilha Kubrusly**
Departamento de Estatística - UFF

---

**Prof. Dr. Douglas Rodrigues Pinto**
Departamento de Estatística - UFF

Niterói, 15 de dezembro de 2020

Ficha catalográfica automática - SDC/BIME
Gerada com informações fornecidas pelo autor

G635t Gonçalves, Marcel Chacon
Tópicos em aprendizagem estatística de máquinas com aplicações em finanças / Marcel Chacon Gonçalves ; Márcio Watanabe Alves de Souza, orientador. Niterói, 2020.
61 f.

Trabalho de Conclusão de Curso (Graduação em Estatística)-Universidade Federal Fluminense, Instituto de Matemática e Estatística, Niterói, 2020.

1. Aprendizado de máquinas. 2. Regressão logística. 3. Regressão linear múltipla. 4. Produção intelectual. I. Souza, Márcio Watanabe Alves de, orientador. II. Universidade Federal Fluminense. Instituto de Matemática e Estatística. III. Título.

CDD -

Bibliotecário responsável: Sandra Lopes Coelho - CRB7/3389

# Resumo

A cada dia, mais e mais empresas buscam formas de aproveitar a grande quantidade de dados disponíveis para melhorar o resultado de seus negócios. Nesse cenário, as técnicas de machine learning, ou aprendizagem de máquinas, tem se destacado por implementar sistemas que buscam automatizar a incorporação de novas informações disponíveis nos dados, proporcionando uma utilização imediata nos processos de tomada de decisão. Nesse projeto de trabalho de conclusão de curso estudaremos alguns dos principais conceitos e modelos estatísticos utilizados em aprendizagem de máquinas e suas respectivas ferramentas no R, focando nos processos de análise de dados e predição, bem como na sua utilização prática em problemas de recente aplicação, como na área de finanças em que modelos estatísticos de machine learning tem substituído métodos clássicos como as medidas de credit score em análise de crédito. Então foi comparado o desempenho dos métodos através das amostras testes criadas com os dados disponíveis. A escolha do método pode ser diferente dependendo da métrica usada como parâmetro, e isso depende do objetivo da análise.

Palavras-chave: Aprendizagem Estatística. Regressão Logística. KNN. LDA e QDA. Métodos baseados em árvores.

# Agradecimentos

Ao professor e orientador Márcio Watanabe Alves de Souza pelo tempo dedicado e ensinamentos ao longo desse trabalho.

Aos professores Doutores Jéssica Kubrusly e Douglas Rodrigues que gentilmente aceitaram participar da Banca Examinadora.

Aos professores do departamento de computação onde cursei algumas disciplinas e a todos os professores do departamento de Estatística da UFF, em especial aos professores Ana Beatriz Monteiro Fonseca, Ana Maria Lima de Farias, Douglas Rodrigues Pinto, Eduardo Ferioli Gomes, Fábio Nogueira Demarqui, Jessica Quintanilha Kubrusly, Jony Arrais Pinto Junior, José Rodrigo de Moraes, Karina Yuriko Yaginuma, Keila Mara Cassiano, Luciane Alcoforado, Ludmilla da Silva Viana Jacobson, Luiz Guillermo Coca Velarde, Márcio Watanabe Alves de Souza, Marco Aurélio dos Santos Sanfins, Maria Cristina Bessa Moreira, Mariana Albi de Oliveira Souza, Núbia Karla de Oliveira Almeida e Patrícia Lusié Velozo da Costa, Os quais tive a honra de ser aluno. Obrigado pelos ensinamentos.

Aos professores do Departamento de Matemática da UERJ, campus Faculdade de Formação de Professores, onde fiz minha graduação de Licenciatura em Matemática.

Aos professores do curso PROFMAT da UFRJ.

Aos amigos que acompanharam essa trajetória e partilharam dos desafios envolvidos, em especial a Akauã, que deu a ideia de fazer essa graduação.

Aos meus familiares que contribuíram para eu chegar até aqui, desde o meu nascimento até a fase adulta.

A todos que, de alguma maneira, contribuíram para a realização deste trabalho.

# Sumário

# Lista de Figuras

# Lista de Tabelas

# 1 Introdução

## 1.1 Motivação

Atualmente há um aumento na quantidade de dados disponíveis para as pessoas, as empresas, órgãos governamentais e não governamentais, que querem usar as informações obtidas através dos dados para auxiliar na tomada de decisões sobre algum problema de interesse como por exemplo políticas públicas, investimentos internos entre outras questões. Como conseguir lidar com isso? A aprendizagem de máquina pode auxiliar na tomada de decisão, oferecendo sistemas automáticos de resolução de problemas, baseados em teoria estatística.

Estudaremos métodos para medir a qualidade da estimação dos modelos, fazendo comparações dos resultados dos diferentes métodos mostrados quando aplicados a um mesmo conjunto de dados. Utilizando o software R, aplicaremos os conceitos e métodos estudados a um conjunto de dados reais do mercado financeiro para estudar o problema de estimar a taxa de inadimplência, comparando os resultados obtidos entre os diferentes modelos.

## 1.2 Aprendizagem estatística

Suponha que há $p$ variáveis preditoras $X^T = (X_1, \ldots, X_p)$ e uma variável resposta $Y$. Assumimos que há alguma relação entre $X$ e $Y$ que pode ser escrita de uma forma geral por

$$Y = f(X) + \epsilon,$$

onde $f$ é uma função fixa mas desconhecida de $X_1, X_2, ..., X_p$, e $\epsilon$ é o termo do *erro aleatório*, que é independente de $X$ e tem média zero. Dessa forma, $f$ representa a informação sistemática fornecida por $X$ sobre $Y$. Em essência, um problema fundamental

em aprendizagem estatística se refere aos métodos para a estimação de $f$ (JAMES et al., 2013).

Há duas razões principais para se estimar $f$: *predição* e *inferência*.

Para Predição é quando o interesse está em obter um estimador para $Y$ a partir de uma certa entrada $\mathbf{x} = (x_1, x_2, ..., x_p)$, ou seja, obter $\hat{Y}$ tal que,

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

onde $\hat{f}$ representa nossa estimativa para $f$ e $\hat{Y}$ representa o resultado da predição para $Y$. Nesse caso, deseja-se saber apenas o valor da previsão do modelo, não importando a relação entre as variáveis preditoras, ou seja, o interesse maior é obter $\hat{Y}$ o mais próximo possível de $Y$ sem se preocupar como é a forma de $\hat{f}$. Então, pode-se usar modelos mais *flexíveis*, ou seja, com mais parâmetros ou que captam características mais irregulares dos dados, que são em geral mais precisos, porém mais complexos, perdendo em interpretação (JAMES et al., 2013). A flexibilidade está ligada, em muitos casos, ao número de parâmetros usados na função.

Para inferência é quando o interesse está em entender a relação entre $X$ e $Y$, ou mais especificamente, entender como $Y$ muda em função de $X_1, X_2, ..., X_p$. Ou seja, o interesse principal é estimar $f$ e de modo a obter uma $\hat{f}$ o mais simples e interpretável possível. Desse ponto de vista, deseja-se saber como cada variável $X_j$ preditora interfere individualmente na variável resposta $Y$. Nesse caso, os modelos mais simples são mais recomendados pois possuem maior interpretabilidade das variáveis preditoras.

Em geral há três tipos de aprendizado de máquinas. O *supervisionado*, quando são amostras rotuladas, e assim temos conhecimento do valor verdadeiro de cada uma das observações para essa variável resposta. O tipo *não supervisionado* é quando não temos os valores da variável resposta para as observações. Assim, as análises são feitas de forma diferente da supervisionada, quando temos o objetivo da predição na resposta $Y$. E o tipo de aprendizagem *por reforço*, quando o modelo toma uma sequência de decisões visando maximizar uma pontuação final, por exemplo. Neste trabalho trataremos do aprendizado supervisionado.

Dependendo do objetivo da modelagem são feitas escolhas e procedimentos de maneira à atingir algum desses objetivos.

## 1.3 Objetivos

### Objetivos Gerais

O objetivo deste trabalho é apresentar alguns dos métodos de aprendizagem de estatística, como Regressão múltipla, Regressão logística, Análise de Discriminante Linear e Quadrática, $K$ Vizinhos Mais Próximos, e os métodos *Bagging*, *Random Forest* e *gradient boosting*, que geram modelos baseados em árvores de decisão e entender um pouco dos seus mecanismos de funcionamento. Tais métodos são comuns em algoritmos de aprendizagem de máquina, ou *machine learning*, de forma a proporcionar uma introdução ao tema. Estudaremos critérios para comparação e seleção de modelos, através da mensuração da qualidade dos ajustes. Além disso, aplicaremos os modelos a um conjunto de dados reais do mercado financeiro com a utilização do software R.

### Objetivos Específicos

- Estudar os conceitos de aprendizagem de máquinas;
- Estudar alguns métodos de aprendizagem de máquinas;
- Compreender como analisar corretamente o desempenho de um método;
- Aplicar os métodos estudados em um banco de dados real para comparação.

## 1.4 Organização

Este trabalho contém uma introdução à aprendizagem de máquina no capítulo 1, com os termos comuns usados e algumas definições. O capítulo 2 apresenta os materiais e métodos utilizados para o desenvolvimento do trabalho, abordando cada técnica estudada e suas principais características. No capítulo 3 é feita uma aplicação dos métodos apresentados em um conjunto de dados, a fim de exemplificar o tema abordado, e no capítulo 4 algumas considerações finais são apresentadas.

# 2 Materiais e Métodos

Ao longo desse trabalho, usaremos notação vetorial e matricial quando couber tal notação. Um dos objetivos da modelagem estatística é entender a relação entre diversas variáveis ou características. Em muitos casos, deseja-se entender o comportamento de uma variável resposta em função de outras, as preditoras. Uma variável preditora é denotada pelo símbolo $X$. Se $X$ é um vetor de variáveis, seus componentes podem ser obtidos com a notação $X_j$. A variável resposta será denotada por $Y$ tanto no caso de resposta quantitativa quanto qualitativa. Valores observados são denotados em letras minúsculas. Então o $i$-ésimo valor observado de $X$ será escrito como $x_i$ (onde $x_i$ pode ser um escalar ou um vetor). Matrizes são representadas por letras maiúsculas em negrito, por exemplo, um conjunto de $N$ entradas de $p$-vetores $x_i$, $i = 1, ..., N$ é representado pela matriz $\mathbf{X}$ de dimensão $N \times p$ (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Em geral, vetores não estão em negrito. Assumindo todos os vetores como vetores coluna, a $i$-ésima linha de $\mathbf{X}$ é $x_i^T$, o vetor transposto de $x_i$.

Frequentemente, o conjunto de dados observados $(x_i, y_i)$, $i = 1, 2, ..., N$ é conhecido como *dados de treino*, usado para ajustar os parâmetros.

Uma variável pode ser classificada em *quantitativa* (numérica) ou *qualitativa* (categórica). Chamaremos de *regressão* quando a variável resposta do problema em questão for quantitativa e de *classificação* quando a resposta for categórica. O tipo das variáveis preditoras é menos importante na maioria dos métodos, pois podem ser aplicados em ambos os casos, apenas tomando o cuidado com a codificação dos níveis das preditoras categóricas.

## 2.1 Qualidade do ajuste

Uma questão discutível é como *medir* a qualidade de um ajuste, de uma função construída por algum método.

Os métodos de estimação de $f$ procuram minimizar o *erro redutível* (definido a seguir) de acordo com algum critério pré-fixado selecionando um método mais apropriado do que outro. Porém, mesmo que se encontre a forma perfeita de $f$, nossa estimativa tomará a forma $\hat{Y} = f(X)$ e ainda conterá erro, pois $Y$ é uma função de $\epsilon$. A quantidade $\epsilon$ pode conter variáveis que não foram mensuradas e que são importantes para a predição de $Y$ (JAMES et al., 2013).

Considerando um estimador $\hat{f}$ e um conjunto de preditoras $X$ e assumindo momentaneamente que ambos $\hat{f}$ e $X$ são fixos, temos que:

$$\begin{aligned} E[(Y - \hat{Y})^2] &= E[(f(X) + \epsilon - \hat{f}(X))^2] = \\ &= E[(f(X) - \hat{f}(X))^2 + 2(f(X) - \hat{f}(X)) \cdot \epsilon + \epsilon^2] = \\ &= E[(f(X) - \hat{f}(X))^2] + 2E[f(X) - \hat{f}(X)]E[\epsilon] + E[\epsilon^2] \end{aligned}$$

Sabemos que $E[\epsilon] = 0$ e que $Var[\epsilon] = E[\epsilon^2] - E^2[\epsilon]$. Então,

$$E[(Y - \hat{Y})^2] = \underbrace{E[(f(X) - \hat{f}(X))^2]}_{} + \underbrace{Var(\epsilon)}_{}$$

onde $E[(Y - \hat{Y})^2]$ representa o *valor esperado* do quadrado da diferença entre o valor predito e o atual valor de $Y$, e $Var(\epsilon)$ é a variância do termo do erro $\epsilon$. Assim, temos a soma em duas partes do erro:

*erro redutível*: $E[(f(X) - \hat{f}(X))^2]$, é o erro que conseguimos reduzir usando métodos de aprendizagem mais precisos.

*erro irredutível*: $Var(\epsilon)$, que pelo modelo adotado é uma constante em $X$, e por isso é chamado de irredutível.

### Erro Quadrático Médio (EQM)

É um dos critérios mais utilizados para medir qualidade de ajuste, no caso de uma regressão. É dado por

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

onde $\hat{f}(x_i)$ é a predição que $\hat{f}$ fornece para a $i$-ésima observação. O valor do EQM será

pequeno se os valores preditos são próximos dos reais, e grande caso contrário.

### Coeficiente de determinação $R^2$

É outra medida usada para qualidade do ajuste, que mede a proporção da variância explicada pelo modelo e assim, independente da escala de $Y$ e assume valores entre 0 e 1. É obtido pela expressão

$$R^2 = \frac{SQT - SQR}{SQT} = 1 - \frac{SQR}{SQT}$$

onde $SQT = \sum_{i=1}^{n}(y_i - \overline{y})^2$ é a *Soma dos Quadrados Totais*, e $SQR = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ é a *Soma dos Quadrados dos Resíduos*.

### Amostra treino e amostra teste

Nesse ponto devemos tomar cuidado para usar adequadamente o critério de medição da qualidade das aproximações. Quando ajustamos um modelo, geralmente o fazemos utilizando um conjunto de observações que têm a função de *treinar* ou ajustar o modelo. Esse conjunto de observações é chamado de amostras treino, ou simplesmente treino. O EQM quando aplicado nos dados de treino, é uma medida nesse conjunto de pontos. Apesar de ser uma medida importante, o objetivo final de qualquer modelo é obter um bom ajuste em pontos que não foram utilizados para ajustar o modelo, por exemplo, em dados futuros ainda não medidos.

Separando um subconjunto dos dados para não ser utilizado no ajuste do modelo, possibilita-nos obter uma estimativa para os erros futuros que cometeremos. Denominamos esse conjunto de pontos de amostra teste, ou simplesmente teste, que são dados que não foram usados para ajustar o modelo.

Em regressão, o EQM de teste é geralmente maior que o EQM de treino. Não há garantias de que o método usado que produz o menor EQM de treino irá gerar o menor EQM de teste (JAMES et al., 2013).

Para mostrar esse fato, considere os gráficos na Figura 1 (JAMES et al., 2013), em que foi plotado o EQM de teste, linha cinza, e o EQM de treino, linha vermelha, para dados simulados de uma $f$, em preto, em função da sua flexibilidade. Quando os dados apresentam um comportamento não linear, observa-se que o comportamento da curva do

EQM de treino é decrescente conforme aumenta-se o número de parâmetros do modelo. Isso ocorre devido ao fato de que com mais parâmetros, a curva estimada $\hat{f}$ consegue se ajustar aos dados, buscando padrões para aproximar as estimativas dos valores reais. Já a curva do EQM de teste em função do número de parâmetros do modelo apresenta um formato de U. Significa que quando aumenta-se a flexibilidade do modelo, inicialmente o erro de teste diminui, porém, a partir de um certo número de parâmetros esse erro começa a aumentar. Essa característica é conhecida como *overfitting*, que ocorre quando o modelo incorpora em $\hat{f}$, variações provenientes da variável independente $\epsilon$. Isso produz dois efeitos: gera um aumento na diferença entre $f$ e $\hat{f}$ e gera uma diminuição do EQM de treino. Assim, apesar do erro de treino diminuir, a aproximação $\hat{f}$ da $f$ piora, provocando um aumento no EQM de teste.

Um dos nossos objetivos é determinar o ponto que minimiza o EQM de teste.

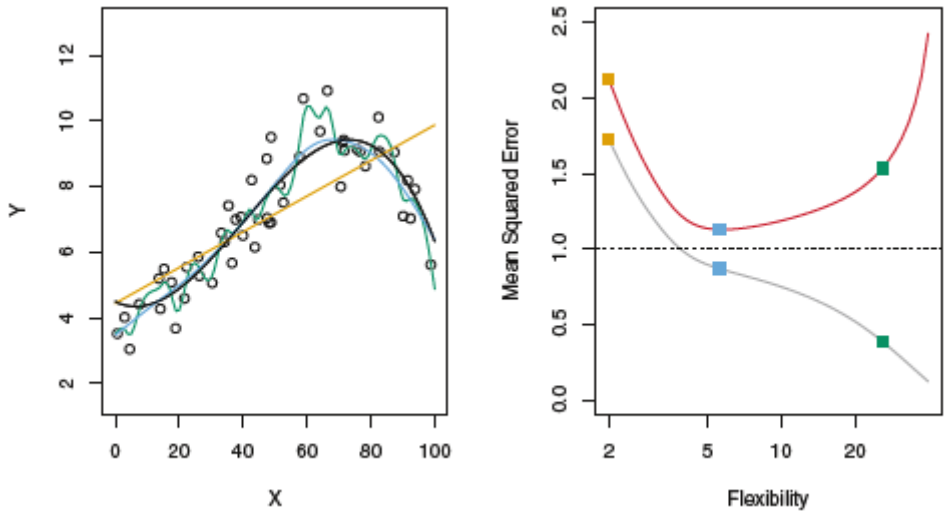


Figura 1: Comparação entre EQM de treino e EQM de teste em dados simulados. (JAMES et al., 2013)

Como escolher os dados de teste e de treino? Em algumas situações os dados de treino são selecionados aleatoriamente. Em outras, alguma característica da natureza dos dados indica que os dados não podem ser escolhidos de maneira aleatória. Por exemplo, em dados de séries temporais, em que o interesse é predizer valores futuros, os dados mais recentes seriam a melhor opção para formarem o conjunto de teste, já que espera-se maior correlação dos dados mais recentes do que os mais antigos para se fazer uma predição.

A proporção dos dados que serão de treino e de teste é critério variável. No caso deste trabalho, escolhemos fazer 75% dos dados para treino e 25% para serem usados como teste, que farão o papel de dados novos. Essa escolha foi feita de maneira aleatória nos

dados, de forma a manter as mesmas proporções da variável resposta em cada grupo.

A seguir, são apresentados os métodos abordados neste trabalho.

## 2.2 Regressão linear

### Introdução

A *regressão linear* é uma abordagem muito simples de métodos de aprendizagem supervisionada (quando sabemos os rótulos ou valores das respostas) sendo particularmente útil para predição de variáveis respostas quantitativas (JAMES et al., 2013).

Ao se abordar um problema, em particular com resposta quantitativa, poderíamos tentar responder algumas perguntas de interesse, como por exemplo: Há uma relação entre a variável resposta e as variáveis preditoras? Quão forte é a relação? Como podemos predizer precisamente a resposta dado certas condições nas preditoras? A relação é linear? Há interação entre as variáveis preditoras?

O modelo linear tem sido um dos pilares da estatística nos últimos 30 anos e continua sendo uma das nossas ferramentas mais importantes (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

### Definição

Dado um vetor de entrada (preditoras) $X^T = (X_1, X_2, \cdots, X_p)$, nosso objetivo é predizer o valor real da resposta $Y$. O modelo de regressão linear tem a forma

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j. \tag{2.1}$$

O modelo linear assume que a função de regressão $E(Y \mid X)$ é linear como função dos $\beta_j$'s ou que o modelo linear é uma aproximação razoável. Aqui os $\beta_j$'s são parâmetros ou coeficientes desconhecidos, e as variáveis $X_j$ podem vir de diferentes fontes: variáveis quantitativas; transformações de variáveis quantitativas, por exemplo $\log X_1$, $\sqrt{X_2}$, $X_1^2$; variáveis qualitativas codificadas numericamente como variáveis *dummy* ou interações entre preditoras, por exemplo $X_3 = X_1 \times X_2$. Independente da natureza dos dados, o modelo é linear nos parâmetros.

### Estimação

Os parâmetros $\beta_0, \beta_1, \cdots, \beta_p$ são desconhecidos e precisam ser estimados a fim de realizar predições ou entender a relação entre as variáveis preditoras e a resposta. Dadas as estimativas $\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_p$, podemos fazer predições usando a expressão

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p. \tag{2.2}$$

Usamos o conjunto de dados de treino $(x_1, y_1), \cdots, (x_n, y_n)$ para estimar os parâmetros $\beta$. Cada $x_i = (x_{i1}, x_{i2}, \cdots, x_{ip})^T$ é um vetor de medidas das características para a $i$-ésima observação.

Uma maneira de estimar esses parâmetros, e a mais popular, segundo Hastie, Tibshirani e Friedman (2009) é pelo método de *Mínimos Quadrados*, que consiste em selecionar $\beta = (\beta_0, \beta_1, \cdots, \beta_p)^T$ para minimizar a Soma dos Quadrados dos Resíduos (SQR).

$$SQR(\beta) = \sum_{i=1}^{n} (y_i - f(x_i))^2 = \sum_{i=1}^{n} \left( y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2. \tag{2.3}$$

Observe que $SQR(\beta)$ é uma função quadrática dos parâmetros e portanto sempre terá um mínimo, porém pode não ser único.

Denotando por $\boldsymbol{X}$ a matriz $N \times (p+1)$ com cada linha sendo um vetor de entrada (com 1 na primeira posição para incluir o intercepto), e também $\boldsymbol{y}$ o vetor com $n$ respostas do conjunto de treino, podemos escrever a Soma dos Quadrados dos Resíduos por (HASTIE; TIBSHIRANI; FRIEDMAN, 2009):

$$SQR(\beta) = (\boldsymbol{y} - \boldsymbol{X}\beta)^T (\boldsymbol{y} - \boldsymbol{X}\beta), \tag{2.4}$$

É uma função quadrática nos $p + 1$ parâmetros. Diferenciando com relação à $\beta$ obtemos as *equações normais* :

$$\frac{\partial SQR}{\partial \beta} = -2\boldsymbol{X}^T (\boldsymbol{y} - \boldsymbol{X}\beta) \tag{2.5}$$

$$\frac{\partial^2 SQR}{\partial \beta \partial \beta^T} = 2\boldsymbol{X}^T \boldsymbol{X} \tag{2.6}$$

Se $\boldsymbol{X}^T \boldsymbol{X}$ é não singular, então a solução única é obtida igualando a primeira derivada a zero

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

e obtemos

$$\hat{\beta} = (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{y}, \tag{2.7}$$

**Distribuição dos Parâmetros**

Para fazer inferências sobre o estimador $\hat{\beta}$, precisamos impor mais hipóteses sobre a distribuição dos dados. Suponha que as observações $y_i$ são não correlacionadas e tem variância constante $\sigma^2$ e que o $x_i$ é fixo (não aleatório). A matriz de variâncias-covariâncias do estimador de mínimos quadrados dos parâmetros é dada por

$$\begin{aligned}
Var(\hat{\beta}) &= Var[(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{y}] = \\
&= (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}X^T((\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^T)^T Var(\boldsymbol{y}) = \\
&= (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}((\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1})^T Var(\boldsymbol{y}) = \\
&= (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}((\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{T})^{-1} Var(\boldsymbol{y}) = \\
&= (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1} Var(\boldsymbol{y}) = \\
&= (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1} Var(\boldsymbol{y}) = (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}\sigma^2
\end{aligned}$$

Portanto,

$$Var(\hat{\beta}) = (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}\sigma^2 \tag{2.8}$$

Um estimador comum da variância $\sigma^2$ é obtido por

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

O valor $n - p - 1$ ao invés de $n$ é para que o estimador seja *não-viesado* de $\sigma^2$, ou seja, $E(\hat{\sigma}^2) = \sigma^2$.

Para fazermos inferências sobre os parâmetros e o modelo, hipóteses adicionais são necessárias. Vamos assumir que (2.1) é o modelo correto para a média, isto é, o valor

esperado condicional de $Y$ é linear em $X_1, \cdots, X_p$. Também assumiremos que os desvios de $Y$ em torno da média são aditivos e com distribuição normal. Assim,

$$\begin{aligned} Y = E(Y \mid X_1, \cdots, X_p) + \epsilon = \\ = \beta_0 + \sum_{j=1}^{p} X_j \beta_j + \epsilon, \end{aligned} \tag{2.9}$$

onde o erro $\epsilon$ é uma variável aleatória Gaussiano com média zero e variância $\sigma^2$, escrevemos $\epsilon \sim N(0, \sigma^2)$.

Sob (2.9), podemos mostrar que

$$\hat{\beta} \sim N\left(\beta, (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}\sigma^2\right), \tag{2.10}$$

que é uma distribuição normal multivariada com parâmetros dados acima. E também temos que

$$(n-p-1)\hat{\sigma^2} \sim \sigma^2 \chi^2_{n-p-1}, \tag{2.11}$$

uma distribuição qui-quadrada com $n-p-1$ graus de liberdade. Ainda mais, $\hat{\beta}$ e $\hat{\sigma^2}$ são estatisticamente independentes.

**Teste de hipóteses sobre os parâmetros**

Usamos as propriedades dessas distribuições para realizar testes de hipóteses e intervalos de confiança para os parâmetros $\beta_j$.

Para testar a hipótese que um particular coeficiente $\beta_j = 0$, calculamos o *Z-score*, que é o coeficiente normalizado, dado por

$$z_j = \frac{\hat{\beta}_j}{\hat{\sigma}^2\sqrt{\upsilon_j}}, \tag{2.12}$$

onde $\upsilon_j$ é o $j$-ésimo elemento da diagonal de $(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-1}$. Sob a hipótese nula de que $\beta_j = 0$, $z_j$ segue uma distribuição $t_{n-p-1}$. Se a variância populacional $\sigma^2$ é conhecida, então $z_j$ teria uma distribuição normal padrão.

Frequentemente, precisamos testar a significância de grupos de coeficientes simultaneamente. Por exemplo, para testar se uma variável categórica com $k$ níveis pode ser

excluída do modelo, precisamos testar se os coeficientes das variáveis *dummy* usadas para representar os níveis podem ser todas consideradas nulas. Nesse caso, usamos a estatística $F$,

$$F = \frac{(SQR_0 - SQR_1)/(p_1 - p_0)}{SQR_1/(n - p_1 - 1)}, \tag{2.13}$$

onde $SQR_1$ é a Soma dos Quadrados dos Resíduos do ajuste de mínimos quadrados do modelo "maior" com $p_1 + 1$ parâmetros e $SQR_0$ do mesmo modelo porém "menor" e aninhado com $p_0 + 1$ parâmetros, tendo $p_1 - p_0$ parâmetros considerados nulos.

A estatística F mede a mudança no SQR pela adição de parâmetros no modelo, e é normalizado por um estimador de $\sigma^2$. Sob as hipóteses de normalidade e de que a hipótese nula do modelo reduzido estar correta, então a estatística F terá uma distribuição $F_{p_1-p_0,n-p-1}$.

**Intervalo de confiança**

Podemos isolar $\beta_j$ em (2.10) para obter um intervalo de confiança de $1 - 2\alpha$ para $\beta_j$.

$$I_{\beta_j,1-2\alpha} = \left(\hat{\beta}_j - z_{1-\alpha} v_j^{\frac{1}{2}} \hat{\sigma}, \hat{\beta}_j + z_{1-\alpha} v_j^{\frac{1}{2}} \hat{\sigma}\right), \tag{2.14}$$

onde $z_{1-\alpha}$ é o $1 - \alpha$ percentil da distribuição normal.

## 2.3 Método KNN

O método dos *K vizinhos mais próximos* (KNN, do inglês $K$ Nearest Neighbor) é um método não paramétrico de aprendizado, baseado em memória, ou seja, utiliza diretamente os dados de treino, e não precisa de modelo para ser ajustado. Do ponto de vista computacional, isso pode ser custoso, já que se a base for muito grande, exigirá espaço de armazenamento das entradas.

Dado um ponto $x_0$, o método KNN usa as observações nos dados de treino que estão mais próximas de $x_0$ no espaço das preditoras para produzir uma predição para $x_0$ na variável resposta (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Ele pode ser tanto usado para problemas de classificação, quanto para problemas de regressão, porém o princípio nos dois casos é o mesmo. É um método simples e que possui a vantagem de não fazer suposições sobre os dados para sua implementação. Possui

algumas vantagens e desvantagens como outros métodos, e veremos nas seções seguintes. Antes disso, precisamos definir alguns termos e valores usados.

Segundo Hastie, Tibshirani e Friedman (2009), mesmo com sua simplicidade, esse método tem gerado resultados satisfatórios em diversos problemas de classificação, como por exemplo reconhecimento de dígitos escritos à mão, imagens de satélite e padrões em gráficos de eletrocardiograma e onde as formas das bordas de decisão são muito irregulares.

Dados um inteiro positivo $K$ e uma observação $x_0$, o classificador KNN primeiro identifica os $K$ pontos nos dados de treino que são mais próximo de $x_0$, representado por $N_k(x_0)$. Então é estimada a probabilidade condicional para a classe $j$ como a fração dos pontos em $N_k(x_0)$ que tem como resposta igual a $j$:

$$Pr(Y = j \mid X = x_0) = \frac{1}{K} \sum_{i \in N_k(x_0)} I(y_i = j). \tag{2.15}$$

Por fim, o método aplica a regra de Bayes e classifica a observação de teste $x_0$ para a classe com a maior probabilidade.

Observe que por "mais próximos" envolve uma métrica, e são medidas diferentes dependendo se a variável é categórica ou numérica.

Para variáveis numéricas, uma escolha possível é a distância euclidiana. Assim, a distância entre duas observações é medida em cada coordenada dada por cada preditora numérica.

No caso de variáveis categóricas nominais, duas observações são "próximas" quando pertencem a um mesmo nível da variável. Nesse caso recebem distância 0, caso contrário rebem distância 1. Nas variáveis ordinais pode ser usado um sistema de distância de ranking. Ranking mais distantes implicam distâncias maiores, por exemplo. A distância final é dada pela soma das distâncias em todas as variáveis preditoras. Outras medidas podem ser usadas. Para mais detalhes, ver (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

No caso de regressão, encontra-se os $K$ pontos mais próximos de $x_0$ no espaço das preditoras e calcula-se a média de suas respostas. Em símbolos,

$$\hat{Y}(x_0) = \frac{1}{K} \sum_{x_i \in N_k(x_0)} y_i, \tag{2.16}$$

onde $N_k(x)$ é a vizinhança mais próxima de $x_0$ definida pelos $K$ pontos $x_i$ na amostra de treino.

**Taxa de erro e erro de teste**

Quando a variável resposta $Y$ é qualitativa, uma maneira de quantificar a acurácia das estimativas do modelo $\hat{f}$ é pela *taxa de erro*, dada por

$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i), \tag{2.17}$$

onde $\hat{y}_i$ é a classe predita para a i-ésima observação usando $\hat{f}$, e $I(y_i \neq \hat{y}_i)$ é uma *variável indicadora* que é igual a 1 se se $\hat{y}_i \neq y_i$ e igual a 0 se $\hat{y}_i = y_i$. Então, essa soma resulta no número de observações classificadas erradas pelo modelo, e dividindo por $n$ temos a fração de classificações incorretas do modelo.

Como já foi discutindo anteriormente, o objetivo de um bom classificador é minimizar o erro de teste.

**Considerações sobre o método**

A escolha do valor de $K$, a quantidade de observações mais próximas, tem muitos efeitos sobre o classificador obtido. Conforme o valor de $K$ aumenta o método se torna menos flexível e produz uma fronteira de decisão próxima do linear. Também podemos perceber que o erro de treino será 0 para $K$ igual a 1.

Assim como na regressão linear, não há uma relação forte entre taxa de erro de teste e taxa de erro de treino em KNN.

Em ambos casos de regressão ou classificação, a escolha do valor de $K$, o nível de flexibilidade adequado, é crucial para o sucesso de qualquer método de aprendizagem.

Observe que em regressão não podemos usar o critério de *soma dos quadrados dos erros* nos dados de treino para escolha do melhor valor de K, pois nesse caso seria sempre escolhido $K = 1$.

Quando as preditoras estão em escalas muito diferentes, podemos padronizar as variáveis para terem média zero e desvio padrão igual a um para evitar problemas no cálculo das distâncias euclidianas. Por exemplo preditoras em escalas muito maiores podem mascarar a participação de outras preditoras que podem ser relevantes porém estão em escala muito menor.

O método KNN enfrenta problemas diante de casos com diversas preditoras, e com um número insuficiente de observações por preditora, é o problema da *alta dimensionali-*

Tabela 1: Frequência para $N$ distribuições binomiais

| | Subgrupos | | | |
|---|---|---|---|---|
| | 1 | 2 | $\cdots$ | $N$ |
| Sucessos | $Y_1$ | $Y_2$ | $\cdots$ | $Y_N$ |
| Fracassos | $n_1 - Y_1$ | $n_2 - Y_2$ | $\cdots$ | $n_N - Y_N$ |
| Totais | $n_1$ | $n_2$ | $\cdots$ | $n_N$ |

*dade*. Nesse caso, os pontos mais próximos de uma observação não estão necessariamente "próximos", e isso pode resultar em erros grandes (BELLMAN, 2015 apud HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

## 2.4 Regressão logística

O modelo de regressão logística surgiu do desejo de modelar probabilidades posteriores de $K$ classes por uma função linear em $x$, enquanto ao mesmo tempo garantir que a soma delas seja 1 e cada uma esteja no intervalo $[0, 1]$. Ao invés de modelar a resposta de $Y$ diretamente, a regressão logística modela a *probabilidade* de $Y$ pertencer a uma particular categoria (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Na tabela 1, sendo $Y_i$ o número de sucessos numa subgrupo de tamanho $n_i$, para $i = i, ..., N$ subgrupos, a ideia é descrever a proporção de sucesso, $P_i = Y_i/n_i$ em cada subgrupo da variável resposta em função dos níveis dos fatores e outras varáveis preditoras que caracterizam os subgrupos. Nesse caso, $E(P_i) = \pi$, onde $\pi$ é a probabilidade de sucesso, e modelamos essas probabilidades por

$$g(\pi) = x_i^T \beta,$$

onde $x$ é um vetor de variáveis explicativas de dimensão $p + 1$, que são $p$ variáveis considerando as variáveis dummies para as categóricas e medidas das covariáveis, mais um para o parâmetro de escala. $\beta = (\beta_0, \beta_1, \cdots, \beta_p)$ é um vetor de parâmetros e $g$ é a função de ligação (ANNETTE; DOBSON; BARNETT, 2008). Para exemplificar, se temos duas variáveis explicativas numéricas e uma categórica com três níveis, teremos $p + 1 = 5$ ou seja, $p = 4$, pois são duas para as variáveis numéricas e duas para a variável categórica, já que para três níveis teremos duas variáveis *dummies*, e mais uma entrada com o valor 1 para o termo de escala ($\beta_0$) entrar no produto de matrizes, que corresponde ao valor assumido pela função quando todas as variáveis preditoras assumem o valor zero. Além

disso, $\beta = \{\beta_0, \beta_1, \beta_2, \beta_3, \beta_4\}$.

O caso mais simples, é usar a função de ligação identidade, que resulta num modelo de regressão linear $\pi = x^T\beta$ para modelar a probabilidade de sucesso. Usar essa função de ligação pode levar a predições menores que zero ou maiores que um, não correspondendo a uma probabilidade.

Discutiremos o caso para $K = 2$, que simplifica o algoritmo significativamente e, além disso, é muito usado em aplicações bioestatísticas onde respostas binárias ocorrem com frequência, por exemplo se o paciente sobreviveu ou não, se possui a doença ou não, uma característica presente ou não, entre outras (ANNETTE; DOBSON; BARNETT, 2008). Em nosso caso, pode ser usado para uma resposta binária como inadimplência ou adimplência de um cliente.

Para garantir que $\pi$ esteja no intervalo $[0, 1]$ ele é frequentemente modelado por uma distribuição acumulada, que dentre outras possibilidades, temos o *modelo logístico* ou *logit*, que nos gera o modelo

$$\pi = \frac{e^{x^T\beta}}{1 + e^{x^T\beta}} \tag{2.18}$$

A partir dessa equação, temos que

$$\frac{\pi}{1-\pi} = \frac{\frac{e^{x^T\beta}}{1+e^{x^T\beta}}}{1 - \frac{e^{x^T\beta}}{1+e^{x^T\beta}}} = \frac{e^{x^T\beta}}{1 + e^{x^T\beta}} \cdot \frac{1 + e^{x^T\beta}}{1} = e^{x^T\beta}$$

Esse resultado é chamado de *odds* e pode assumir valores de 0 a $\infty$. Tomando o logaritmo em ambos os lados da equação, obtemos a função de ligação

$$\log\left(\frac{\pi}{1-\pi}\right) = x^T\beta, \tag{2.19}$$

onde $\beta^T = (\beta_0, \beta_1, \cdots \beta_p)$, e $\beta_0$ é o logaritmo da chance quando $x$ assume valor 0 e $\beta_j$ é o quanto varia o logaritmo da chance ao se aumentar em uma unidade o valor de $x_j$ mantendo constante as outras variáveis, para $j = 1, \cdots, p$.

O termo $log[\pi/(1-\pi)]$ também é chamado de **função logística** e tem a interpretação natural de ser o logaritmo da chance (ANNETTE; DOBSON; BARNETT, 2008).

Usualmente, modelos de regressão logística são ajustado por meio do método de

*máxima verossimilhança*, usando a verossimilhança condicional de $G$ dado $X$, onde $G$ é uma das classes (Grupos) da variável resposta. Como $P(G \mid X)$ especifica completamente a distribuição condicional, uma distribuição *multinomial* é adequada. A log-verossimilhança para $N$ observações nesse caso é

$$l(\theta) = \sum_{i=1}^{N} \log p_{g_i}(x_i; \theta) \tag{2.20}$$

onde $p_k(x_i; \theta) = P(G = k \mid X = x_i; \theta)$.

No caso de duas classes ($K = 2$) é conveniente codificar as duas classes $g_i$ por 0/1 nas respostas, onde $y_i = 1$ quando $g_i = 1$ e $y_i = 0$ quando $g_i = 2$. Também podemos fazer, nesse caso, $p_1(x; \theta) = p(x; \theta)$, e $p_2(x; \theta) = 1 - p(x; \theta)$. A log-verossimilhança pode ser escrita como

$$\begin{aligned} l(\beta) &= \textstyle\sum_{i=1}^{N} \{y_i \log p(x_i; \beta) + (1 - y_i) \log(1 - p(x_i; \beta))\} = \\ &= \textstyle\sum_{i=1}^{N} y_i \beta^T x_i - \log(1 - p(x_i; \beta)), \end{aligned} \tag{2.21}$$

onde $\beta = (\beta_0, \beta_1, \cdots, \beta_p)^T$ e assumimos que o vetor $x_i$ inclui o termo constante 1 para considerar o intercepto.

Para maximizar a log-verossimilhança, fazemos as derivadas no ponto zero. As equações de *score* são

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{N} x_i (y_i - p(x_i; \beta)) = 0, \tag{2.22}$$

a qual possui $p + 1$ equações não lineares em $\beta = (\beta_0, \cdots, \beta_p)$. Note que, como a primeira componente do vetor $x_i$ é 1, a primeira equação score especifica que $\sum_{i=1}^{N} y_i = \sum_{i=1}^{N} p(x_i; \beta)$, isto é, o número esperado na classe 1 corresponde ao número observado, e consequentemente, também o número esperado na classe 2, tomada como 0.

Para resolver as equações scores em (2.22), usamos algum método numérico iterativo, como por exemplo o algoritmo de Newton-Raphson que requer a segunda derivada (ou matriz Hessiana). Os detalhes do algoritmo aplicado podem ser encontrados em Hastie, Tibshirani e Friedman (2009, p. 120).

Os modelos de regressão logística são usados mais como uma ferramenta de análise de dados e inferência, onde o objetivo é entender a relação entre as variáveis explicativas

e a variável resposta (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Por exemplo, as estimativas obtidas dos parâmetros $\beta_i$'s podem ser interpretadas como sendo o quanto varia o *logaritmo da chance* de sucesso da variável resposta ao aumentar em uma unidade a variável preditora correspondente $X_i$, ou da mesma forma, ao aumentar uma unidade em $X_i$ a chance fica multiplicada por $e^{\beta_i}$. Isso é chamado de *razão de chances*, obtido fixando todas as variáveis preditoras, e aumentando em uma unidade a variável de interesse.

**Qualidade do ajuste**

Uma maneira de avaliar a qualidade de um ajuste com regressão logística é Definir a *deviance* que tem a forma $D = 2\sum o \log \frac{o}{e}$ onde $o$ são as frequências observadas e $e$ as frequências esperadas estimadas em cada amostra subgrupo dos dados na tabela 1.

Note que D não envolve nenhum parâmetro. Então a qualidade do ajuste pode ser medida e testes de hipóteses podem ser feitos diretamente usando a aproximação $D \sim \chi^2(N-p)$, onde $N$ é o número de grupos nos dados, e $p$ o número de parâmetros estimados (ANNETTE; DOBSON; BARNETT, 2008).

O método de estimação e distribuições amostrais usados para inferência dependem de resultados assintóticos, por isso pode ter aproximações ruins para amostras pequenas.

## 2.5 Análise de Discriminante Linear

A regressão logística modela $P(Y = k|X = x)$ diretamente. Uma alternativa é a Análise de Discriminante Linear (LDA, do inglês *Linear Discriminant Analysis*), que modela a distribuição das preditoras $X$ em cada categoria da variável resposta $Y$ e depois usa o teorema de Bayes para contorná-los em estimativas para $P(Y = k|X = x)$.

Algumas das vantagens dessa abordagem em relação a regressão logística é que ela é mais estável quando as classes da resposta são bem separadas, quando a amostra é pequena e provém de uma distribuição aproximadamente normal e, por fim, é um método popular quando há mais de duas classes na variável resposta.

Seja $\pi_k$ a probabilidade a *priori* de uma observação pertencer a $k$-ésima categoria e $f_k(x) \equiv P(X = x|Y = k)$ a função de densidade de $X$ para uma observação da $k$-ésima categoria e $p_k(x) \equiv P(Y = k|X)$. Então, o teorema de Bayes afirma que

$$p_k(x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

Assim, faz-se necessário propor valores para a quantidade $\pi_k$, e para a distribuição de $f_k(x)$ . Com isso, iremos então classificar a observação $x$ na categoria em que $p_k(x)$ é maior.

Por exemplo, assumindo que $X = (X_1, X_2, ..., X_p)$ segue uma distribuição normal multivariada, escrevemos $X \sim N(\mu, \Sigma)$, onde $E(X) = \mu = (\mu_1, ..., \mu_p)$ e $Cov(X) = \Sigma$ é a matriz $p \times p$ de variâncias e covariâncias das preditoras. Nesse caso, a função de densidade é dada por

$$f(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

No caso de múltiplas preditoras, o classificador LDA assume que a observação da $k$-ésima categoria foi retirada de uma distribuição normal multivariada $N(\mu_k, \Sigma)$, onde $\mu_k$ é o vetor de médias específico da classe e $\Sigma$ é a matriz de covariâncias comum a todas as $K$ classes.

Alguns dos estimadores usado para as quantidades $\pi_k$, $\mu$ e $\Sigma$ são

$$\hat{\pi}_k = \frac{n_k}{n},$$

$$\hat{\mu}_k = \frac{1}{n_k}\sum_{i:y_i=k} x_i$$

e

$$\hat{\sigma}^2 = \frac{1}{n-K}\sum_{k=1}^{K}\sum_{i:y_i=k}(x_i - \hat{\mu}_k)^2$$

Com a hipótese de normalidade multivariada das preditoras $X$ e um pouco de manipulação algébrica, é possível mostrar que o classificador de Bayes prediz a observação $X = x$ para a categoria na qual a quantidade

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

é maior. Essa é o discriminante da função e recebe o nome linear devido ao termo linear em $x$.

## 2.6 Análise de Discriminante Quadrático

A diferença para o LDA é que em Análise de discrimante quadrático (QDA, do inglês *Quadradtic Discriminant Analysis*) não se supõe variâncias iguais nas classes da variável resposta, ou seja, cada classe tem a sua própria matriz de variâncias e covariâncias $\boldsymbol{\Sigma_k}$, onde $k$ é a classe.

Com essa hipótese, o classificador de Bayes irá determinar uma observação $X = x$ para a classe que

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

é maior. Então, o QDA envolve estimativas para $\Sigma_k$, $\mu_k$ e $\pi_k$ para obter os valores $\delta_k(x)$ e assim determinar a classe de predição para uma observação. O valor $x$ aparece aqui como uma função quadrática, de onde vem o nome desse classificador.

**Regressão logística ou LDA?**

Embora as motivações sejam diferentes, os métodos de LDA e regressão logística são bem semelhantes. Os dois modelos, quando possuem uma preditora, são funções lineares de $x$, o que faz ambos produzirem uma borda de decisão linear. A principal diferença está nas estimações dos parâmetros, feita por máxima verossimilhança na regressão logística e por estimação de média e variância no LDA. Além disso, o LDA assume que as observações são retiradas de uma população com distribuição normal e mesma variância em cada categoria. Caso isso seja verdade, isso pode resultar em melhores ajustes para a LDA do que a regressão logística (JAMES et al., 2013).

Porém, esse nem sempre é o caso. Na prática, quando alguma das variáveis explicativas é categórica, isso já pode contradizer a hipótese da distribuição normal, e geralmente em problemas práticos, temos a presença de variáveis categóricas.

O artigo de Press e Wilson (1978) exemplifica vários argumentos contra o uso em geral de função de estimadores de LDA, como por exemplo se as variáveis explicativas não seguem uma distribuição normal os estimadores de funções discriminantes tendem à não mostrar sinais de problemas em alguns casos; a presença de variáveis explicativas binárias não garante boas estimativas pelo método de função discriminante; os estimadores de máxima verossimilhança forçam o número de casos observados ser igual ao número de casos esperados, o que é desejável; A regressão logística possui uma estatística suficiente

associada com ela, entre outros.

## 2.7 Métodos baseados em árvores

Os métodos de árvores de decisão são baseados em dividir o espaço das preditoras em regiões e depois ajustar um modelo simples, como uma constante a partir da média, para a predição de um valor nessa região. Teoricamente, as regiões podem possuir qualquer forma, porém regiões retangulares ou caixas de várias dimensões são mais simples de descrever (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

A representação gráfica lembra um pouco uma árvore invertida. Por isso seu nome e alguns termos também associados à arvores. Considere a partição binária, ou seja, o domínio de cada variável selecionada é dividida em duas partes. Por exemplo, no caso de uma variável numérica é algo como $x \leq a$ ou $x > a$, e no caso de uma variável categórica com três níveis $A, B$ e $C$ é algo como $x \in \{A, B\}$ ou $x \in \{C\}$. Então, esse ponto da árvore é chamado de *nó*, onde se faz um teste de "sim" ou "não" para decidir a qual região da variável preditora pertence a observação. Começamos no topo da árvore, onde temos o primeiro nó, ou nó raiz, que é o primeiro teste na variável preditora selecionada. De acordo com a resposta (sim ou não), é escolhido o caminho a seguir contendo outros nós ou o nó terminal. Essa última é a *folha* da árvore. Um conjunto desses testes é chamado de *Ramo*. Um ramo é uma *subárvore* da árvore maior. Cada nó gera dois ramos. Partindo do nó raiz, quando não há mais testes a fazer nas variáveis preditoras selecionadas, chega-se a resposta ou decisão final para a observação. A *profundidade* de uma árvore é determinada pelo maior número de nós que se passa a partir do nó raiz, até chegar ao nó final. Se uma árvore só possui um nó, ela tem profundidade 0. Se uma árvore tem profundidade 1 também é chamada de *toco* (em ingles, *stump*).

Também temos o termo *poda* em árvores, que é para indicar que a árvore podada é obtida da árvore original, geralmente com muitos nós, e então faz-se a árvore sobre os mesmos ramos da primeira porém com menos nós. Observe que não necessariamente a árvore podada será menos profunda.

Métodos baseados em árvores de decisão são simples, e fáceis de interpretar, pois de certa forma imita a maneira de se tomar algumas decisões baseadas em testes simples de "se, então". Porém, veremos mais a frente que há métodos que perdem a interpretabilidade em troca de ganho em predição.

É um método diferente das técnicas clássicas de regressão em que a relação entre a

variável resposta e a preditora é especificada a priori e posteriormente é realizado um teste para provar ou refutar essa relação. O método baseado em árvores não assume essa relação. Ao contrário disso, constroem-se um conjunto de regras nas preditoras, categorizando-as, obtendo grupos menores. Em seguida, é feito um teste exaustivo para selecionar a melhor partição da preditora, segundo algum critério com relação à variável resposta, como por exemplo no caso de uma árvore de regressão, um critério selecionado é a divisão que maximiza a homogeneidade dos dois grupos resultantes da divisão em relação à variável resposta (PRASAD; IVERSON; LIAW, 2006).

Além disso, esse método pode ser usado tanto para problemas de regressão quanto para problemas de classificação, porém, os resultados para problemas de classificação em geral são melhores. Isso se deve ao fato de que foi um método desenvolvido originalmente para problemas de classificação, que diferente de regressão, possui um número finito de resultados possíveis.

Descreveremos a seguir brevemente três métodos que utilizam árvores, que são *bagging, random forests* e *gradient boosting*. Cada um desses métodos envolve produzir muitas árvores que depois são combinadas para produzir uma predição (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Porém, é importante observar que a estrutura de árvore é perdida, ou seja, perde-se em interpretabilidade com esses métodos em troca de melhores predições. O modelo final não é uma árvore de decisão, mas uma combinação de várias árvores de decisão, o que impossibilita sua representação gráfica em uma única árvore.

Antes disso, é apresentado um exemplo de árvore de classificação e um de regressão com as respectivas medidas de qualidade adotadas para avaliar a qualidade do ajuste de uma árvore.

### Árvore de classificação

Na árvore de classificação predizemos pela categoria mais frequente na mesma região nas observações de treino. Geralmente, é interessante saber as proporções de cada categoria nas regiões a fim de saber se a região discrimina bem as categorias.

O método para construção da árvore de classificação é o mesmo para a árvore de regressão, por meio de cortes binários recursivos, ou seja, uma variável é selecionada para o nó segundo algum critério, e então é escolhido o melhor ponto de corte dessa variável criando duas regiões dentre as realizações (numéricas ou categóricas). O melhor ponto de corte é descrito mais adiante.

- Taxa de erro de classificação: Uma predição possível de uma observação numa região é naquela categoria com maior frequência nessa região, ou seja, pela *moda*. A taxa de erro de classificação é simplesmente a fração das observações de treino daquela região que não pertencem à categoria mais comum. Sendo $\hat{p}_{mk}$ a proporção das observações de treino na $m$-ésima região que são da $k$-ésima categoria, a taxa de erro de classificação $E$ é dada por

$$E = 1 - \max_{k}(\hat{p}_{mk}). \tag{2.23}$$

  Essa taxa de erro não é suficientemente sensível para para o "crescimento" da árvore. Na prática, outras duas medidas são preferíveis.

- índice de Gini: É uma medida da variância total em todas as categorias da variável resposta dada por

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}) = 1 - \sum_{k=1}^{K} \hat{p}_{mk}^2. \tag{2.24}$$

  Dada a característica da expressão do índice de Gini, ela assume valores pequenos se todos os $\hat{p}_{mk}$'s são próximos de zero ou um. Por isso esse índice é conhecido como uma medida da pureza de um nó. Quanto mais puro um nó, mais próximo de zero ou de um é $G$ e o nó tem mais observações de uma mesma categoria da variável resposta.

- entropia: É uma alternativa ao índice de Gini. É dado por

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log(\hat{p}_{mk}) \tag{2.25}$$

  Como $0 \leq \hat{p}_{mk} \leq 1$ e $\log(\hat{p}_{mk}) < 0$, então $0 \leq -\hat{p}_{mk} \log(\hat{p}_{mk})$. É possível mostrar que a entropia assume valores próximos de zero se os $\hat{p}_{mk}$'s são todos próximos de zero ou próximos de um. Como o índice de Gini, a entropia assume valores pequenos se o $m$-ésimo nó é puro.

Durante a construção de uma árvore de classificação, geralmente é usado ou o índice de Gini ou a entropia para avaliar a qualidade de um "corte" em particular na variável preditora. Qualquer uma das três medidas acima pode ser usada durante a poda de uma árvore, porém para medir a acurácia de uma predição na árvore final podada, é preferível

usar a taxa de erro de classificação (HASTIE; TIBSHIRANI; FRIEDMAN, 2009), já que nessa etapa é apenas avaliação do modelo e não serão feitos mais ajustes nos nós e o índice de Gini não é uma medida de erro.

Uma maneira de construir uma árvore de decisão aproveitando as duas medidas é gerar uma árvore preliminar usando o índice de Gini, com um número de nós (e assim regiões) limitado. Assim tenta-se separar o espaço das preditoras em regiões mais homogêneas em relação à variável resposta, já que usa-se um índice de pureza. No segundo momento refazer o ajuste da localização dos nós e das classificações desta árvore fixando as regiões obtidas preliminarmente, mas agora usando o erro de classificação, pois agora, com as regiões criadas com nós puros, buscaremos nas preditoras desses nós os pontos de corte que se aproximam mais dos valores reais, ou seja, com menos erro.

Agora vamos considerar dois casos em relação as variáveis preditoras, dependendo do tipo da variável, categórica ou numérica.

Quando a preditora é categórica, o procedimento consiste em separar a variável em dois grupos, separado em cada nível da variável, e medir o índice de Gini para cada separação, nos dados de treino. Por exemplo, Se uma variável categórica $X$ possui os níveis $A, B, C$, então o procedimento é fazer três medições do índice de Gini da seguinte forma: primeiro faz-se um nó separando em "se a observação é da categoria $A$ de um lado e $B$ ou $C$ do outro", e mede-se o índice de Gini desse nó. A seguir faz-se a separação "se é da categoria $B$ de um lado e $A$ ou $C$ do outro", e faz-se a medição do índice de Gini desse nó, e por fim a separação em "se é da categoria $C$ de um lado e $A$ ou $B$ do outro". A melhor separação (ou corte) é escolhido para o nó que obtiver menor índice de Gini.

Quando a preditora é numérica, o procedimento é semelhante, porém o que se faz é colocar as observações da variável numérica em ordem crescente e obter a separação pela média aritmética de duas observações consecutivas. Ou seja, categoriza-se a variável numérica em dois grupos. O melhor corte será o que obtiver menor índice de Gini. Perceba que quando maior o número de observações diferentes da variável, maior será o número de testes a serem feitos e assim maior custo computacional.

Abaixo temos um exemplo de árvore de classificação aplicado aos dados IRIS do *datasets* do $R$, para classificar uma folha como da espécie setosa, versicolor ou virginica, segundo o comprimento e largura da pétala. Foi usado o software $R$, com o pacote *rpart* (THERNEAU; ATKINSON, 2019).

Na árvore de decisão na figura 2, temos algumas informações que são nome da espécie,

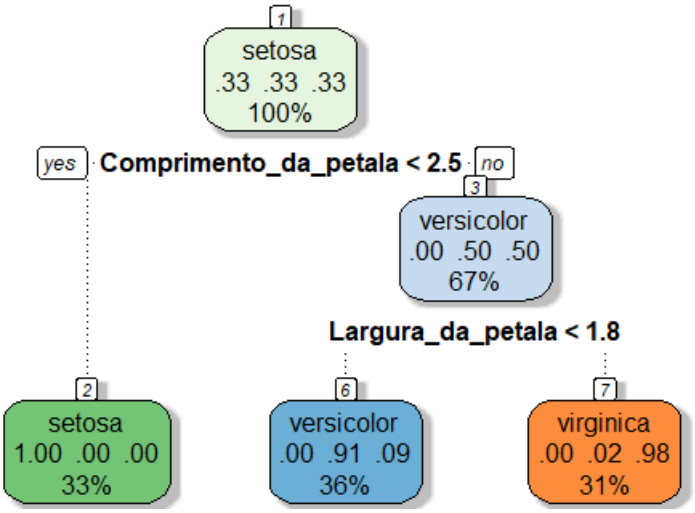


Figura 2: Árvore de decisão para IRIS

três proporções e a porcentagem dos dados incluídos nesse nó. No nó raiz e no nó interno, a divisão fica visível. Na raiz, as proporções são de $0,33$ para cada espécie e 100% dos dados estão na raiz. Então nesses dados temos a mesma quantidade de cada espécie e todos os dados estão na raiz.

A divisão ("comprimento da pétala" $< 2,5$) coloca 33% dos dados na folha *setosa* e 67% no nó interno da direita. A folha *setosa* mostra as proporções de $1,00$, $0,00$ e $0,00$ indicando que todos os casos nessa folha são corretamente classificados como *setosas*.

O nó interno mostra $0,00$, $0,50$ e $0,50$ o que significa que nenhum desses casos é de *setosas*, metade é *versicolor* e a outra metade é *virginica*. Essa divisão do nó interno ("largura da pétala" $< 1,8$") coloca 36% dos casos na folha *versicolor* e 31% dos casos na folha *virginica*. Isso já mostra um problema: com a classificação perfeita, esses percentuais seriam iguais, porque cada espécie aparece igualmente nos dados.

Na folha *versicolor*, as proporções são de $0,00$, $0,02$ e $0,98$. Portanto, 2% dos casos classificados como *virginica* são na verdade *versicolor*.

Em geral, para a grande maioria dos 150 casos nos dados, as regras de classificação funcionam na árvore de decisão, porém apresentam algum erro como geralmente acontece numa decisão.

### Árvore de regressão

Para a construção de uma árvore de regressão, resumidamente, há dois passos (HASTIE; TIBSHIRANI; FRIEDMAN, 2009):

1. Particiona-se o espaço das preditoras em $J$ regiões $R_1, R_2, \cdots, R_J$.

2. Para toda observação que pertença a mesma região $R_j$, fazemos a mesma predição, que é por exemplo simplesmente a média das observações de treino que pertencem à $R_j$.

Na figura 3 temos o gráfico de dispersão para as variáveis "Hits" - número de acertos do jogador na temporada de 1986 e "Years" - tempo do jogador na liga, para os dados HITTERS no pacote ISLR (JAMES et al., 2017) por meio do software R, que contém diversas informações sobre jogadores da liga principal de basebol nas temporadas de 1986 e 1987. Observa-se as três regiões encontradas para construção da árvore de decisão, com o objetivo de predizer o salário anual do jogador em milhares de dólares, em escala logarítmica.

Na figura 4, temos um exemplo de árvore de regressão ajustada pela função *rpart* (THERNEAU; ATKINSON, 2019), selecionadas duas preditoras *Years* e *Hits*. Para *Years*, o ponto de divisão (parâmetro) foi $4,5$ anos e para *Hits* foi $117,5$, porém no gráfico foi arredondado para 118. O número na parte superior da caixa é o salário predito para um jogador naquela região e abaixo é a porcentagem dos dados que estão naquela região. Observe que o nó raiz possui 100% dos dados, então o logaritmo do salário anual médio dos jogadores é de $5,93$. Se um jogador possui menos de $4,5$ anos na liga principal, seu salário predito pelo modelo, em escala logarítmica será de 5.11, independente do número de acertos. Se o jogador possui mais de quatro anos de liga principal, o modelo prediz um salário de $6,35$, porém é feita mais uma separação, segundo o número de acertos na temporada anterior. Se o jogador possui mais de $4,5$ anos de liga e obteve menos de 118 acertos (no modelo é menos de $117,5$, porém como Hits é número inteiro isso não faz diferença) então o modelo prediz um salário de $6,0$ e se obteve mais de 118 acertos é predito um salário de $6,74$ milhares de dólares anuais, em escala logarítmica.

Para construir as regiões $R_j$ do passo 1 acima, teoricamente, elas podem ter qualquer forma. Como dissemos antes, geralmente escolhe-se dividir as regiões em retângulos ou caixas de altas dimensões pela simplicidade e por ser mais fácil de interpretar, inclusive graficamente. Os pontos das divisões das preditoras para encontrar as caixas $R_1, \cdots, R_j$, é encontrado com o objetivo de minimizar o $SQR$ dado por

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2, \tag{2.26}$$

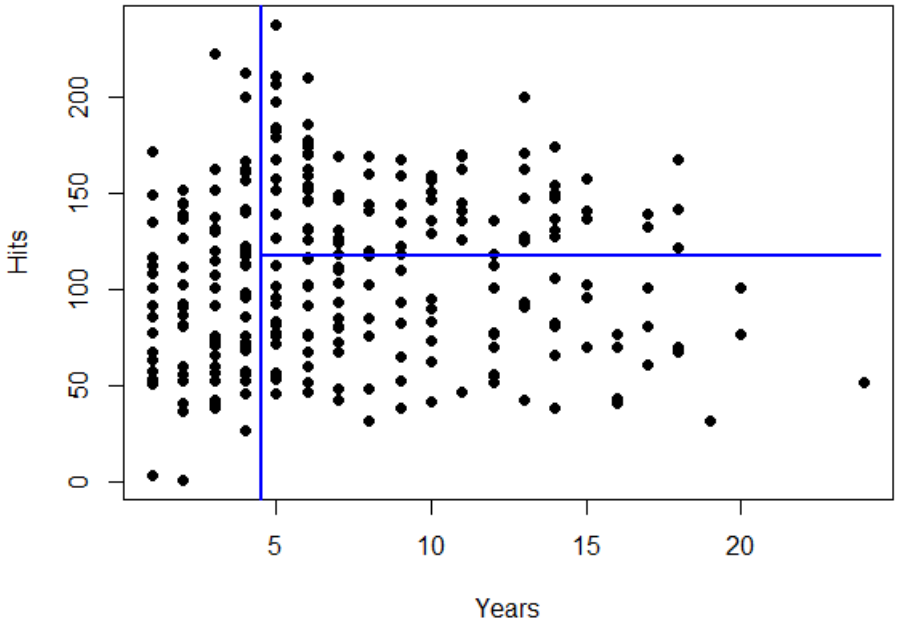


Figura 3: Gráfico de dispersão para HITTERS

onde $\hat{y}_{R_j}$ é a resposta média para as observações de treino na $j$-ésima caixa.

O método de árvore de decisão, em geral varia muito na predição, dependendo do dado de treino escolhido. Segundo James et al. (2013) um ponto negativo desse método é que pequenas mudanças nos dados podem mudar bastante o resultado final. A seguir temos algumas técnicas que buscam melhorar a predição obtida por uma única árvore de decisão.

### 2.7.1 Bagging

O método de bagging que utiliza várias árvores de decisão para gerar sua predição com menor variância do erro de teste ou de classificação. Assim, acaba sendo melhor do que uma única arvore ajustada, porém perde a interpretabilidade, que é uma das vantagens de uma árvore de decisão. Vejamos com mais detalhes como esse método funciona.

O método da árvore de decisão anterior sofre com alta variância, pois dependendo da amostra de treino selecionada, a predição para uma observação pode mudar significativamente. O método de *Bagging* (Bootstrap AGGregatING) (BREIMAN, 1996) é utilizado para diminuir a variância do erro de predição. A técnica consiste na geração de um número grande, $B$, de amostras de treino pelo método de bootstrap (amostras com reposição) e em

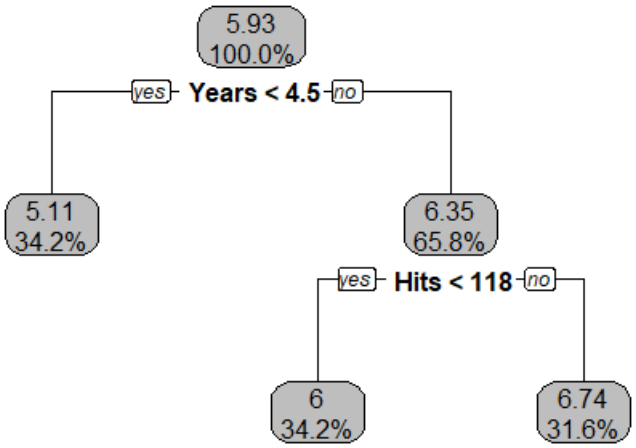


Figura 4: Árvore de decisão para HITTERS

cada uma dessas amostras é ajustado um modelo de árvore. Com isso, quando tentamos predizer o valor (ou classe, no caso de classificação) de uma nova observação, obteremos $B$ predições. No caso de regressão, o método calcula a média de todas essas $B$ predições. No caso de classificação, prediz a classe pela moda das $B$ predições.

Mais tecnicamente, no caso de regressão

$$\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x), \tag{2.27}$$

onde $\hat{f}^{*b}(x)$ é a predição para uma observação $x$ do modelo ajustado na $b^*$ amostra por bootstrap.

É interessante observar que cada árvore é construída profunda, sem poda, de maneira que possuem alta variância. Então quanto mais árvores construídas (maior o valor de $B$) mais valores diferentes podem ser preditos e assim a média dessas predições é um valor com menor variância do erro.

Note também que quanto maior o valor de $B$ menos variável fica o erro de predição, isso é devido ao fato de o método de bootstrap produzir um número limitado, mesmo que grande, de conjuntos de dados diferentes. Então aumentar o número de reamostragens

não gera novas ou melhores predições.

- Out-Of-Bag (OOB)

No processo de geração dos conjuntos de treino por bootstrap, algumas observações podem ficar de fora da amostra, devido à amostra ser com reposição. Mais tecnicamente, considere um conjunto de treino com $n$ observações, da qual serão retiradas $n$ amostras com reposição para gerar um conjunto por bootstrap. Considerando um espaço equiprovável, a probabilidade de uma particular observação $x_j$ fazer parte do conjunto de bootstrap é

$$1 - \left(1 - \frac{1}{n}\right)^n. \tag{2.28}$$

Quando $n$ tende para o infinito o limite dessa probabilidade é $(1 - e^{-1}) \approx 0,63$. As amostras que não fazem parte do conjunto de bootstrap são chamadas de amostras *Out Of Bag*. As amostras OOB podem ser usadas para produzir uma estimativa do erro de teste, pois são amostras que não participaram da construção da árvore de decisão. Como foram construídas $B$ árvores, cada observação fica de fora de, em média, $0,37B$ das árvores e assim usar essas observações para estimar o erro de teste é válido e é chamado de *erro OOB*. É uma estimativa conveniente quando se possui um banco de dados muito grande, para os quais computar o método de validação cruzada seria muito custoso computacionalmente. Isso se deve ao fato de que para fazer uma única estimativa pelo processo de bagging, o modelo irá testar essa observação OOB em cada uma das $B$ árvores. A função que prediz o valor de saída de qualquer observação é inviável para grandes amostras, pois quanto mais preditoras mais valores possíveis podem ser gerados de resultado. Por exemplo, pelo método de partição binária com $p$ preditoras, podem ser gerados até $2^n$ valores diferentes (o número máximo de conjuntos de dados diferentes criado pelo bootstrap) e o modelo calcula a predição para cada observação específica.

- Medidas de importância de variáveis

Há uma observação importante sobre esse método. Os conjuntos de treino acabam sendo correlacionados na maioria das vezes, isso acaba gerando valores semelhantes para predição em cada árvore o que não diminui significativamente o erro de teste. De outra maneira, podem selecionadas as mesmas preditoras nos primeiros nós, que são os mais "importantes". No caso de regressão o conceito de importância de uma variável,

superficialmente, é medido pelo quanto de erro ela diminui no SQR sendo selecionada para a partição em relação às outras, e no caso de árvores de classificação é pelo maior decrescimento no índice de Gini (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

### 2.7.2 Random Forest

O método de floresta aleatória é pensado de maneira a melhorar a diminuição na variância da predição do método bagging diminuindo a correlação entre as árvores. O método de bagging, como dito acima, acaba gerando árvores correlacionadas devido às amostras de treino serem correlacionadas e devido ao critério de seleção de variáveis, que considera a variáveis com maior pureza dentre todas, acabam selecionando as mesmas variáveis em muitas árvores. Fazer a média de muitas quantidades altamente correlacionadas não leva à uma grande redução na variância quanto fazer a média de muitas quantidades com baixa correlação. O método de seleção de amostras é o mesmo do bagging, o que muda é a maneira de escolher as preditoras para a separação dos nós (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Usamos o método de Bootstrap para gerar $B$ amostras de treino como no bagging e em cada conjuntos constrói-se uma árvore. Porém, a diferença para o bagging é que no momento de decidir as variáveis candidatas à separação do nó são selecionadas de forma aleatória apenas um subconjunto com $m$ das $p$ variáveis originais, $m \leq p$. Daí o nome do método. Geralmente, usa-se $m = \sqrt{p}$ em árvores de regressão e $m = p/3$ em árvores de classificação, porém isso é controlável (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Essa restrição na escolha aleatória das variáveis candidatas à separação de cada nó torna possível a construção de árvores diferentes e menos correlacionadas, pois em algumas árvores, as variáveis com maior importância dentre todas as $p$ variáveis nem farão parte das candidatas a separação do nó raiz. É importante observar que para $m = p$ isso é simplesmente o método de bagging, ou seja, bagging é um caso particular de random forest. Além disso, no caso de existirem muitas preditoras fortemente correlacionadas, o método de random forest funciona melhor com valores de $m$ menores (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

O método de floresta aleatória possui algumas vantagens como não sofrer sobre ajuste do modelo, lidar bem com dados em alta dimensão (muitas preditoras) e também alguns pontos fracos como não termos muito controle sobre o que o modelo faz (caixa preta) e ele funciona melhor em problemas de classificação do que problemas de regressão, por causa do número teoricamente infinito de possibilidades da variável resposta no caso contínuo,

o que não acontece em classificação.

### 2.7.3 Gradient Boosting

O método de *gradient boosting* também é um método que busca melhorar os resultados de um método baseado em árvores de decisão. Diferente do método de *bagging* e *random forest* que utilizam reamostragem como *bootstrap*, e constroem as árvores nas reamostragens separadamente umas das outras, o *gradient boosting* constrói árvores em sequência e sobre os mesmos dados de treino, porém modificado a cada árvore da sequência.

O método busca dar mais peso as amostras que foram classificadas erroneamente na árvore anterior, para "forçar" a próxima árvore a classificar melhor essa observação. As observações classificadas corretamente acabam recebendo menor peso.

O método possui três hiperparâmetros:

1. O número de árvores $B$ a serem construídas na sequência. Diferente dos métodos de *bagging* e *random forest*, é possível ocorrer o *sobreajuste* se $B$ é muito grande. É possível selecionar o número adequado com *validação cruzada*.

2. O parâmetro de *aprendizado* $\lambda$, ou regularização. Controla a velocidade com que o *gradient boosting* "aprende". Geralmente são usados valores pequenos, como $0,01$ ou $0,001$. Aqui há uma troca entre o valor de $\lambda$ e $B$.

3. o número $d$ de nós em cada árvore. Controla a complexidade do modelo. Geralmente, $d = 1$ funciona bem, que são os *tocos*. Nesse caso, o *gradient boosting* ajusta um modelo aditivo, já que acaba sendo a soma de uma variável por árvore.

Com isso, resumidamente o que o algoritmo faz é uma atualização nos resíduos de cada árvore. Cada árvore $f^b(x)$ construída na sequência é feita de maneira que os resíduos (erros) vão diminuindo a cada árvore e o modelo final é dado por

$$\hat{f}(x) = \sum_{b=1}^{B} \lambda f^b(x). \tag{2.29}$$

Os resíduos são atualizados a cada iteração da sequência, ou seja, a cada nova árvore, acumulando o que foi aprendido na árvore anterior, controlado pelo parâmetro $\lambda$. A cada árvore $b = 1, \cdots, B$, faz-se $r_i \leftarrow r_i - \lambda \hat{f}^b(x_i)$ (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

A ideia é que a cada árvore nova na sequência, o erro se aproxime de zero. A velocidade dessa aproximação é controlada pelo $\lambda$.

# 3 Resultados

Neste capítulo é apresentado os resultados dos modelos descritos anteriormente em um banco de dados real para fins de comparação de desempenho.

O banco de dados é de um banco alemão (ASUNCION; NEWMAN, 2007) com vinte características (variáveis) de seus clientes e uma variável resposta binária que indica se o cliente foi adimplente ou inadimplente. Portanto, trata-se de um problema de classificação e faremos uma aprendizagem supervisionada.

O banco de dados possui 1000 observações, não possui dados faltantes, e das 20 variáveis preditoras, 17 são do tipo categórica e 3 são numéricas. A variável resposta está com $70,0\%$ das observações como adimplente e $30,0\%$ como inadimplente. Mais detalhes sobre o banco de dados estão disponíveis no apêndice 1.

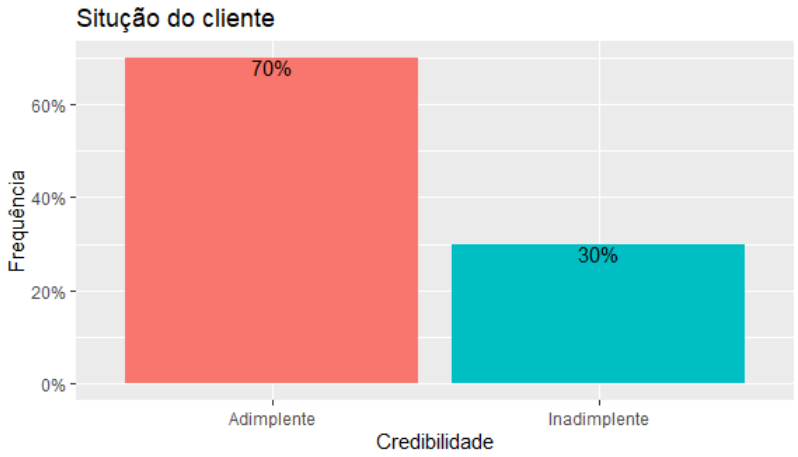


Figura 5: distribuição da credibilidade nos dados

A análise dos dados foi feita utilizando o programa *R*, versão 4.0.3. As funções utilizadas serão descritas ao longo do texto.

Para o tratamento dos dados, foi feita uma renomeação das variáveis para a língua portuguesa para facilitar a associação no contexto do problema. Além disso, inicialmente todas as variáveis estão definidas como numérica, foi então feita a categorização das

variáveis correspondentes desse tipo.

Para as variáveis "Numero_de_creditos_anteriores_neste_banco" e "Objetivo_do_credito" fizemos uma aglutinação de algumas categorias para obter um número maior de observações por categoria. Em Objetivo do crédito, juntamos as categorias 1- carro novo, e 2 - carro usado, na mesma categoria como 1, e também juntamos as categorias 4, 5, 6, 8, 9 e 10 na mesma categoria como 2. A categoria 7 não possui observações nessa base de dados. Em créditos anteriores, juntamos as categorias em 1 - um, e 2 - dois ou mais. Inicialmente estas estavam separadas em 1 - um, 2 - dois ou três, 3 - quatro ou cinco, 4 - seis ou mais.

A duração do crédito em meses foi mantida como numérica, pois possui 33 valores diferentes.

Todos os modelos foram obtidos pela função *train* do pacote *caret* (KUHN, 2020), exceto o modelo para o método de bagging e random forest, que foram obtidos pela função *randomForest* do pacote *randomForest* (LIAW; WIENER, 2002). Além disso, consideramos todas as vinte variáveis restantes do tratamento como preditoras.

O procedimento foi obter as métricas *Sensibilidade, Especificidade, Acurácia balanceada e Acurácia*, onde a acurácia balanceada, que no contexto do problema são:

A *sensibilidade* é a proporção de inadimplentes preditos corretamente pelo modelo, no grupo dos Inadimplentes reais.

A*especificidade* é a proporção de adimplentes preditos corretamente pelo modelo, no grupo de Adimplentes.

A *acurácia balanceada* é a média aritmética entre a especificidade e a sensibilidade.

A *acurácia* é a proporção de observações preditas corretamente dentre todas as observações, tanto para adimplentes quanto para inadimplentes.

O objetivo considerado aqui é maximizar a acurácia obtida pelos métodos, ou seja, maximizar a taxa de acerto geral, tanto para adimplentes quanto inadimplentes.

Foram feitas cento e cinquenta separações diferentes nos dados em treino (75%) e teste (25%) usando a função *sample* do pacote *DescTools* (AL., 2020) e fixadas uma semente para cada separação. Esta função foi usada para selecionar aleatoriamente 750 números de 1 a 1000, resultando em um vetor indicador das linhas a serem selecionadas para fazerem parte da amostra de treino e os números não selecionados são as linhas que farão parte da amostra de teste. Observe que dessa maneira, não necessariamente a proporção de

adimplentes e inadimplentes será igual ao da base toda de 70% e 30% respectivamente.

No pré-processamento dos dados de treino, foi feita uma avaliação onde constatou-se como dito anteriormente que não há presença de dados faltantes, não sendo necessário um tratamento com eliminação ou imputação de dados faltantes. Além disso, utilizando a função *nearZeroVar* do pacote *caret* (KUHN, 2020), foram retiradas as variáveis "Garantias" e "Trabalhador_estrangeiro" por possuírem a razão entre o número de observações na categoria mais frequente e na segunda categoria mais frequente maior do que 10, que foi o critério utilizado. Isso significa que para cada 10 observações da categoria mais frequente é observado uma da segunda categoria mais frequente. Caso hajam mais de duas categorias, a partir da terceira mais frequente é mais escasso ainda obtê-la. Em particular, as variáveis "Garantias" possui três categorias e "Trabalhador_estrangeiro" duas categorias.

Em seguida, foram padronizadas as variáveis quantitativas no treino, pois as escalas diferem muito entre as variáveis *Idade* e *Duracao_do_credito_meses* e *Montante_de_credito*, o que pode prejudicar o desempenho de alguns métodos como o K vizinhos mais próximos (KNN) que usa distância euclidiana aqui para calcular as distâncias entre as observações e o regressão logística (LOG) em que os coeficientes podem sair viesados quando não padronizados. Para isso, foi usada a função *preProcess* também do pacote *caret* (KUHN, 2020). A padronização utilizada foi baseada nos dados de treino, subtraindo-se a média e dividindo o resultado pelo desvio padrão de cada variável.

O mesmo foi feito nos dados de teste, utilizando a padronização do treino, ou seja, foi utilizada a média e desvio padrão obtida nos dados de treino, pois assim é como se as observações de teste estivessem sob a hipótese de serem obtidas da mesma população das observações de treino. E também foram retiradas as variáveis "Garantias" e "Trabalhador_estrangeiro".

A seguir é descrito a seleção dos parâmetros e hiperparâmetros dos métodos.

Cinquenta das cento e cinquenta partições foram usadas para escolha do parâmetro $K$ a ser fixado no método de KNN. Foram testados todos os valores para $K$ de 1 a 50, em cada uma das 50 partições e então foi escolhida a média das acurácias de cada partição para representar a acurácia do respectivo valor para $K$. Foi selecionado $K = 16$ com a maior acurácia no teste. Podemos observar na figura 6 os resultados obtidos para a seleção do melhor valor de $K$, que maximiza a acurácia. Foram plotados os valores de *taxa de erro* (1 - acurácia) e $1/K$ nos eixos para melhor visualização do ponto de mínimo do erro de teste na acurácia. É possível observar no gráfico que o menor erro de teste não necessariamente corresponde ao menor erro no treino. E quando $K = 1$ com o modelo

mais flexível possível, o erro atinge o valor mínimo de 0 nos dados de treino, porém no teste não foi o menor valor. É possível observar a forma de "*U*" no erro de teste, e o decrescimento do erro no treino como dito anteriormente neste trabalho.

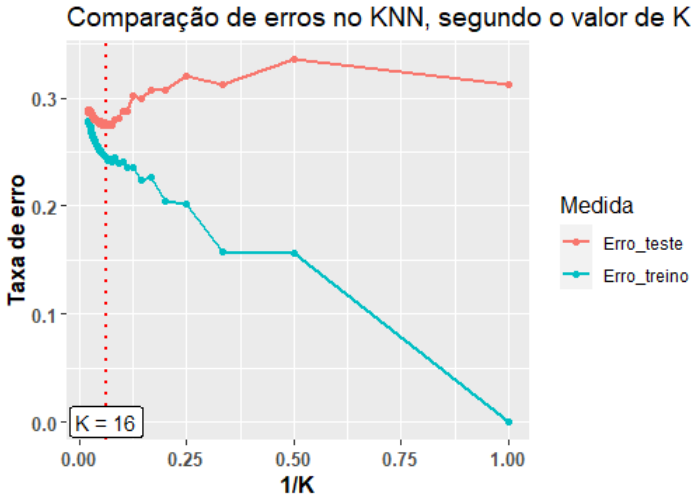


Figura 6: Erro de teste e treino para escolha do K

Para o método de *bagging* (BAG) foi fixado o número de árvores em 500 e o $mtry = 18$.

O $mtry = 4$ foi fixado no método de Floresta Aleatória (RF), que é o parâmetro que controla o número de variáveis candidatas em cada separação da árvore. Este valor foi escolhido por ser aproximadamente a raiz quadrada do número de variáveis preditoras, como recomendado por James et al. (2013). No caso, são 18 variáveis preditoras. É importante observar que este modelo não foi feito pelo *train* por terem sido contabilizadas por ele 43 variáveis preditoras, pois o *train* constrói variáveis dummies para as categóricas, o que a função *randomForest* não faz. E o número de árvores foi fixado em 500, assim como no bagging.

Para o método de *gradient boosting* (GBM - Gradient Boosting Method), os parâmetros foram fixados em $0,01$ para $\lambda$, o número mínimo de observações em cada folha em 10. O número de árvores foi escolhido nas cinquenta amostras de treino e teste também utilizada para escolha do $K$ no KNN. Foram testados os valores para o número de árvores em 10, 20, 30, 50, 100, 300, 500, 1000, 2000, 5000, 6000, 8000 e 15000. Para cada valor, foram obtidos o erro de treino e de teste de cada uma das 50 separações e por fim calculada a média obtida. Foi selecionado o número de árvores em 6000, como podemos observar na figura 7, com menor erro de teste apresentado. Note que o menor erro de treino não atingiu o valor 0 com o número de árvores em 15.000. Porém, para um número suficientemente grande de árvores, teremos um erro de treino em 0, que é a característica

do sobreajuste no número de árvores para o GBM. Caso a taxa de aprendizagem $\lambda$ seja maior, o sobreajuste tende a ocorrer mais rapidamente e vice-versa (JAMES et al., 2013).

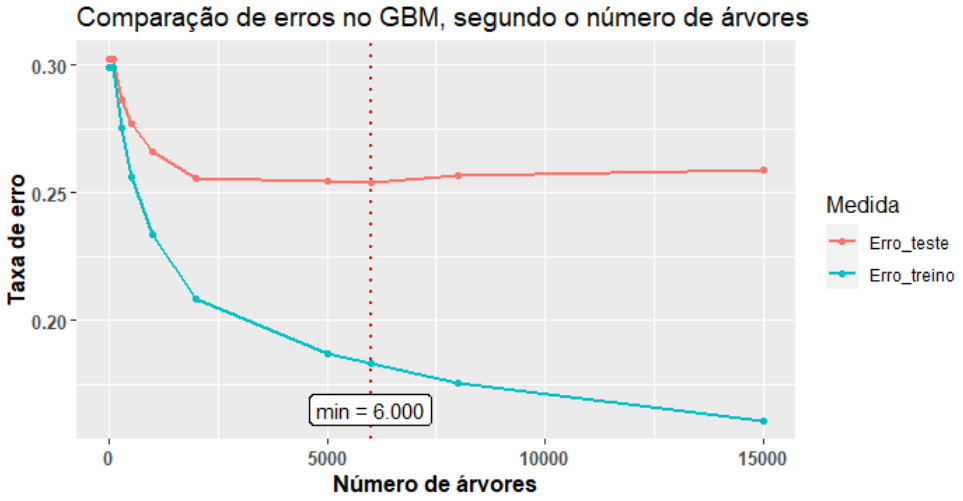


Figura 7: Erro de teste e treino para escolha das árvores no GBM

Os métodos de análise de discriminante linear (LDA), discriminante quadrático (QDA) e regressão logística (LOG) não possuem hiperparâmetros a serem fixados. Porém, é importante destacar que no caso da regressão logística que estima probabilidades, a classificação foi determinada pela probabilidade de inadimplência com ponto de corte em $0,50$. Observações com probabilidade de inadimplência acima de $0,50$ foram classificadas como inadimplentes e classificadas como adimplentes caso contrário.

É importante observar que o método de validação cruzada também pode ser utilizado para gerar uma estimativa do erro de teste. Neste trabalho, isso foi excluído, dado que fizemos o procedimento descrito.

Após fixados os parâmetros e hiperparâmetros, a comparação de desempenho foi feita através das 100 amostras de treino e teste não utilizadas para escolha dos hiperparâmetros. As 100 partições e resultados foram obtidos pelo procedimento a seguir.

(a) Fixada uma semente da aleatorização diferente das cinquenta anteriores, é feita a separação dos dados em treino (75%) e teste (25%) usando a função *sample* do pacote *DescTools* (AL., 2020). Observe que foi feita uma nova separação independente da separação anterior.

(b) Foram padronizadas as variáveis quantitativas nos novos treino, pois a padronização anterior foi feita separadamente. É preciso que essa segunda separação seja independente da primeira, com o objetivo de simular dados novos, descorrelacionados de

Tabela 2: Resultados da acurácia nas amostragens

| Método | Média | Desvio Padrão |
|---|---|---|
| RF | 0,756 | 0,027 |
| BAG | 0,756 | 0,024 |
| LDA | 0,749 | 0,025 |
| LOG | 0,749 | 0,026 |
| GBM | 0,744 | 0,025 |
| QDA | 0,730 | 0,026 |
| KNN | 0,721 | 0,027 |

qualquer observação utilizada anteriormente. O mesmo foi feito nos dados de teste, utilizando a padronização do novo treino, ou seja, foi utilizada a média e desvio padrão obtida nos dados do novo treino, pois assim é como se as observações de teste estivessem sob a hipótese de serem obtidas da mesma população das observações do novo treino. Também foram eliminadas as variáveis "Garantias" e "Trabalhador_estrangeiro", para manter a coerência com o feito anteriormente.

(c) Para cada iteração, fixamos a mesma semente aleatória para todos os métodos e então é treinado o modelo de cada método nos dados de treino.

(d) Fizemos a predição nos dados de teste pela função *predict* com o modelo treinado no item anterior e obtemos a matriz de confusão pela função *confusionMatrix* do pacote *caret* (KUHN, 2020), de onde salvamos as métricas desejadas.

(e) Fazemos de (a)-(d) novamente a cada iteração, mudando a semente da aleatorização.

O objetivo de mudar a partição a cada iteração é evitar uma possível influencia aleatória nos resultados devido a escolha da amostra de treino e teste.

Na tabela 2 temos os resultados obtidos da acurácia nas amostras e o gráfico de *boxplot* com os resultados obtidos nas 100 amostragens nos dados de teste.

A seguir, analisamos os resultados nas reamostragens por meio dos boxplots das métricas. Lembrando que os hiperparâmetros foram escolhidos maximizando a acurácia. Vejamos o comportamentos das outras métricas nos modelos nessa situação.

Na acurácia (figura 8), os resultados estão entre 65% e 75% em geral. Observa-se que o método KNN obteve os menores resultados em comparação com os outros métodos.

Nota-se que todos os métodos tiveram uma sensibilidade (9) relativamente baixa, entre 20% e 55% em média. Ou seja, no grupo dos inadimplentes, os métodos não acertaram

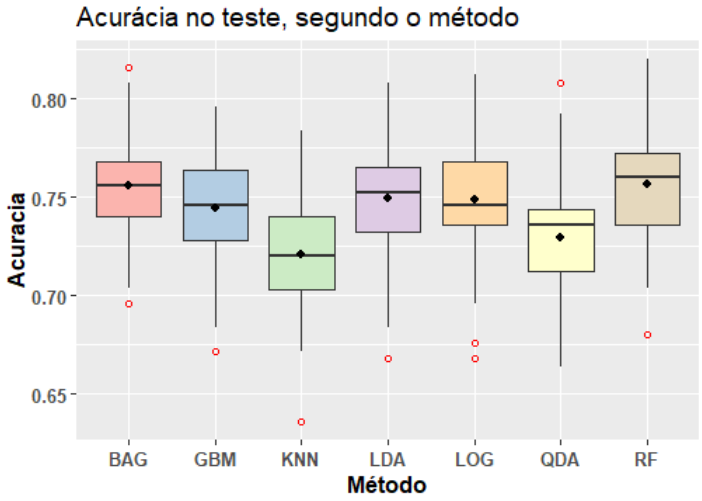


Figura 8: Comparação entre métodos pela acurácia

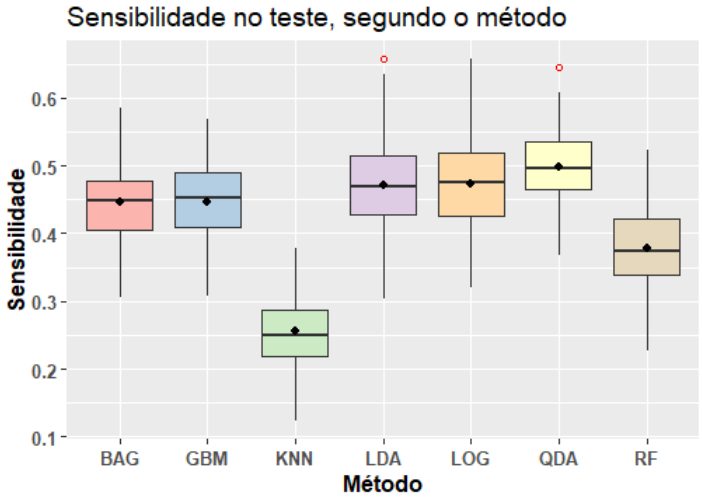


Figura 9: Comparação entre métodos pela sensibilidade

mais do que um sorteio aleatório os clientes que de fato são inadimplentes, com os modelos tendo sidos construídos visando maximizar a acurácia. Os métodos que se saíram relativamente melhor entre os outros foram os QDA, LOG e LDA, o que será constatado a seguir, com testes estatísticos.

Nota-se que em média todos os métodos obtiveram especificidade acima de 80%em geral, sendo o RF e o KNN com os melhores resultados e o KNN com resultado abaixo dos os outros métodos. Isso significa que no grupo dos adimplentes, os métodos tem resultados relativamente bons, acertaram a predição em mais de 80% das vezes.

Observa-se que a acurácia balanceada (11) em geral está entre 55% e 70% em todos os métodos, sendo os métodos de KNN e RF o com os menores resultados em geral.

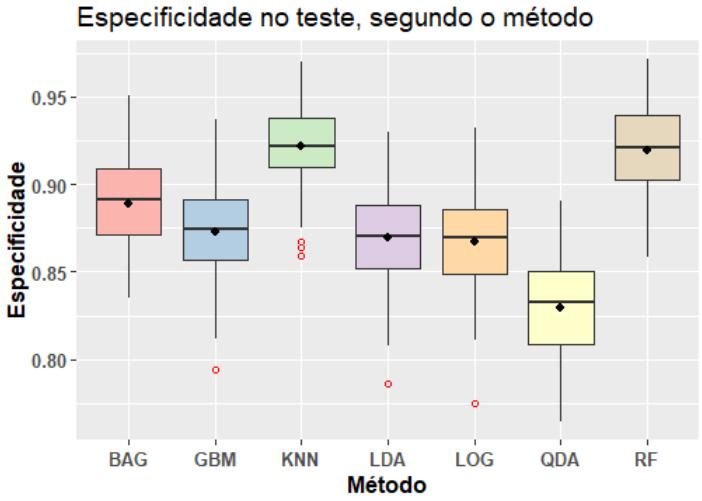


Figura 10: Comparação entre métodos pela especificidade

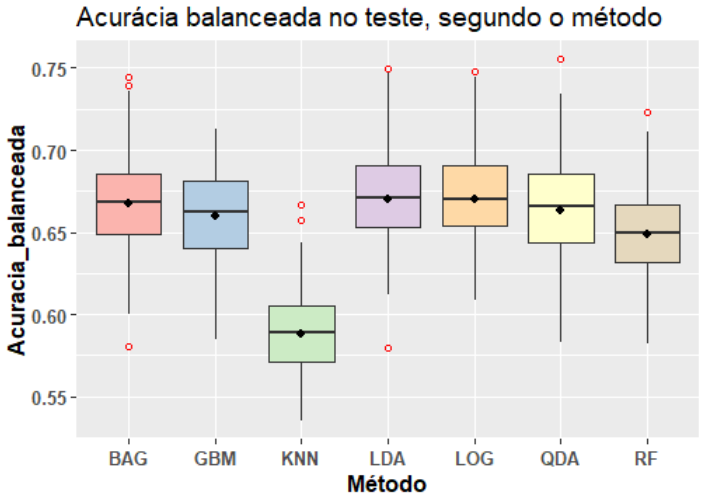


Figura 11: Comparação entre métodos pela acurácia balanceada

Para comparar se as médias das acurácias dos métodos diferem significativamente uma das outras, fizemos uma análise de variância (ANOVA), em que é obtida a estatística $F$ para o teste de hipóteses com $H_0$ sendo as médias iguais de todos os grupos e $H_1$ pelo menos uma média é diferente das outras. A estatística $F$ é definida por:

$$F = \frac{QMEnt}{QMDen}, \tag{3.1}$$

Onde $QMEnt$ é o quadrado médio entre os grupos e $QMDen$ é o quadrado médio dentro dos grupos, ambos obtidos pela divisão da soma dos quadrados dividido pelos respectivos graus de liberdade. Assim, sob a hipótese de normalidade, média zero, independência dentro e entre os grupos (no caso, são os métodos) e homocedasticidade dos

Tabela 3: Tabela ANOVA para acurácia dos métodos

| Fonte | Graus de Liberdade | Soma dos Quadrados | Quadrado Médio | F observado | Valor p |
|---|---|---|---|---|---|
| Método (Entre) | 6 | $0,1074$ | $0,017901$ | $27,09$ | $< 0,001$ |
| Resíduos (dentro) | 693 | $0,4579$ | $0,000661$ | | |
| Total | 699 | $0,5653$ | | | |

erros por grupo, $F$ tem distribuição $F$ de Snedecor com $k-1$ e $N-k$ graus de liberdade, onde $k$ é o número de grupos e $N$ o número de total de observações.

A tabela 3 fornece os resultados obtidos nos dados da acurácia entre as estimativas.

Com o valor p do teste menor do que 5%, rejeita-se a hipótese nula de que as médias são iguais, ou seja, pelo menos um dos métodos possui média da acurácia estatisticamente diferente dos outros.

É necessário verificar as hipóteses básicas da ANOVA citadas anteriormente para validação dos resultados.

Para a distribuição normal dos erros foi usado o teste de *Kolmogorov-Smirnov*, que é baseado na diferença vertical entre a distribuição empírica ($F_n(x)$) e a teórica ($F^*(x)$). A estatística de teste é definida por

$$D = max_{1 \leq i \leq n} \mid F(x_i) - F^*(x_i) \mid, \tag{3.2}$$

onde $x_i$ é a estatística de ordem. A hipótese nula é $H_0 : F(x_i) = F^*(x_i), \forall x_i \in \mathbb{R}$ e $H_1 : H_0 : F(x_i) \neq F^*(x_i)$, para algum $x$ (MORETTIN; BUSSAB, 2017).

Também foi usado teste de *Shapiro-Wilk*, que detecta diferença na normalidade dos dados devido a curtose ou assimetria (ou ambos). A estatística de teste $W$ é definida por

$$W = \frac{\left(\sum_{i=1}^{n} a_i y_i\right)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}, \tag{3.3}$$

onde $y_i$ é a estatística de ordem, $\overline{y}$ é a média amostral e $a_i$ são os pesos baseados na distribuição normal padrão. É indicado para amostras de tamanho 3 a 5000. Para saber mais detalhes sobre testes de normalidade ver (RAZALI; WAH et al., 2011).

Nos resíduos da ANOVA pela acurácia por método, foram obtidos a estatística $W = 0,99637$ com valor p de $0,1096$ pelo teste de Shapiro-wilk e $D = 0,033255$ com valor p

de $0,4212$ no teste de Kolmogorov-Smirnov para a acurácia dos modelos. Portanto, não rejeita-se a hipótese nula de normalidade da distribuição dos erros de acurácia ao nível de 5% de significância.

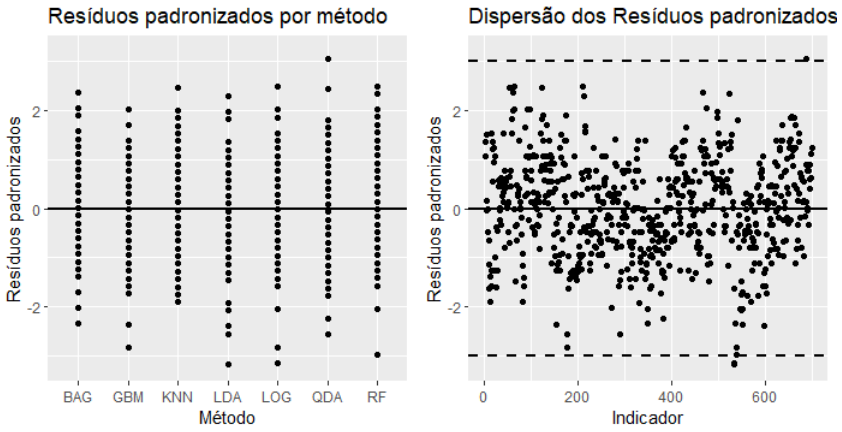


Figura 12: Dispersão dos resíduos padronizados geral e por método

Para testar a homoscedasticidade nos erros das sensibilidade dos grupos, fizemos pelo teste de Levene e Bartlett.

O teste de Levene usa a estatística de teste $W$ definida por

$$W = \frac{N-k}{k-1}\frac{\sum_{i=1}^{k} N_i(Z_i - Z_{..})^2}{\sum_{i=1}^{k} N_i(Z_{ij} - Z_i)^2}, \tag{3.4}$$

onde $k$ é o número de grupos, $N_i$ é o número de observações no $i$-ésimo grupo, $N = \sum_{i=1}^{k} N_i$, $Y_{ij}$ é o valor da variável medida (no caso é a acurácia) para a $j$-ésima observação do $i$-ésimo grupo, e

$$Z_{ij} = \begin{cases} \mid Y_{ij} - \overline{Y}_{i.} \mid, & \overline{Y}_{i.} \text{ é a média do } i\text{-ésimo grupo,} \\ \mid Y_{ij} - \widehat{Y}_{i.} \mid, & \widehat{Y}_{i.} \text{ é a mediana do } i\text{-ésimo grupo.} \end{cases}$$

$W$ com distribuição aproximada $F_{N-k,k-1}$. Para mais detalhes, ver Shukor (2016).

Pela função *leveneTest* do pacote *car* (FOX; WEISBERG, 2019) no $R$, obtivemos um valor $W = 0,7492$ pela mediana, pois a função permite escolher o teste pela mediana ou pela média, e um valor $p = 0,6102$.

E o teste de Bartlett com estatística de teste $M/C = 2,7041$ (valor $p = 0,845$), no qual sua estatística de teste $M/C$ é baseada na distribuição qui-quadrada com $I-1$ graus de liberdade,onde $I$ é o número de grupos, e é obtida pela razão de duas quantidades que dependem do tamanho da amostra e das variâncias amostrais (MORETTIN; BUSSAB, 2017).

Podemos observar pela gráfico da figura 12 que os resíduos padronizados estão dispersos em torno de zero, e estão aleatoriamente distribuídos, características de homocedasticidade dos resíduos.

Satisfeitas as hpóteses básicas, para obter qual ou quais métodos possuem médias diferentes em acurácia, primeiramente observe que temos $\binom{7}{2} = 21$ pares de médias diferentes. A tabela 4 mostrando todas as comparações foi feita usando o teste de Tukey para comparações múltiplas, disponível no pacote *stats* (R Core Team, 2020) com a função *TukeyHSD* no *R*. Esse teste só faz sentido quando rejeita-se a hipótese nula de que todas as médias são iguais. A estatística de teste é obtida por

$$q = \frac{\overline{x}_i - \overline{x}_j}{\sqrt{\frac{QMDen}{S}}}, \tag{3.5}$$

onde $\overline{x}_i$ é a média do grupo $i$, $i \neq j$, $S$ é o número de observações por grupo, considerando o caso em que todos os grupos possuem o mesmo tamanho e $QMDen$ é obtido pela tabela ANOVA (o mesmo que quadrado médio dos resíduos). Sob a hipótese nula de que todas as médias são iguais, $q$ segue uma distribuição de amplitude studentizada com amplitude (grupos) $A$ e $N - A$ graus de liberdade ($N$ é o total de observações). O termo $HSD$ significa *Honestly Significant Difference* (Diferença Honestamente Significativa, em português). Para mais detalhes, consultar Abdi e Williams (2010). A tabela 4 mostra a diferença entre as médias observadas de cada grupo comparado, os limites do intervalo de confiança e o valor $p$ ajustado para comparações múltiplas, que indica se a diferença é significativa ou não.

Podemos concluir que os métodos de LOG, LDA, RF e BAG possuem média de acurácia estatisticamente iguais. E também os métodos de LOG, LDA e GBM, porém, o método de GBM não possui média estatisticamente igual ao método de RF e BAG.

Tabela 4: Tabela comparações múltiplas de Tukey para a acurácia dos métodos

| Métodos | Diferença | $L_{inf}$ | $L_{sup}$ | Valor p |
|---|---|---|---|---|
| GBM-BAG | $-0,011$ | $-0,022$ | $0,000$ | $0,034$ |
| KNN-BAG | $-0,035$ | $-0,046$ | $-0,024$ | $< 0,001$ |
| LDA-BAG | $-0,006$ | $-0,017$ | $0,004$ | $0,590$ |
| LOG-BAG | $-0,007$ | $-0,018$ | $0,004$ | $0,442$ |
| QDA-BAG | $-0,026$ | $-0,037$ | $-0,015$ | $< 0,001$ |
| RF-BAG | $0,001$ | $-0,010$ | $0,011$ | $1,000$ |
| KNN-GBM | $-0,024$ | $-0,034$ | $-0,013$ | $< 0,001$ |
| LDA-GBM | $0,005$ | $-0,006$ | $0,016$ | $0,826$ |
| LOG-GBM | $0,004$ | $00,007$ | $0,015$ | $0,918$ |
| QDA-GBM | $-0,015$ | $-0,025$ | $-0,004$ | $0,001$ |
| RF-GBM | $0,012$ | $0,001$ | $0,023$ | $0,020$ |
| LDA-KNN | $0,028$ | $0,018$ | $0,039$ | $< 0,001$ |
| LOG-KNN | $0,028$ | $0,017$ | $0,038$ | $< 0,001$ |
| QDA-KNN | $0,009$ | $-0,002$ | $0,020$ | $0,187$ |
| RF-KNN | $0,035$ | $0,025$ | $0,046$ | $< 0,001$ |
| LOG-LDA | $-0,001$ | $-0,012$ | $0,010$ | $1,000$ |
| QDA-LDA | $-0,020$ | $-0,030$ | $-0,009$ | $< 0,001$ |
| RF-LDA | $0,007$ | $-0,004$ | $0,018$ | $0,479$ |
| QDA-LOG | $-0,019$ | $-0,030$ | $-0,008$ | $< 0,001$ |
| RF-LOG | $0,008$ | $-0,003$ | $0,018$ | $0,340$ |
| RF-QDA | $0,027$ | $0,016$ | $0,037$ | $< 0,001$ |

# 4 Conclusões

O objetivo de salvar especificamente as métricas descritas é estudar os tipos de erros em cada método, pois o interesse pode estar em minimizar o erro no grupo dos inadimplentes ou apenas nos adimplentes e utilizar a acurácia geral pode mascarar esses erros se os dados forem desbalanceados. No nosso caso, por exemplo, um classificador que resulte que todos os clientes serão adimplentes acertará 70% das vezes. Em compensação, errará 100% das predições no grupo dos inadimplentes, ou seja, uma especificidade de 0% e 0% de erro no grupo dos Adimplentes, gerando 100% de sensibilidade. Nesse caso, a acurácia balanceada é de 50%, que não é melhor do que um sorteio aleatório.

Em certas situações, como na abordada neste trabalho, é interessante maximizar a sensibilidade, ou seja, minimizar o erro no grupo dos inadimplentes, já que classificar um cliente como adimplente e ele vir a se tornar um cliente inadimplente gera perda de lucro para o banco. Esse erro é pior do que o contrário, quando um cliente é classificado como inadimplente e ele será adimplente.

Como observado no gráfico da sensibilidade, o método de RF e o KNN ficaram mais baixos do que os outros, como podemos confirmar pelo teste $t$ para médias. Porém, os modelos foram ajustados maximizando a acurácia.

Observa-se que, em geral, os métodos diferem em torno de 10% da acurácia para a acurácia balanceada e em geral, os métodos mais tradicionais como regressão logística e análise de discriminante linear obtiveram desempenho maior ou igual aos métodos mais dispendiosos computacionalmente, caso do *random forest*, *bagging* e *gradient boosting*.

Nenhum dos métodos analisados obteve sensibilidade significativamente acima de $0,5$, o que significa que para predizer um cliente como inadimplente, sendo que ele é inadimplente, os métodos se saem piores do que um sorteio aleatório, em geral. Isso também pode ser explicado pelos dados serem desbalanceados por natureza, já que $70,0\%$ das observações no teste são adimplentes. Outra possibilidade, é que as variáveis preditoras não conseguem discriminar bem os dois grupos da variável resposta.

Em geral, observo que pela acurácia, maximizando a acurácia, os métodos ficaram com resultados entre $65,0\%$ e $75,0\%$, exceto o KNN que ficou entorno de $65,0\%$. Como a amostra possui $70,0\%$ de observações adimplentes, isso significa que para esses dados e com essas variáveis preditoras os métodos não conseguem distinguir mais que 5% acima de uma predição de adimplência de uma observação, sem assumir nenhum modelo.

Seria necessário buscar outros métodos, outros tratamentos dos dados ou outras variáveis preditoras tais que consigam discriminar melhor a variável resposta.

# Referências

ABDI, H.; WILLIAMS, L. J. Tukey's honestly significant difference (hsd) test. *Encyclopedia of research design*, Sage Thousand Oaks, CA, v. 3, p. 583–585, 2010.

AL., A. S. et mult. *DescTools: Tools for Descriptive Statistics*. [S.l.], 2020. R package version 0.99.38. Disponível em: ⟨https://cran.r-project.org/package=DescTools⟩.

ANNETTE, J.; DOBSON, A.; BARNETT, A. An introduction to generalized linear models 3rd edn. *Chapman Hall/CRC*, p. 149–163, 2008.

ASUNCION, A.; NEWMAN, D. *UCI machine learning repository*. 2007.

BELLMAN, R. E. *Adaptive control processes: a guided tour*. [S.l.]: Princeton university press, 2015. v. 2045.

FOX, J.; WEISBERG, S. *An R Companion to Applied Regression*. Third. Thousand Oaks CA: Sage, 2019. Disponível em: ⟨https://socialsciences.mcmaster.ca/jfox/Books/Companion/⟩.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The elements of statistical learning: data mining, inference, and prediction*. [S.l.]: Springer Science & Business Media, 2009.

JAMES, G. et al. *An introduction to statistical learning*. [S.l.]: Springer, 2013. v. 112.

JAMES, G. et al. *ISLR: Data for an Introduction to Statistical Learning with Applications in R*. [S.l.], 2017. R package version 1.2. Disponível em: ⟨https://CRAN.R-project.org/package=ISLR⟩.

KUHN, M. *caret: Classification and Regression Training*. [S.l.], 2020. R package version 6.0-86. Disponível em: ⟨https://CRAN.R-project.org/package=caret⟩.

LIAW, A.; WIENER, M. Classification and regression by randomforest. *R News*, v. 2, n. 3, p. 18–22, 2002. Disponível em: ⟨https://CRAN.R-project.org/doc/Rnews/⟩.

MORETTIN, P. A.; BUSSAB, W. O. *Estatística básica*. [S.l.]: Saraiva Educação SA, 2017.

PRASAD, A. M.; IVERSON, L. R.; LIAW, A. Newer classification and regression tree techniques: bagging and random forests for ecological prediction. *Ecosystems*, Springer, v. 9, n. 2, p. 181–199, 2006.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2020. Disponível em: ⟨https://www.R-project.org/⟩.

RAZALI, N. M.; WAH, Y. B. et al. Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *Journal of statistical modeling and analytics*, v. 2, n. 1, p. 21–33, 2011.

THERNEAU, T.; ATKINSON, B. *rpart: Recursive Partitioning and Regression Trees*. [S.l.], 2019. R package version 4.1-15. Disponível em: ⟨https://CRAN.R-project.org/package=rpart⟩.

# APÊNDICE 1 – Tabela das variáveis utilizadas na análise

| Variável | Descrição | Categoria |
|---|---|---|
| Credibilidade | Se o cliente é merecedor ou não do crédito | 0: não merecedor<br>1: merecedor |
| Saldo_da_Cc | Balanço da conta corrente em Marco Alemão (DM) | 1: não possui conta<br>2: sem balanço ou débito<br>3: entre 0 (inclusive) e 200 DM<br>4: mais de 200 DM ou conta há pelo menos um ano |
| Duracao_do_credito_meses | Tempo de duração do crédito em meses | Numérico |
| Pagamento_de_creditos_anteriores | Pagamento de créditos no banco | 0: Pagamento hesitante em créditos anteriores<br>1: Conta problemática ou há créditos ativos em outro banco<br>2: Sem créditos anteriores ou pagou todos os créditos<br>3: Sem problemas com o crédito atual neste banco<br>4: Pagou todos os créditos neste banco |
| Objetivo_do_credito | Propósito do crédito | 0:Outros |

| | | |
|---|---|---|
| | | 1: Carro novo<br>2: Carro usado<br>3: Itens de mobília<br>4: Radio ou televisão<br>5: Eletrodomésticos<br>6: Reparos<br>7: Educação<br>8: Férias<br>9: Retreinamento ou reciclagem<br>10: Negócios |
| Montante_de_credito | Valor do crédito em DM | Numérico |
| Valor_da_poupanca_ou_acoes | Valor para emergências | 1: sem poupança ou não aplicável<br>2: Até 100 DM (exclusive)<br>3: De 100 a 500 DM (exclusive)<br>4: De 500 a 1000 DM (exclusive)<br>5: 1000 ou mais DM |
| Tempo_no_atual_emprego | Vínculo empregatício atual | 1: Desempregado<br>2: Menos de 1 ano<br>3: De 1 a 4 anos (exclusive)<br>4: De 4 a 7 anos (exclusive)<br>5: 7 sete anos ou mais |
| Percentual_da_renda_disponivel | Parcela em porcentagem da renda disponível | 1: 35 ou mais<br>2: Entre 25 (inclusive) e 35<br>3: Entre 20 (inclusive) e 25<br>4: Menos de 20 |
| Estado_civil_sexo | Estado civil por sexo | 1: Masculino: Divorciado ou separado<br>2: Masculino: Solteiro<br>3: Masculino: Casado ou Viúvo |

| | | |
|---|---|---|
| | | 4: Feminino |
| Garantias | Outros devedores ou fiadores | 1: Nenhum<br>2: Co-requerente<br>3: Fiador |
| Tempo_na_moradia_atual | Tempo que vive no local atual | 1: Menos de um ano<br>2: Entre 1 e 4 anos (exclusive)<br>3: entre 4 e 7 anos (exclusive)<br>4: 7 anos ou mais |
| Ativo_disponivel_mais_valioso | Bens mais valiosos | 1: Não possui ou não aplicável<br>2: Carro ou outros<br>3: Contrato de poupança com sociedade de construção ou seguro de vida<br>4: Proprietário de casa ou terreno |
| Idade | Idade em anos | numérico |
| Mais_creditos_em_execucao | Outros créditos ativos | 1: Em outros bancos<br>2: Na loja de departamentos ou casa de vendas pelo correio<br>3: Sem créditos ativos |
| Tipo_de_moradia | Categoria da moradia atual | 1: Apartamento livre<br>2: Apartamento alugado<br>3: Apartamento ocupado pelo dono |
| Numero_de_creditos_anteriores_neste_banco | Número de créditos neste banco, incluindo o atual | 1: um<br>2: dois ou três<br>3: quatro ou cinco<br>4: seis ou mais |

| | | |
|---|---|---|
| Ocupacao | Ocupação atual | 1:Desempregado ou não qualificado sem residência permanente<br>2: Não qualificado com residência permanente<br>3: Trabalhador qualificado ou funcionário qualificado ou funcionário público menor<br>4: Executivo ou autônomo ou funcionário público superior |
| Numero_de_dependentes | Número de pessoas com direito a alimentos | 1: 3 ou mais<br>2: de 0 a 2 |
| Telefone | Possui telefone para contato | 1: Não<br>2: Sim |
| Trabalhador_estrangeiro | Trabalhador estrangeiro | 1: Sim<br>2: Não |