

Machine Learning

Elton Massahiro Saito Loures

2020-12-29

Contents

Prefácio	5
0.1 Por que ler esse livro?	5
0.2 Estrutura	5
0.3 Informações a respeito do conteúdo	5
0.4 Agradecimentos	5
1 Introdução	7
1.1 Dicas de estudo	7
1.2 Dicionário	8
2 Inteligência Artificial (IA)	11
2.1 O que é IA? De onde veio esse conceito?	11
2.2 A arte de uma IA	14
3 Vertentes de uma IA e fundamentação filosófica	17
4 O Aprendizado de Máquina	21
4.1 Como a máquina aprende?	22
5 Fundamentos	23
5.1 Medidas de Importância	23
5.2 Teste de hipóteses e Análise de Variância	27
6 Pré-processamento	29
6.1 Dados faltantes e a Limpeza de dados	30
6.2 Transformação de dados	38
6.3 Features Selection - Seleção de atributos (SA)	43
7 Algoritmos de Aprendizagem - Parte I	47
7.1 Naive Bayes	47
7.2 Regressão	50
7.3 Gradiente Descendente (GD)	58
8 Algoritmos de Aprendizagem - Parte II	67
8.1 Máquina de Vetores Suporte - <i>Support Vectors Machine</i>	67

8.2	Árvore de Decisão (<i>Decision Tree</i>)	77
8.3	Elastic Net	84
8.4	KNN	84
8.5	Análise de Componentes Principais	84
8.6	Análise de Agrupamentos - Clusters	99
8.7	modelos nível III	121
8.8	grad boosting -> estudar boosting e bagging dentro de emseamble	121
8.9	Redes Neurais	121
9	Validação de um modelo	123
9.1	<i>Overfitting, Underfitting</i>	123
9.2	Validação Cruzada	124
9.3	Como escolher um bom modelo?	125
9.4	AOC e ROC	125

Prefácio

0.1 Por que ler esse livro?

0.2 Estrutura

0.3 Informações a respeito do conteúdo

0.4 Agradecimentos

```
install.packages("bookdown")  
# or the development version  
# devtools::install_github("rstudio/bookdown")
```

Remember each Rmd file contains one and only one chapter, and a chapter is defined by the first-level heading #.

To compile this example to PDF, you need XeLaTeX. You are recommended to install TinyTeX (which includes XeLaTeX): <https://yihui.name/tinytex/>.

Chapter 1

Introdução

Caro leitor, se você veio até esse livro é bem provável que passou e/ou ainda passa pelas mesmas dificuldades que todo estudante interessado nessa área.

Ao elevado número de pesquisas que fiz para aprender o que era a Inteligência Artificial, o que era o *Machine Learning* (Aprendizado de Máquina) e todos os outros temas similares, é nítido que ainda não está totalmente definido o conceito de cada um. É um ramo novo na área acadêmica, na indústria e em todo o mercado, com diversos temas, diversos modelos matemáticos, diversos modelos computacionais, diversos *softwares*, diversas aplicações e em diversas áreas. Diversos “diversos”... E o mais assustador é que esse campo une todos esses “diversos”, tornando o universo **caótico** ainda maior. Quando destaco o termo “caótico”, refiro exatamente pela ironia deste mote, todo esse universo confuso é aplicado em nosso cotidiano para organizar, analisar, diagnosticar e facilitar as coisas.

Poucos instruem como devemos enxergar todo esse cosmos que ao longo da história está passando por diversas construções para estruturar seu conceito. Com uma tentativa de trazer isso com base em artigos, livros, vídeos, podcasts e cursos, disponho este simples livro com o propósito de organizar a imagem que você, leitor, tem de Aprendizado de Máquina e entender os principais modelos utilizados tanto no meio acadêmico, quanto no mercado de trabalho.

1.1 Dicas de estudo

Não cabe a mim dizer como estudar, mas o que posso lhe aconselhar como principal ponto é a **paciência**. Temas como esse podem abranger qualquer campo, desde a filosofia até a área da saúde e portanto, do mesmo modo que se aplica a qualquer conteúdo, o mais importante é a base. Leia, releia, pesquise, veja vídeos, ouça um podcast, converse e discuta com colegas e professores a

respeito. Não se cobre de que precisa aprender o mais rápido possível, mas preze a qualidade do estudo.

Com intuito de explicar sobre Aprendizado de Máquina. Na seção **AQUI VOU COLOCAR A REFERENCIA DA SESSAO**, para facilitar o leitor dependendo de sua demanda de conteúdo, busquei separar em subseções a lógica computacional e a matemática. Tornando mais prático para o público que não tem interesse no modelo matemático e que busca o conhecimento de determinado assunto quanto ao público que demanda esse conteúdo.

1.2 Dicionário

- **Escalaes e Vetores:**
- **Espaço Vetorial e Transformação Linear:**
- **Assimetria e Curtose:**
- **Variância e Desvio padrão (Erro padrão):**
- **Covariância:** A covariância mede a relação linear entre duas variáveis. É possível utilizar a covariância para compreender a direção da relação entre as variáveis. Valores de covariância positivos indicam que valores acima da média de uma variável estão associados a valores médios acima da outra variável e abaixo dos valores médios são igualmente associado. Valores de covariância negativos indicam que valores acima da média de uma variável estão associados com valores médios abaixo da outra variável.
- **Distribuição normal:**
- **Distribuição binomial:**
- **Disitruição de Poisson:** (Banzatto and Kronka, 1992) Quando número de plantas daninhas por parcela, número de insetos capturados em armadilhas luminosas, número de pulgões ou ácaros por folhas, etc.
- **Teorema de Bayes:** quando tratamos de probabilidades, $P(A|B)$ e $P(B|A)$ podem ser parecidos, mas possuem grande diferença entre as probabilidades que representam. Por exemplo $P(A|B)$ pode se referir sobre a probabilidade de uma pessoa que cometeu um furto (B) ser condenada (A) e $P(B|A)$ seria a probabilidade de uma pessoa que foi condenada por furto ter efetivamente cometido um crime. A causa se torna o efeito e o efeito se torna a causa (Freund, 2009).

Pela regra geral de multiplicação que afirma que a probabilidade da ocorrência de dois eventos é o produto da probabilidade da ocorrência de um deles pela probabilidade condicional da ocorrência do outro evento, temos:

$$P(A \cap B) = P(A).P(B|A) \text{ ou } P(A \cap B) = P(B).P(A|B) \quad (1.1)$$

Igualando ambas expressões, temos: $P(A) \cdot P(B|A) = P(B) \cdot P(A|B)$ e portanto, dividindo por $P(B)$, obtém-se o Teorema de Bayes que descreve a probabilidade de um evento, baseado em um conhecimento *a priori* que pode estar relacionado ao evento:

$$P(A|B) = \frac{P(A) \cdot P(B|A)}{P(B)} \quad (1.2)$$

Para B_n e A_k atributos, podemos reescrever:

$$P(A_k|B_1, \dots, B_n) = \frac{P(A_k) \cdot P(B_1, \dots, B_n|A_k)}{P(B_1, \dots, B_n)} \quad (1.3)$$

Exemplo: este exemplo pode ser encontrado em Freund (2009). Numa certa empresa, 4% dos homens e 1% das mulheres têm mais de 1,75m de altura, respectivamente, sendo que 60% dos trabalhadores são mulheres. Um trabalhador é escolhido ao acaso.

A) Qual a probabilidade de que tenha mais de 1,75m?

Solução: Temos de informação de que 60% dos trabalhadores são mulheres e que 1% delas possuem mais de 1,75m. Portanto 40% dos trabalhadores são homens, sendo 4% deles com mais de 1,75m. Logo temos que:

$$P(> 1,75m) = (0,04 \cdot 0,4) + (0,01 \cdot 0,6) = 0,022 \rightarrow 2,2\% \text{ de probabilidade de que tenha mais de 1,75m.}$$

B) E que seja homem dado que o trabalhador escolhido tenha mais de 1,75m?

Solução: pelo enunciado “que seja homem dado que o trabalhador escolhido tenha mais de 1,75m”, podemos perceber que já possuímos uma afirmação que já foi escolhido uma pessoa que tenha mais que 1,75m e queremos saber se é homem. Por meio da questão anterior sabemos a probabilidade $P(> 1,75m)$. Portanto:

$$P(H|> 1,75m) = \frac{P(> 1,75m|H) \cdot P(H)}{P(> 1,75m)} = \frac{0,04 \cdot 0,4}{0,022}$$

$\rightarrow 72,73\%$ de probabilidade de ser homem dado que seja maior que 1,75m.

- **Função de verossimilhança:** a verossimilhança L de um conjunto de parâmetros θ , com dada informação x . É igual a probabilidade da mesma observação x ter ocorrido dados os valores dos mesmos parâmetros θ . Conhecendo um parâmetro θ , a probabilidade condicional de x é $P(x|\theta)$, mas se o valor de x é conhecido, pode-se realizar inferências sobre o valor de θ (Bolfarine and Sandoval, 2001).

$$L(\theta|x) = P(x|\theta) \quad (1.4)$$

Para “ n ” valores:

$$L(\theta|x_1, \dots, x_n) = \prod_{i=1}^n P(x_i|\theta) \quad (1.5)$$

Geralmente utiliza-se o logaritmo natural em verossimilhança $L(\theta|x) = \ln L(\theta|x)$ como função suporte e facilitar em seu estudo.

Para facilitar a compreensão, considere a observação de que você esteja ouvindo barulho em sua sala de estar num dia de natal (observação x), você parte da hipótese inicial que poderia ser o “Papai Noel” lhe entregando presentes (hipótese θ). A probabilidade de ser Noel lhe entregando presente apenas porque ouviu o barulho, isto é, $P(\theta|x)$ é baixa. No entanto o contrário, você com a afirmação de que é o Noel lhe entregando presentes, a probabilidade de haver barulho em sua sala de estar é bem alta, logo a verossimilhança $L(\theta|x) = P(x|\theta)$.

- **Parâmetros:** podem ser vistos como características numéricas de um modelo ou população. Os valores não podem ser mensurados diretamente mas que podem ser estimados através dos dados de uma amostra.
- **Paramétrico x Não Paramétrico:**
- **Correlação:**
- **Supervisionada x Não supervisionada:**
- **Produto Interno:**
- **Multiplicadores de Lagrange:***
- **Karush-Kuhn-Tucker (KKT):**
- **Bias:** <https://iaexpert.academy/2020/09/28/importancia-do-bias-nas-redes-neurais/>

Chapter 2

Inteligência Artificial (IA)

2.1 O que é IA? De onde veio esse conceito?

Humano (taxonomicamente *Homo sapiens*), termo que derivado do latim “homem sábio”. Pensamos, analisamos, aprendemos, prevemos e manipulamos. Somos seres **inteligentes**. Já pesquisou o significado de “inteligência” no dicionário?

É importante entender o conceito de inteligência, pois nem tudo que o ser humano faz pode ser classificado como inteligente. Aprender somar para calcular a soma de $2 + 2$ é uma ação inteligente, mas copiar o resultado e colocar em sua folha de resultados que é 4 pode não ser tanto assim. Da mesma forma uma calculadora que executa um código passado por um humano, contendo dentro todos os passos a serem executados (algoritmos) para resolver esse cálculo, não é considerada.

Quando tratamos da inteligência artificial não é fácil definir o que ela é. O seu próprio conceito vem sendo discutido e moldado ao longo do tempo. A idéia de construir uma máquina pensante ou um ser artificial que se assemelhasse aos humanos é muito antigo. O mito do Golem, por exemplo, um dos primeiros seres artificiais criados pelo homem. Dizia a lenda que o mito do Golem surgiu no século XIII quando uma matéria informe tornou-se num homúnculo a partir da invocação mágica de Elijah de Chelm que escreveu em sua fronta “*Shemhamforash*” - nome secreto de Deus (MOSER, 2006). Na literatura foi publicado o famoso romance *Frankenstein* (Shelley, 1818) que relata a história de um estudante que constrói um monstro em seu laboratório. Mas como ela realmente surgiu?

O primeiro trabalho a ser reconhecido como IA foi elaborado por McCulloch and Pitts (1943) que tinha como propósito estudar como os neurônios podiam funcionar, modelando uma rede neural simples com circuitos elétricos. Os mesmos

autores sugeriram que as redes neurais definidas em conformidade poderiam ser capazes de aprender. Por seguinte, Hebb (1949) escreveu *The Organization of Behavior* que fortalecia as teorias de que o condicionamento psicológico estava presente em qualquer parte dos animais. Teve como a premissa de que dois neurônios participantes de uma sinapse, têm ativação simultânea, então a força da conexão entre eles deve ser seletivamente aumentada, ou seja, os caminhos neurais são fortalecidos cada vez que são utilizados.

Em 1950, o matemático Claude E. Shannon publicou um artigo sobre como “ensinar” seu computador a jogar xadrez (Shannon, 1950); no mesmo ano Alan Turing, em “*Computing Machinery and Intelligence*” (TURING, 1950), sugeriu que, ao invés de perguntarmos se as máquinas podem pensar, devemos perguntar se as máquinas podem passar por um teste de inteligência comportamental, o teste de Turing. Uma forma de avaliar se uma máquina consegue se passar por um humano em uma conversa por escrito com um avaliador passando no teste caso o avaliador não conseguisse identificar se estava conversado com um computador ou com outro ser humano. No ano seguinte, os estudantes Marvin Minsky e Dean Edmonds construíram o SNARC, o primeiro computador de rede neural que simulava uma rede de 40 neurônios.

Em 1956 houve a conferência de verão em Dartmouth College (Hanover, New Hampshire), foi oficializada o nascimento da IA. John McCarthy, Minsky, Claude Shannon e Nathaniel Rochester elaboram uma proposta a fim de reunir pesquisadores dos Estados Unidos interessados em teoria de redes neurais, autômatos e estudo da inteligência:

Propusemos que um estudo de dois meses e dez homens sobre inteligência artificial fosse realizado durante o verão de 1956 no Dartmouth College, em Hanover, New Hampshire. O estudo foi para prosseguir com a conjectura básica de que cada aspecto de aprendizado ou qualquer outra característica da inteligência pode, em princípio, ser descrita com tanta precisão a ponto de que uma máquina pode ser feita para simulá-la. Será realizada uma tentativa para descobrir como fazer com que as máquinas usem a linguagem, a partir de abstrações e conceitos, resolvam os tipos de problemas hoje reservados aos seres humanos e se aperfeiçoarem. Achamos que poderá haver avanço significativo em um ou mais desses problemas se um grupo cuidadosamente selecionado de cientistas trabalhar em conjunto durante o verão.

— “A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence” , McCarthy et al. (2006) , Agosto de 1955.

Entre diversas ideias e apresentações, Allen Newell e Herbert Simon apresentaram o programa logic theorist, capaz de provar diversos teoremas e segundo Simon, capaz de pensar não numericamente. Apesar de muitos editores não se agradarem, esta importante proposta trouxe nos próximos anos, uma dominação nesse campo (Russel and NORVIG, 2004):

- *General Problem Solver* (GPS), projetado por Newell and Shaw (1959), é um sistema que buscava imitar o homem na forma de resolver problemas. Concluíram de que a forma em como dividia um objetivo em sub objetivos e possíveis ações era similar à forma em como o homem fazia. Esta pesquisa ajudou a estabelecer os fundamentos teóricos dos sistemas de símbolos e forneceram à área da IA uma série de técnicas de programação voltadas à manipulação simbólica, por exemplo, as técnicas de busca heurística;
- a IBM produz alguns dos primeiros programas de IA, entre os quais, em 1959 o Geometry Theorem Prover;
- Arthur Samuel desenvolveu um programa capaz de jogar damas ao nível de um jogador de torneio. O programa jogava melhor do que o seu autor;
- John McCarthy no MIT, em 1958, define a linguagem de programação Lisp (List Processing) que se transformou na linguagem dominante da IA e publicou um artigo intitulado “Programs with common sense” (McCarthy, 1968), onde descrevia um programa hipotético designado por “Advice taker”, o qual pode ser visto como o primeiro sistema completo da IA;
- Slagle (1963), com o programa SAINT, foi capaz de resolver problemas de cálculo integral;
- Evans (1964) e Bobrow (1967), com os respectivos programas ANALOGY e STUDENT, resolviam problemas de análises geométricas semelhantes aos testes de QI e problemas clássicos de álgebra.
- Em base de Huffman (1971), Waltz (1975), Winograd (1972), Winston (1970) e Fahlman (1974), foi elaborado o mundo de blocos, que consiste em um conjunto de blocos sólidos colocados sobre uma mesa de modo que a mão de um robô reorganize-os.

Claro que os primeiros sistema houveram dificuldades com problemas mais difíceis. Desde traduções que exigiam conhecimento profundo para solucionar ambiguidades, por exemplo, como situações de necessidade de hardwares melhores e limitações fundamentais nas estruturas simples. Com ressalva, em *Perceptrons* (Minsky and Papert, 1969) demonstra que embora suas redes neurais simples (*perceptrons*) pudessem aprender, eram capazes de representar muito pouco. Mas com exigência da formalização acadêmica na década de 70, permitiu o desenvolvimento de sistemas com grande desempenho intelectual com perspectivas industriais e comerciais, surgindo novos sistemas dispostos a resolver problemas mais complexos do que antes:

- DENDRAL (Buchanan et al., 1969), analisa compostos orgânicos a fim de determinar sua estrutura molecular;
- MYCIN (Buchanan and Shortliffe, 1984), Sistema pericial (expert system) foi capaz de diagnosticar infecções no sangue.

E sucessivamente foi crescendo este enorme e maravilhoso campo. O Japão lança o projeto “*Fifth Generation*” para construir em dez anos computadores inteligentes com capacidade de fazer milhões de inferências por segundo em 1981; uso de IA na guerra do Golfo em 1991; sistemas de perícia para casos médicos no mesmo ano; sistemas para condução de veículos automotores e detectores de colisões nas ruas (1993); reserva de viagens (1994); brinquedos inteligentes (2000); computador que se comunica ao nível de uma criança com 15 meses (2001). Ao longo dos anos da história da ciência da computação, a ênfase em *algoritmos* e tratamento de dados vem aumentando.

2.2 A arte de uma IA

Atualmente, existem muitas atividades, pesquisas e aplicações em diversos temas que muitas vezes nem perbemos:

- Recomendações de mídia: com base em seu perfil de uso, o algoritmo compara filmes, músicas, clips, etc com base em vários usuários que possuem os gostos similares ao nosso. Recomendando aquilo que provavelmente irá nos agradar. Por exemplo *Spotify*, *YouTube* e *Netflix*.
- Reconhecimento de fala e assistentes virtuais: já refletiu sobre como funciona sua Google Assistente? Com ondas sonoras emitidas pela voz, o algoritmo reconhece palavras, frases e até mesmo o timbre, fornecendo respostas de acordo com o que recebe.
- Jogos: a inteligência artificial desenvolvida pela *OpenAi* conseguiu derrotar uma das melhores equipes do *Dota 2* do mundo.
- Logística: a crise de 1991, por exemplo, no Golfo Pérsico. Foi utilizada a DART (Cross and Walker, 1994), uma ferramenta que envolveu até 50.000 veículos, transporte de carga aérea e pessoa simultaneamente com o objetivo de realizar um planejamento logístico automatizado levando em conta rotas, pontos de partida e resolução de conflitos.
- Reconhecimento de imagens: identificação de objetos, pessoas, animais e qualquer figura com base em exemplos prévios, como por exemplo identificador de pessoas em uma foto do Facebook.
- Verificação de compras: detecção de comportamentos suspeitos a partir do histórico e perfil do usuário, como a *e-commerce*.
- Automóveis autônomos: por meio do algoritmo, visualiza a estrada, as placas, condição climática, outros veículos e diversos outros obstáculos para tomar decisões de seu trajeto sem a necessidade de uma pessoa.

Poderíamos falar desde exemplos de inteligência artificial aplicados em casos jurídicos, diagnósticos na área da saúde, identificadores de *fake news* (notícias falsas) até a robótica. É uma extensa lista de exemplos na área que até hoje estão em desenvolvimento em busca de cada vez mais melhorar. A AGI (*Artificial*

General Intelligence), ou Inteligência Artificial Geral, trabalha na criação de uma inteligência artificial generalista, similar a humana, capaz de ser especialista em uma área, mas também aprender com facilidade outras. Uma área que se tornou uma das principais linhas de pesquisa e nos dias de hoje gera discussões sobre até onde a IA pode alcançar.

Chapter 3

Vertentes de uma IA e fundamentação filosófica

Os filósofos têm estado por aí há muito mais tempo que os computadores e vêm tentando resolver algumas questões que se relacionam à IA: como a mente funciona? É possível que as máquinas ajam com inteligência, de modo semelhante às pessoas, e, se isso acontecer, elas realmente terão mentes conscientes? Quais são as implicações éticas de máquinas inteligentes?

“Inteligência Artificial”, RUSSEL and Norvig (2013).

Com todo o desenvolvimento da IA, os algoritmos podem funcionar em níveis humanos em tarefas que aparentemente envolvem julgamento humano ou, como Turing acrescentou, “aprender a partir da experiência” e a capacidade de “distinguir o certo do errado”(RUSSEL and Norvig, 2013). Paul Meehl (Meehl, 1954) analisou os processos de tomada de decisão de especialistas treinados em tarefas subjetivas como prever o sucesso de um aluno em um programa de treinamento ou a reincidência de um criminoso e descobriu que algoritmos simples de aprendizado estatístico fizeram previsões melhores que os especialistas.

A reflexão sobre “máquinas inteligentes e pensantes” é recente em nossa história e passa por longas discussões sobre o alcance dessa inteligência. Desde a classificação elaborada pelo filósofo John Searle em 1980, tomou-se na doutrina em geral a divisão do uso da inteligência artificial em “**fraca**” e “**forte**” (Searle, 1980).

A inteligência artificial **fraca** “nos permite formular e testar hipóteses de forma mais rigorosa e precisa”, no entanto, ela é dependente da inserção do conhecimento fornecido pelo ser humano que a programa. A máquina não é capaz de produzir raciocínios próprios, autônomos (Searle, 1980; Guimarães, 2019). Searle também explica que a máquina adequadamente preparada é realmente uma

mente, no sentido de que os computadores que recebem os programas certos poderiam estar, literalmente, preparados para compreender e ter outros estados cognitivos (Searle, 1980).

Searle (1980) em seu *naturalismo biológico*, critica a inteligência artificial forte pois, segundo ele, as máquinas não possuem a complexidade de sistema nervoso, neurônios com axônios e dendritos e tudo mais. Para corroborar sua crítica, Searle descreve uma situação hipotética simulando um programa que passa pelo teste de turing e que “não entende nada de suas entradas e saídas”, não havendo os requisitos para ser considerada uma mente.

O sistema foi nomeado como “**quarto chinês**”. Ele se usa como exemplo com a situação de que não tem conhecimento da língua chinesa, estaria trancada e isolado num quarto recebendo uma folha de papel com ideogramas em chinês escritos. Por não conhecer a língua, não possui ideia alguma do que se trata. Em seguida, ele recebe uma segunda folha com ideogramas chineses acompanhados de um conjunto de regras em inglês (língua nativa) que permitem a correlação da segunda folha com a primeira. Por fim, recebe uma terceira folha com ideogramas chineses, com regras em inglês que orientam a dar em respostas específicos ideogramas chineses associados a outros ideogramas da terceira folha, correlacionando os elementos da atual com as duas anteriores. As pessoas externas do quarto denominam a terceira folha como o “script”, a segunda folha de “história” e a primeira folha de “questões”. Essas pessoas consideram que os símbolos que Searle entregou em resposta à terceira folha são as “respostas às questões” e todo o conjunto de regras que lhe foi entregue são o “programa” (Guimarães, 2019).

Com o tempo Searle se torna melhor em dar respostas de acordo com as regras que permitem manipular os ideogramas chineses e de maneira similar ocorre com os programadores externos do quarto, que ficam bons em escrever os programas do ponto de vista externo. Qualquer pessoa que observa as respostas de Searle não contestaria de que Searle não fala chinês. Da mesma forma se o mesmo experimento fosse feito com textos em inglês, sua língua nativa, ele daria respostas em patamares semelhantes (Guimarães, 2019).

Searle conclui que no caso em chinês ele opera como um computador, respondendo corretamente mas sem a menor ideia do que está respondendo. Ao caso em inglês, ele irá responder como um ser humano e com consciência de suas respostas. O quarto se refere ao computador, o ser humano ao *software* de IA. Com isso ele assume que só seria possível produzir artificialmente uma máquina com sistema suficientemente semelhante a nós se poder duplicar exatamente as causas e seus efeitos, assim de fato seria possível produzir consciência, intencionalidade (fenômeno biológico dependente da bioquímica específica de suas origens) e tudo o mais usando princípios químicos diferentes dos usados por seres humanos (Searle, 1980).

Em contestamento a Searle, Daniel Dennett defende o projeto de Turing porque agir inteligente consiste na capacidade de processamento de informação (Den-

nett, 2009). Segundo Dennett, o problema da mente deve ser abordado com base na teoria evolutiva darwiniana pois o que entendemos por mental está relacionado ao tipo de resposta que nosso organismo dá para as demandas que estão para além daquelas que dizem respeito à manutenção da vida (da Silveira, 2013). Para ele, como ele denomina de *intencionalidade intrínseca*, Seartle errou em atribuir aos humanos a intencionalidade produzida exclusivamente pela interação das partes que constituem uma totalidade complexa, não necessitando de influências ou interferências externas. Para Dennet nossa intencionalidade não é original (Dennett, 2009).

Para Dennet o principal argumento criticando o argumento do quarto chinês, é a forma como investigamos os fenômenos mentais. É uma região que possibilita infinitas especulações, sendo o método das ciências empíricas o mais apropriado ao estudo da mente (da Silveira, 2013).

A diferença entre ambos é de natureza filosófica com ontologias e epistemologias divergentes. É notável a importância das discussões filosóficas. O antagonismo dicotômico dos dois filósofos possuem fundamentações que auxiliam na compreensão da mente. Quando teremos estas respostas? As máquinas serão capazes de raciocinar algum dia? Até onde uma IA pode chegar?

Chapter 4

O Aprendizado de Máquina

Agora que entendemos o conceito e a origem de uma IA, podemos entrar no tão esperado **Machine Learning (ML)**. Alguns pensam erroneamente ser algo distinto de uma IA, mas é importante entender que ela é um campo específico da inteligência artificial que tem como base a ideia de que sistemas podem aprender com dados e iterações, identificar padrões para que aprimorem seu desempenho diante de problemas específicos e possam tomar decisões com a menor intervenção humana possível. Como modelos estatísticos, busca entender a estrutura dos dados modelos que atendam a certos pressupostos - muitas vezes não temos conhecimento de como essa estrutura se parece.

Samuel (1959), engenheiro do MIT popularizou o termo “*Machine Learning*” (Aprendizado de Máquina), descrevendo o conceito com “um campo de estudo que dá aos computadores a habilidade de aprender sem terem sido programados para tal” (Simon, 2013). Com a expansão da internet e seu abundante armazenamento de dados na web, o *Big data*, foi necessário - ainda é - aprimorar sistemas de organização, classificação, análise de dados e identificação de padrões para tratá-los. Isso fez com que o Aprendizado de Máquina entrasse em destaque e passasse a ser uma das áreas mais importantes. Na seção 2 foi apresentado alguns exemplos de aplicações de IA, o mesmo se aplicam para o ML.

Um aprendizado de máquina **não** é o mesmo que uma lista de instruções. Imagine uma criança aprendendo a andar de bicicleta, ela pode até receber algumas instruções para melhorar seu aprendizado, mas provável que ela irá aprender melhor com a tentativa e erro. Pedala, cai, levanta, pedala novamente e assim sucessivamente até ela realmente saber andar. Da forma similar ocorre com o Aprendizado de Máquina.

4.1 Como a máquina aprende?

Você é um vendedor e está interessado em clientes “bons pagadores” e “maus pagadores”. Para cada cliente, possui um conjunto de dados como: idade, quantidade de faturas pagas antes do vencimento nos últimos 12 meses, quantidade de faturas atrasadas nos últimos 12 meses, região que reside, tempo de cadastro, etc. Você já se encontra com um banco de dados muito grande de clientes com seus respectivos dados e classificações como bons pagadores e maus pagadores e pretende utilizar um algoritmo de ML para aprender com esses dados de modo que, quando você receber o banco de dados de um novo cliente, esse algoritmo pode prever se a tendência desse cliente seria de bom pagador ou mau pagador.

Primeiramente, você iria alimentar seu algoritmo de ML com os dados históricos que passaram por toda uma análise se havia dados faltantes, redundantes, etc e já classificados entre cliente bom pagador e mau pagador e suas respectivas características para treiná-lo. Com estes dados o algoritmo irá aprender por meio de com quais condições são necessárias para o cliente ser classificado como bom pagador ou mau pagador. Importante ressaltar que existem diferentes algoritmos de Aprendizado de Máquina que poderiam resolver esse problema, de acordo com modelos estatísticos e comandos computacionais que atendam a certos pressupostos.

- **Como verificarmos se os dados já estão bons para aplicar o algoritmo? Quais modelos podemos aplicar? Como sabemos que essas previsões são confiáveis? Como evitar problemas de um modelo ruim?**

Agora que temos todo o contexto histórico podemos ir adiante do conteúdo aplicado. A tópico a seguir trataremos um pouco de alguns conceitos muito importantes que são utilizados em muitos algoritmos de Aprendizado de Máquina para que possamos gradualmente responder estes questionamentos.

Chapter 5

Fundamentos

5.1 Medidas de Importância

Um atributo é dito importante se quando removido a medida de importância considerada em relação aos atributos restantes é deteriorada, seja a precisão da medida, consistência, informação, distância ou dependência

Tradução de Liu and Motoda (2012).

É fundamental estimarmos a importância de um atributo, tanto uma avaliação individual quanto à avaliação de subconjuntos de atributos. É uma questão complexa e multidimensional (Liu and Motoda, 2012). Podemos avaliar se os atributos selecionados pela etapa do pré-processamento auxiliam a melhorar a precisão do classificador ou a simplificar algum modelo construído. A seguir, apresenta-se algumas medidas utilizadas (Lee, 2005).

5.1.1 Medidas de Informação

As medidas de informação determinam o ganho de informação a partir de um atributo. O ganho de informação é definido como a diferença entre a incerteza *a priori* e a incerteza *a posteriori* considerando-se o atributo X_i . X_i é preferido ao atributo X_j se seu ganho de informação for maior que de X_j . Uma das mais utilizadas é a entropia que normalmente é usada na teoria da informação para medir a pureza ou impureza de um determinado conjunto.

Shannon (1948), tomou como “ponto de partida” encontrar uma forma matemática de medir o quanto de informação existe na transmissão de uma mensagem de um ponto a outro, denominando-a entropia. Sua proposta baseava-se na ideia de que o aumento da probabilidade do próximo símbolo diminuiria o tamanho da informação. Com isso, a entropia pode ser definida como a quantidade de incerteza que há em uma mensagem e que diminui à

medida que os símbolos são transmitidos (vai se conhecendo a mensagem), tendo-se então a informação, que pode ser vista como redução da incerteza (Shannon, 1948; Paviotti and Magossi, 2019). Por exemplo: ao utilizarmos como idioma a nossa língua portuguesa e ao transmitir como símbolo a letra “q”, a probabilidade do próximo símbolo ser a letra “u” é maior que a de ser qualquer outro símbolo, enquanto que a probabilidade de ser novamente a letra “q” é praticamente nula (Paviotti and Magossi, 2019).

Shannon define que a entropia pode ser calculada por meio da soma das probabilidades de ocorrência de cada símbolo pela expressão $\sum p_i = 1 = 100\%$, em que p_i representa a probabilidade do i -ésimo símbolo que compõe a mensagem. Segundo ele, estes símbolos devem ser representados através de sequências binárias, utilizando das propostas de Nyquist (1924) e Hartley (1928). Sua proposta consistia em representar símbolos de um alfabeto através de um logaritmo de acordo com suas respectivas unidades de informação. A entropia proposta por ele é obtida pela média das medidas de Hartley (Moser and Chen, 2012).

Se A é discreto com distribuição de probabilidade $p(A)$, a entropia será:

$$H(A) = - \sum p(A) \log_2(p(A)) \quad (5.1)$$

Para facilitar a compreensão, vamos supor um exemplo de um questionário com resposta binária entre “sim” e “não”: quanto mais distribuído as probabilidades das respostas, mais desorganizada é, logo maior sua entropia, do contrário caso for uma probabilidade de ser zero “sim”/“não” ou de ser 1 (100%), ou seja, ter apenas uma opção de resposta, será menos distribuído e portanto menor sua entropia.

O ganho de informação portanto mede a redução da entropia (nesse caso) causada pela partição dos exemplos de acordo com os valores do atributo.

$$\text{Ganho de Informação}(D, T) = \text{entropia}(D) - \sum_{i=1}^k \frac{|D_i|}{|D|} \cdot \text{entropia}(D_i) \quad (5.2)$$

É muito utilizado em algoritmo de **Árvore de decisão** que será apresentado na seção 8 com um exemplo de seu uso.

5.1.2 Medidas de Similaridade e Dissimilaridade

É provável que já tenha ouvido falar em algo como medidas de distância, separabilidade, discriminação, divergência, similaridade ou dissimilaridade. É uma questão importante decidir até que ponto dois elementos de um conjunto de dados podem ser considerados com características semelhantes ou não.

feitos/Livro ML/bookdown-demo-master/Figuras/entropia.bb

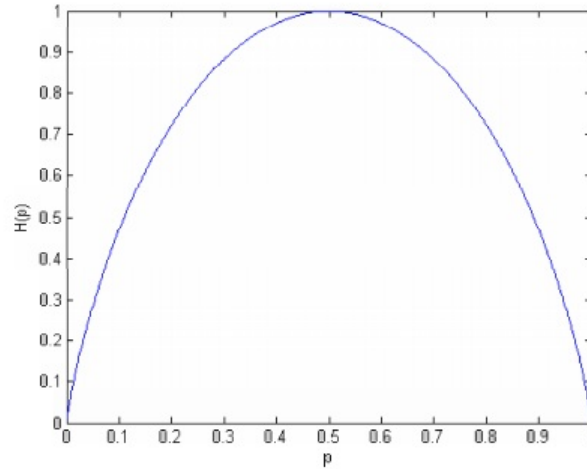


Figure 5.1: Gráfico de Probabilidade x Entropia.

Supondo que possuímos um conjunto de dados constituído por n elementos amostrais. O objetivo é agrupar esses elementos em g grupos de acordo com um vetor $X_j = [X_{1j} X_{2j} \dots X_{pj}]'$, $j = 1, 2, \dots, n$ que representa o valor observado da variável i medida no elemento j . Lembrando que existem diversas medidas diferentes e que produzem um determinado tipo de agrupamento de acordo com sua metodologia. A seguir será apresentado algumas comuns no ramo.

5.1.2.1 Distância Euclidiana

A distância Euclidiana entre dois elementos X_l e X_k , com $l \neq k$, é definida por:

$$d(X_l, X_k) = [(X_l - X_k)'(X_l - X_k)]^{1/2} = \left[\sum_{i=1}^p (X_{il} - X_{ik})^2 \right]^{1/2} \quad (5.3)$$

sendo comparado os dois elementos amostrais em cada variável pertencente ao vetor. Por exemplo, a tabela a seguir apresenta a renda mensal (em salários mínimos) e a idade de seis indivíduos de uma localidade (Mingoti, 2007).

Table 5.1: Renda e Idade de 6 indivíduos (Mingoti, 2007).

Indivíduo	A	B	C	D	E	F	Média	Desvio Padrão
Renda	9,6	8,4	2,4	18,2	3,9	6,4	8,15	5,61
Idade	28	31	42	38	25	41	34,17	7,14

A distância Euclidiana entre os indivíduos A e B nas variáveis Renda e Idade será:

$$d(X_A, X_B) = [(9,60 - 8,40)^2 + (28 - 31)^2]^{1/2} = 3,23$$

Assim sucessivamente para cada uma das observações, obtemos a matriz:

$$\begin{bmatrix} & A & B & C & D & E & F \\ A & 0 & & & & & \\ B & 3,23 & 0 & & & & \\ & & & & & & \end{bmatrix}$$

5.1.2.2 Distância Ponderada

Também conhecido como **distância generalizada**, esta metodologia tem como base uma matriz $A_{p \times p}$ de ponderação. Quando $A_{p \times p}$ for uma matriz identidade, esta distância generalizada será a **distância Euclidiana**; se $A_{p \times p}$ for a matriz inversa da matriz de covariâncias amostrais $S_{p \times p}^{-1}$, será a **distância de Mahalanobis (1936)** e se $A_{p \times p} = \text{diag}(\frac{1}{p})$, teremos a distância **Euclidiana Média**. A escolha dessa matriz $A_{p \times p}$ reflete o tipo de informação que o pesquisador pretende utilizar na ponderação das diferenças das coordenadas dos vetores em estudo (Mingoti, 2007), a distância de Mahalanobis, por exemplo, leva em consideração as possíveis diferenças de variâncias e as relações lineares entre as variáveis, em termos de variância, na ponderação.

$$d(X_l, X_k) = [(X_l - X_k)' A_{p \times p} (X_l - X_k)]^{1/2} \quad (5.4)$$

Continuando com dados da @tab:dadosrendaa

5.1.3 Medidas de Dependência

feitos/Livro ML/bookdown-demo-master/Figuras/correlacao.bb

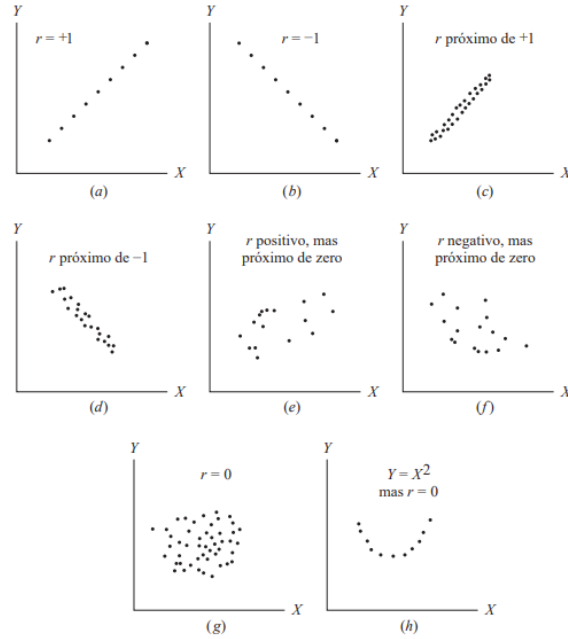


Figure 5.2: Padrões de correlação. Elaborado por Gujarati and Porter (2011) e adaptado Henri (1978).

5.1.4 Medidas de Precisão

5.1.5 Medidas de consistência

5.2 Teste de hipóteses e Análise de Variância

Chapter 6

Pré-processamento

Para o profissional que trabalha com Aprendizado de Máquina ou outras áreas, embora exigindo boa parte do tempo nesta etapa, é uma das mais importantes. O pré-processamento é um conjunto de atividades que buscam preparar, organizar e estruturar o banco de dados (*dataset*) para que possa trabalhar com os dados. Ela torna a informação de seus dados mais consistentes, com organização rígida e geralmente classificados de acordo com o seu formato (caracteres, binários, numéricos, etc). Podemos dizer que ele é um conjunto de técnicas do campo de **Mineração de dados (*Data mining*)**, uma outra área além de Inteligência Artificial (que engloba Aprendizado de Máquina) – que já é grande por si só -, que trata-se de uma outra dimensão de estudos e metodologias, isso sem falarmos de outros campos além destes dois. Neste tópico, vamos abordar algumas delas que são muito utilizadas nesta área. Note que em todos os procedimentos de Aprendizado de Máquina existe inúmeras metodologias para serem aplicadas em cada etapa e, de acordo com o interesse do pesquisador, pode ser utilizado diferentes estratégias com diferentes combinações. Não há uma só receita de bolo: sabemos que precisamos extrair dados, pré-processá-los (aplicar uma(s) estratégia para analisar, classificar os atributos, eliminar os redundantes, preencher ou eliminar os faltantes), desenvolver seus modelos de Aprendizado de Máquina, treiná-los e por fim, avaliar todo o seu modelo e cada etapa se encontra com diversos métodos. É... Não é fácil, mas todo esse procedimento é fundamental para que se obtenha um modelo adequado. Portanto nesta seção busquei separar em alguns tópicos para facilitar a compreensão, porém entenda que **TODAS** as metodologias e estratégias podem ser combinadas e estão entrelaçadas. É como vários conjuntos em um *Diagrama de Venn* que estão dentro do Pré-processamento que está dentro de Mineração de dados e que está interseccionada com Aprendizado de Máquina (dentro de IA).

Não se assuste: No último capítulo deste livro estará um diagrama e uma explicação mais “cronológica” de todo esse cosmos, com suas “gáxias” e sistemas “solares” de conteúdo.

6.1 Dados faltantes e a Limpeza de dados

Durante o desenvolvimento destes modelos é comum se deparar com dados faltantes em seu banco de dados e que podem ser ocasionadas por razões diversas como não preenchimento cadastral, problemas de armazenamento de dados ou até mesmo situações aleatórias não identificadas. A escolha da forma de tratar esses dados faltantes é fundamental para o modelo. Os valores faltantes total quando todas as informações são perdidas ou parcial quando somente uma parte delas são perdidas

(Little and Rubin, 2019), descrevem que os motivos de aparecimento de dados faltantes são comumente classificados em:

1. **Missing Completely at Random (MCAR)**: neste caso, as observações faltante surgiram de maneira aleatória, portanto as razões para as perdas não são relacionadas às respostas do sujeito. O único problema gerado pelos dados faltantes é a perda de poder da análise a ser realizada. Por exemplo, um jovem que deixou de responder uma questão de sua prova sem querer, sem motivo algum.
2. **Missing at Random (MAR)**: os dados faltantes dependem das variáveis preenchidas e, portanto, podem ser totalmente explicadas pelas variáveis presentes no conjunto de dados. É possível não viesar a análise, considerando as informações que causam estes dados faltantes. Como por exemplo uma pesquisa elaborada por uma universidade com a finalidade de analisar a renda das mulheres em sua cidade porém não possui recursos financeiros suficiente para entrevistar todas as mulheres. A pesquisa é respondida por uma parcela de mulheres na cidade e todas as envolvidas estão com os dados completamente observados, seria analisado uma amostra aleatória de mulheres.
3. **Missing Not at Random (MNAR)**: nesta situação os dados faltante são gerados de forma não mensurável, isto é, de eventos que o pesquisador não consegue observar e não tem controle. É o pior caso e algumas vezes, é necessário técnica mais robustas. Em geral, dados situados nos extremos da distribuição são mais propensos a serem faltantes (muito baixos ou altos em relação ao padrão da amostra).

6.1.1 Tratamento de dados faltantes

Existem diversas metodologias de tratamentos em dados faltantes. Quando os dados são faltantes em um conjunto de dados, existem cinco grandes categorias de tratamento de análise que um pesquisador deve escolher. Como mencionado anteriormente e ainda reforço, a escolha do tratamento de análise de dados faltantes tem implicações importantes para a acurácia e o viés das estimativas.

Table 6.1: Metodologia de dados faltantes (de Andrade et al., 2019).
Determinados termos estão na seção 1.2 e alguns outros serão apresentados ao longo do livro .

Técnicas de Análise para dados faltantes	Definições	Maiores Problemas
Listwise Deletion	Exclui todos os casos para os quais alguns dados estão faltando	Descarta dados de respondentes com respostas parciais. Menor amostra, menor potência. Viés em MAR e MNAR.
Pairwise Deletion	Calcula as estimativas (médias, EP, correlações) usando todos os casos disponíveis com dados relevantes para cada estimativa.	Diferentes correlações representam misturas de subpopulação. Às vezes, a matriz de covariância não é definida positiva. Viés em MAR e MNAR. Nenhuma amostra faz sentido para a matriz de correlação (EP impreciso).
Imputação Simples	Preenche cada valor faltante, por exemplo média, por regressão, etc.	A imputação média (entre casos) e a imputação por regressão são ambas tendenciosas sob MCAR! Nenhuma amostra faz sentido para a matriz de correlação (EP impreciso). EP's subestimados se você tratar o conjunto de dados como completo.

Técnicas de Análise para dados faltantes	Definições	Maiores Problemas
Máxima Verossimilhança (MV)	Estima diretamente os parâmetros de interesse a partir de uma matriz de dados incompleta; ou calcula estimativas como média, desvio padrão, ou correlação usando algum algoritmo.	Não-viesada sob MCAR e MAR. Melhora à medida que adiciona mais variáveis ao modelo de imputação. Número de variáveis deve ser menor que 100. EP'S preciso para FIML. para o algoritmo EM, nenhuma amostra faz sentido para a matriz de correlação (EP impreciso).
Imputação Múltipla (IM)	Imputa valores faltantes várias vezes, cria-se m conjuntos de dados completamente imputados. Executa a análise em cada conjunto de dados imputado. Combina os m resultados para obter estimativas de parâmetros e erros padrão.	Imparcial sob MCAR e MAR. Melhora à medida que adiciona mais variáveis ao modelo de imputação. O número de variáveis deve ser menor que 100. EP's precisos. Fornece estimativas ligeiramente diferentes a cada vez que analisa os dados. Em Equações Estruturais, piora a convergência.

- **Listwise deletion:** exclui todos os casos para os quais alguns dados estão faltando. A eliminação dos casos frequentemente reduz muito o tamanho da amostra e o poder estatístico do teste de hipóteses. Importante o pesquisador se atentar que mesmo quando o poder do teste parece adequado, este método pode produzir estimativas de parâmetros tendenciosas sob dados faltantes sistemático (MAR e MNAR). O *listwise deletion* restringe a população-alvo do estudo, assim em geral quase nunca se utiliza esse procedimento. Uma vez que ele descarta dados que custaram tempo, disponibilidade dos participantes e até mesmo recursos financeiros, a eliminação desses participantes da pesquisa pode violar o princípio ético da pesquisa (Rosenthal, 1994).

Resumo geral: elimina todos os casos que possuem dados faltantes em sua pesquisa.

- **Pairwise deletion:** este método tenta minimizar a perda que ocorre em *Listwise deletion*. Como exemplo a matriz de correlação. Uma correlação como explicada em 1.2, mede a força da relação entre duas variáveis. Para cada par de variáveis para os quais os dados estão disponíveis, o coeficiente de correlação indicará a força. Em *Listwise* será o mesmo tamanho para todas as correlações excluindo toda observação faltante, em *Pairwise deletion* irá variar. Ela exclui apenas os casos que não tem respostas completas dentro da observação, aproveitando o maior número de casos possíveis.

Resumo geral: ao invés de eliminar as observações (coluna ou linha inteira da matriz) com dados faltantes, como *listwise deletion*, este método elimina apenas os casos que não tem respostas completas nas combinações das observações, aproveitando o maior número possível.

- **Imputação simples:** envolve o preenchimento de cada dado faltante com uma suposição de qual deve ser o valor que está faltando no conjunto de dados. Os exemplos mais comuns de imputação simples são: imputação pela média - substituição de cada valor faltante pela média do grupo para a variável correspondente; imputação hot deck* - substituição de cada dado faltante por um valor “doador” que possui um escore similar em outras variáveis; e imputação por regressão – substituindo cada valor faltante por um valor predito com base em um modelo de regressão múltipla (será explicado conceito de regressão posteriormente), obtido a partir dos valores observados (de Andrade et al., 2019). A maioria das técnicas de imputação simples é tendenciosa. Por exemplo, a imputação pela média insere uma média constante para cada valor faltante, as estimativas da variância e da correlação serão tendenciosas – mesmo que o mecanismo de dados faltantes seja completamente aleatório (MCAR). A imputação por regressão leva à subestimação da variância e superestimação da correlação (pois os valores imputados estarão exatamente na linha de regressão). Pode-se melhorar ao caso de regressão adicionando um termo de erro aleatório aos valores imputados (regressão estocástica), no entanto, ainda são imprecisas. Ao caso dos testes de hipóteses, não estima com precisão o erro padrão (de Andrade et al., 2019).

Resumo geral: envolve o preenchimento de cada dado faltante com uma “boa adivinhação” de qual deve ser o valor que está faltando no conjunto de dados, sendo essa estimação de acordo com o pesquisador e sua pesquisa (média, regressão, etc).

- **Imputação múltipla (IM):** cada valor faltante é substituído por dois ou mais valores imputados e ordenados a fim de representar a incerteza sobre qual valor imputar, permitindo que as estimativas das variâncias estimadas sejam calculadas com dados completos (Rubin, 2004). Assim, m imputações atribuídas a cada valor faltante gera n conjuntos de dados completados que são analisados inerente aos valores observados da amostra.

Muitos utilizam este método, visto que aumenta a eficiência de estimação, facilita o estudo direto da sensibilidade de inferências, abrange uma variedade de análises e geralmente válidas por incorporar incertezas devido à falta de dados. Tornando-os mais eficientes que a imputação simples, porém mais trabalhosa e ocupa mais espaço de armazenamento. Em desvantagem desse método, pode surgir discrepância na variância quando se admite pressupostos equivocados (modelo escolhido não consistente com os dados), com isso um m pequeno se torna mais adequado com menor gravidez. Uma das características mais importantes desse método é que os valores faltantes para cada envolvido é predito a partir de seus próprios valores observados, com o ruído aleatório adicionado para preservar uma correta quantidade de variabilidade nos dados imputados (Schafer and Graham, 2002).

Schafer (1999) recomenda que a quantidade necessária de imputações para que a estimativa de conjunto de dados tenha relativa eficiência, com a seguinte equação:

$$RE = \sqrt{1 + \frac{\lambda}{m}} \quad (6.1)$$

onde, m é a quantidade do conjunto de dados completados e λ é a taxa de informação - caso fosse 50% dos dados faltantes, $\lambda = 0,5$.

Claro que o método para mensurar a quantidade necessária **varia de acordo com o tema da pesquisa e a escolha do pesquisador**. Dependendo área que o pesquisador está interessado, pode-se haver outras recomendações para mensurar a quantidade.

A IM é composto basicamente por três passos (Assunção, 2012):

1. **Imputação dos dados:** são gerados m bancos de dados completos através de técnicas adequadas que devem levar em conta ao máximo a relação entre os dados faltantes e os observados. Existe diversos métodos que podem ser utilizadas para este primeiro passo, um dos mais utilizados atualmente é o método de **regressão linear bayesiana** - ao caso de não entender o que são as técnicas de Regressão linear nem de Bayes, as seções XXXXXXXXXXXX instruem.

Este método tem como resposta a variável que possui dados faltantes (Y) e como variáveis preditoras são utilizadas as demais variáveis presentes (X_1, X_2, \dots, X_k), com k número de preditoras. Na abordagem Bayesiana, a regressão linear é formulada através de distribuições de probabilidade ao invés da abordagem clássica. Seu modelo será:

$$Y_i \sim N(\beta^T X_k, \sigma^2 I)$$

A variável dependente Y_i é gerada a partir de uma Distribuição Normal (Gaussiana) 1.2 caracterizada pela média e variância (σ^2). A média é o produto entre

os parâmetros β e variáveis independentes X_k . O objetivo deste método é determinar a distribuição posterior para os parâmetros do modelo ao invés de encontrar um único valor. A resposta e seus parâmetros são gerados por meio de uma distribuição de probabilidade.

Para encontrar as distribuições dos parâmetros do modelo, a inferência bayesiana utiliza o Teorema de Bayes para combinar informações prévias ao experimento e dados de amostra com o objetivo de deduzir as propriedades sobre um parâmetro de interesse a partir dos dados de entrada X_k e de saída Y . A aplicação de Bayes neste contexto seria:

$$P(\beta|y, X) = \frac{P(y|\beta, X)P(\beta|X)}{P(y|X)} \quad (6.2)$$

onde $P(\beta|X)$ reflete a incerteza de β . Qualquer informação que se tenha inicialmente sobre o parâmetro é tratado como ela (pode ser utilizada como não informativa). Em $P(y|\beta, X)$ é a verossimilhança que diz respeito a distribuição característica dos dados (interpretada como no caso clássico). O denominador $P(y|X)$ é tratada como uma constante de normalização para a equação e reflete a probabilidade que pode-se obter qualquer dado.

Ressalto que existe diversos métodos nesta primeira etapa e recomendo o leitor interessado, buscar outras literaturas.

2. **Análise dos bancos de dados gerados pelo passo 1:** ao criar o conjunto de dados imputados, é importante fazer uma análise separadamente para cada um dos m banco de dados da mesma forma como tradicionalmente se faz, o modelo pode variar de acordo com o pesquisador - são apresentadas na seção SEIS AQUI COLOCAR A SEÇÃO DEPOIS.
3. **Combinar os resultados:** com as análises realizadas, precisa-se combinar os resultados apropriados para obter a inferência da imputação repetida. Por meio do passo 2, obtém-se estimativas para o parâmetro de interesse D . Estas estimativas podem ser qualquer medida escalar como médias, variâncias, correlações, coeficientes de regressão por exemplo. A estimativa D será a combinação será a média das estimativas individuais.

$$\bar{D} = \frac{1}{m} \sum_{s=1}^m \hat{D}_s \quad (6.3)$$

Em seguida, a variância combinada é calculada:

$$T = \bar{E} + (1 + \frac{1}{m})F \quad (6.4)$$

em que $\bar{E} = \frac{1}{m} \sum_{s=1}^m E_s$ é a média das variâncias que preserva a variabilidade natural (E) do parâmetro de interesse nos m banco de dados e $F = \frac{1}{(m+1)} \sum_{s=1}^m (\hat{D}_s - \bar{D})^2$ o componentes que estima a incerteza causada pelos dados faltantes. Se F for muito pequeno as estimativas dos parâmetros são muito semelhantes, com menos incerteza. Do contrário as incertezas variam muito.

Resumo geral: a imputação múltipla executa uma rotina de imputação simples repetidamente (múltiplas associações sobre os valores plausíveis) e consegue estimar sem vies o erro padrão. Ocorre as imputações muitas vezes contabilizando a imprecisão de cada imputação.

- **Método de máxima verossimilhança (EM - Expectativa-maximização):** proposto por Fisher (1912), é um método paramétrico (ver 1.2) que parte do princípio de especificar como a função de verossimilhança (ver 1.2) deveria ser utilizada como um instrumento de redução de dados Casella and Berger (2010). Este método consiste na escolha do conjunto de valores para os parâmetros que torne um máximo a função de verossimilhança. A inferência de verossimilhança pode ser considerada como um processo de obtenção de informação sobre um vetor de parâmetros θ , a partir do ponto x do conjunto amostral, por meio da função de verossimilhança. Vários vetores podem produzir a mesma verossimilhança, reduzindo a informação de θ (Cordeiro, 1999).

O objetivo é encontrar uma estimativa do parâmetro θ , $\hat{\theta}$, que maximize a verossimilhança. Portanto, utiliza-se o conceito de derivada (diferenciação) e igualamos a zero (Bolfarine and Sandoval, 2001).

$$L'(\theta; x) = \frac{\delta L(\theta; x)}{\delta \theta} = 0 \quad (6.5)$$

Para inferir se é um ponto máximo, aplica-se a segunda derivada e verificar se o resultado é menor que zero (Bolfarine and Sandoval, 2001).

$$L''(\hat{\theta}; x) = \frac{\delta^2 \log L(\theta; x)}{\delta \theta^2} < 0 \quad (6.6)$$

Com algoritmo EM (Expectativa-maximização), por Dempster et al. (1977) é um procedimento que realiza a estimativa dos parâmetros (vetor de médias e a matriz de covariância) por meio da máxima verossimilhança em conjuntos amostrais incompletos (dados faltantes) e pode ser utilizado como uma ferramenta para inserção de dados. Por um processo iterativo, na etapa E(Estimação/Esperança) se estima os dados faltantes para completar a matriz dos dados, no caso calcula-se a esperança condicional (média condicional) da função de log-verossimilhança; no passo M (Maximização), com os dados

completados, encontra-se um $\hat{\theta}$ que maximiza a esperança condicional da log-verossimilhança e então seu resultado é usado para fazer a inferência no passo E e assim sucessivamente até que o algoritmo processado tenha convergido, ou seja, a diferença entre o valores da verossimilhança dos dados incompletos na k -ésima e na $(k + 1)$ -ésima iteração seja tão pequena (Enders, 2010, ; Pereira, 2019).

Resumo geral: o algoritmo EM, faz a etapa E com a função de verossimilhança para encontrar um valor médio e preencher os dados faltantes, faz a etapa M utilizando a máximização de verossimilhança para encontrar um valor médio com o menor erro possível e continua, a partir do resultado do segundo passo, sucessivamente até convergir no melhor valor e menor erro possível (global) para preencher os dados faltantes.

Além de dados faltantes, é possível lidarmos com grande volume de dados. Por isso, o processamento computacional se torna cada vez mais complexo e para aumentarmos a eficiência e reduzir os custos usamos o processo de redução de dados ou a hierarquização para separarmos os conjuntos a serem estudados. Pode-se por meio de **Agregação de cubo de dados** (atividade de construção de um cubo de dados) que apesar de gerar maior necessidade de armazenamento, permite um processamento mais rápido por não necessitar varrer toda a base em busca de determinado valor. A **Seleção de subconjuntos de atributos** para utilizar os atributos altamente relevantes em detrimento dos menos relevantes (como por exemplo verificar pela significância). Ou também **reduzir a numerosidade** ou **dimensionalidade** que permitem que os dados seja estimados por alternativas de representação de dados menores e compactados e alguns métodos para hierarquizar as variáveis. Na seção de XXXXXXXXXXXX serão apresentados as principais estratégias.

6.1.2 *Outlier*

Um *outlier* é um valor que se encontra distante da normalidade e que provavelmente causará anomalias nos resultados obtidos, pois pode viesar negativamente todo o resultado de uma análise e que seu comportamento pode ser justamente o que está sendo procurado. São basicamente dados que se diferenciam drasticamente dos outros, conhecidos como anomalias, pontos fora da curva, dados discrepantes, ruídos, e que estão fora da distribuição normal.

Pode-se verificar dados incomuns apenas verificando a tabela, mas dependendo do tamanho de seu banco de dados não é uma boa recomendação. Uma das melhores maneiras de identificarmos dados *outliers* é utilizando gráficos. Ao plotar um gráfico o analista consegue verificar que existe algo diferente. Como exemplo, um estudo no sistema de saúde brasileiro pela AQUARELA (2017) utilizando dados da prefeitura de Vitória no Espírito Santo, analisando fatores que levam as pessoas a não comparecerem em consultas agendadas no sistema público de saúde da cidade. Padrões encontrados de que mulheres comparecerem muito mais que os homens e crianças faltam poucos às consultas, porém,

uma senhora *outlier*, com 79 anos agendou uma consulta e com 365 dias de antecedência apareceu à consulta. Neste caso, convém ser estudado o *outlier* pelo comportamento trazer informações relevantes que podem ser adotadas para aumentar a taxa de assiduidade nos agendamentos. *Outlier* do caso indicado pela seta vermelha 6.1.

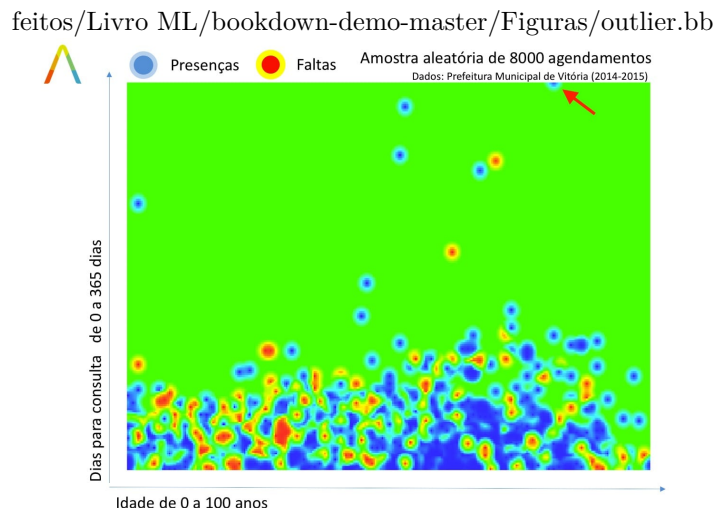


Figure 6.1: “Gráfico de estudo no sistema de saúde apresentando *outlier* (AQUARELA, 2017).”

Por diversos motivos pode ocorrer de ter presença de *outlier* nos dados e podem viesar negativamente todo resultado de uma análise e seu comportamento pode muitas vezes ser o que justamente o pesquisador está procurando. Há possibilidade do *outlier* ser importante para o pesquisador entender o por que da anomalia estar acontecendo, ou para identificar algum dado extraído erroneamente, por exemplo.

Uma maneira mais complexa e muito precisa, é de identificá-los através de análise dos dados. Encontrando a distribuição estatísticas que mais e aproxima à distribuição dos dados e utilizar métodos estatísticos para detectar as anomalias. Como por exemplo o uso de histograma e a distribuição normal para verificar os dados que estão dentro e fora do intervalo de confiança (ver 1.2 Distribuição normal).

6.2 Transformação de dados

6.2.1 Tipos de *datasets*

A escolha das medidas estatísticas para sua análise ou modelo de Aprendizado de Máquina dependem muito dos tipos de dados das variáveis em observação.

Estes tipos de dados podem ser numéricos (como uma sala de aula, com alunos que variam sua altura de 1,51 metros a 1,98 metros) e categórico (como uma classificação num hospital de pacientes doentes ou não doentes), embora esses dois tipos podem ser subdivididos como números inteiros e ponto flutuante para variáveis numéricas e booleano, ordinal ou nominal para variáveis categóricas.

As subdivisões mais comuns são: - Variáveis Numéricas: 1. Variáveis inteiras (exemplo: 1, 2, 3, ..., n); 2. Variáveis de ponto flutuante (parte fracionária, por exemplo: 1,17; 0,10; 47,2).

- Variáveis categóricas:
 1. Variáveis booleanas (dicotômicas, binárias: Verdadeiro e Falso).
 2. Variáveis ordinais (1º, 2º, 3º, etc).
 3. Variáveis nominais (não possuem ordenação como por exemplo, cor dos olhos: azuis, castanhos, pretos e verdes).

Importante ressaltar que quando trabalhamos dentro da programação, possuem mais tipos além de *int* (numéricos inteiros) *char* (caracteres) e *float* (pontos flutuantes), como o *double* que armazena números com ponto flutuantes com precisão dupla com o dobro da capacidade de *float*, *string* como cadeia de caracteres.

Muitos algoritmos possuem a limitação de trabalhar somente com atributos qualitativos (variáveis categóricas), com isso muitas vezes é necessário aplicar algum método capaz de transformar um atributo quantitativo em um atributo qualitativo (faixas de valores). Uma estratégia que cresce ao longo do tempo é o processo de **discretização** que transforma atributos contínuos em atributos discretos como por exemplo, dividir alturas entre menor que 1,70 metros e maior igual que 1,70 metros. Dependendo do estudo pode ser adequado, embora o pesquisador precisa tomar muito cuidado pois é provável que possa perder algumas informações. De mesmo modo, é possível transformar variáveis categóricas em numéricas, como por exemplo classificar tamanhos como pequeno = 1, médio = 2 e grande = 3 possibilitando por meio do mapeamento manter a ordem dos valores (Batista et al. (2003)).

É bem comum estes tipos de tratamento de dados ao caso de datas, como trabalhos que aplicam-se **séries temporais** em que o pesquisador precisa estudar a sazonalidade de algum objeto de estudo. A soja por exemplo pode-se analisar sua tendência ao longo dos anos, mas quando tratamos os dados e analisamos em outro período podemos verificar que possui sazonalidades em sua produção. Em análises para investimentos também, atentar o comportamento mensal e diário das ações de uma empresa, muitas vezes está com tendência de alta num âmbito mensal, porém ao analisar diariamente é possível que esteja em baixa.

Para facilitar a compreensão, considere a série temporal *AirPassengers* que representa o número de passageiros mensalmente em uma empresa de transporte aéreo ao período de 1949 a 1960 (Box and Jenkins, 1976).

Para o campo de transformação de dados e séries temporais, ao leitor que pre-

feitos/Livro ML/bookdown-demo-master/Figuras/airpassengers.bb

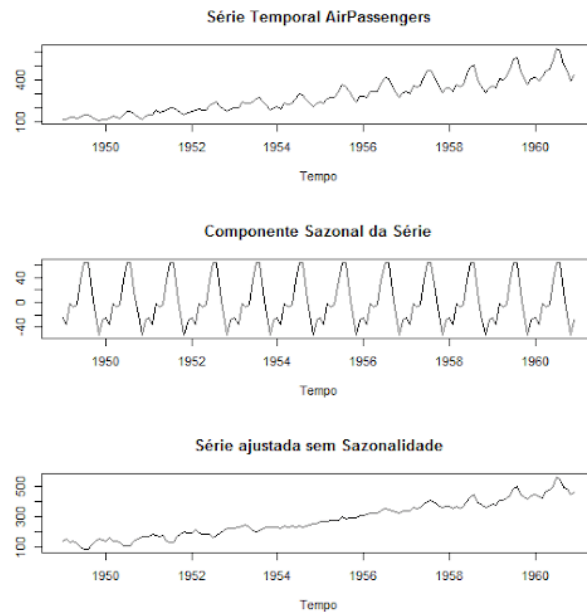


Figure 6.2: “Número de passageiros tratados mensalmente (Box and Jenkins, 1976).”

tende ir mais a fundo nestes outros “galhos” de estudos. Recomendo buscar outras literaturas que tem como foco este temas. Em discretizações por exemplo, Dougherty et al. (1995) e Garcia et al. (2012) abordam diversos métodos que podem agradá-lo.

6.2.2 Normalização e padronização

Muitos conjuntos de dados apresentam atributos contínuos que espalham-se em diferentes faixas de valores ou possuem distintas variações, devido às suas naturezas ou escalas em que foram medidas. Estas diferenças podem ser fundamentais e precisam ser levadas em conta (CARVALHO et al., 2011). Em situações também para validarmos a análise variância precisa-se dos requisitos de atiditividade, independência, normalidade e homogeneidade de variâncias - será apresentada em ANOVA seção XXXXXXXX. Quando alguma das características mencionadas acontece ou não verifica seus requisitos o pesquisador, antes de fazer uma análise não-paramétrica (1.2), pode-se transformar seus dados (Banzatto and Kronka, 1992).

1. **Normalização por reescala:** através de um valor mínimo e um máximo, gera um novo intervalo onde os valores de um atributo estão contidos. Um intervalo entre 0 e 1.

$$x_{ij} = \frac{x_{ij} - \min_j}{\max_j - \min_j} \quad (6.7)$$

sendo x_i a observação de ordem i , \min_j e \max os valores mínimos e máximos do atributo j respectivamente.

2. **Transformação de raiz quadrada:** frequentemente utilizada para dados de contagens que geralmente segue uma distribuição de Poisson (1.2), onde a média é igual à variância (Banzatto and Kronka, 1992).

$$\sqrt{x_i} \quad (6.8)$$

sendo x_i representando as observações do banco de dados. Quando ocorrem zeros ou valores baixos (menores que 10 ou 15), recomenda-se $\sqrt{x+0,5}$ ou $\sqrt{x+1,0}$ (Banzatto and Kronka, 1992).

3. **Transformação angular:** recomenda-se para dados expressos em porcentagens, que geralmente seguem a distribuição binomial (1.2). Atualmente existe tabelas apropriadas para essa transformação (Banzatto and Kronka, 1992). Segundo Banzatto and Kronka (1992) porcentagens entre 30% e 70% ou as porcentagens são resultantes da divisão dos valores observados nas parcelas por um valor constante tornam-se desnecessárias e pode-se analisar diretamente os dados originais, mas atente-se pois algumas vezes variar essas exceções de acordo com sua área e pesquisador que a propõe.

$$\text{arc sen} \sqrt{\frac{x}{100}} \quad (6.9)$$

4. **Transformação logaritmica:** quando verificada determinada proporcionalidade entre as médias e desvios padrões dos diversos tratamentos. É geralmente utilizada para problemas de assimetria (1.2). Em casos, por exemplo, tratamentos com amplitude alta como uma população numerosa que varia de 1.000 a 10.000 indivíduos ou tratamentos de baixa amplitude de 10 a 100 indivíduos. Esta transformação pode ser útil.

$$\log(x) \text{ ou } \ln(x) \quad (6.10)$$

Uma vez transformados os dados em logaritmos, a soma de dados logarítmicos não tem o mesmo valor que a soma de seus antilogaritmos, mas representa o produto destes.

5. **Padronização:** é um método muito utilizado por diversas áreas de pesquisa. Neste caso diferentes atributos podem abranger diferentes intervalos, porém possuir os mesmos valores para alguma medida de posição e de variação (CARVALHO et al., 2011). Imagine você como economista interessado em avaliar o desempenho da produção de soja com as variáveis econômicas e monetárias o Brasil e possui as seguintes variáveis: produção de soja anual medida em milhares de toneladas, taxa básica de juros SELIC medida em porcentagem, receita média anual em milhares de reais, área plantada de soja medida em hectares. Já podemos perceber que todos possuem medidas e grandezas bem diferente uma das outras. Este o propósito da padronização, deixar com que todas as variáveis tenham uma medida em comum.

$$Z_{ij} = \frac{x_{ij} - \bar{X}}{\sigma_j} \quad (6.11)$$

em que \bar{X}_j e σ_j representam a média e o desvio padrão do atributo j respectivamente. Após a transformação todos os atributos terão a média zero e desvio-padrão unitário.

Caso transformado seu banco de dados e seu banco de dados apresentarem uma distribuição contínua não-normal, ou não-homogênea ou não-aditiva, não há outra alternativa senão utilizar a estatística não-paramétrica.

Resumo geral: Muitos conjuntos de dados apresentam atributos contínuos que espalham-se em diferentes faixas de valores ou possuem variações diferentes, por motivo de suas naturezas ou escalas medidas. Estas diferenças podem ser muito importantes e precisam ser levadas em conta para não causar erros em sua pesquisa. Para isso usam-se alguns métodos para transformar seus dados para que possam ser trabalhados, apresentados os principais neste livro. Em

situações para fazermos análise variância precisa-se também ser transformado seus dados caso não cumpra seus requisitos. Caso o problema ainda persistir, precisa-se utilizar estatística não-paramétrica.

6.3 Features Selection - Seleção de atributos (SA)

Uma literatura que achei bastante interessante foi Parmezan et al. (2012). Seguindo sua estrutura a respeito de Seleção de atributos. Podemos definir SA como a determinação de um subconjunto ótimo de atributos, partindo de algum critério ou medida de importância, que representa a informação importante dos dados (Parmezan et al., 2012). Extraímos um subconjunto de P atributos a partir de um conjunto original de N atributos, sendo $P \leq M$ (Parmezan et al., 2012; Liu and Motoda, 1998; Lee, 2005). A cada conjunto de dados com M atributos, existem 2^M subconjuntos de atributos candidatos (Langley et al., 1994).

Existem diversas metodologias para selecionarmos os atributos que podem variar em sentido de buscas e estratégias para a seleção. Repare que os tópicos mencionados anteriormente também são utilizados para remoção e seleção, foi fragmentado apenas para facilitar a compreensão.

O “sentido de busca” influencia na determinação do(S) ponto(s) de partida no espaço de busca, ou seja, na direção em que a busca será realizada e os operadores que serão utilizados. Elas são categorizadas, seguindo Parmezan et al. (2012) e Liu and Motoda (2008), em:

- **Forward Selection - Seleção para Frente:** o estado inicial é estabelecido como vazio (subconjunto vazio de atributos), e os atributos são incluídos um por vez;
- **Backward Elimination - Eliminação por Trás:** o ponto de partida é iniciado com o conjunto de todos os atributos (completo), tais quais são removidos sucessivamente;
- **Bidirectional Search - Pesquisa Bidirecional:** como o próprio nome diz, duas buscas são processadas simultaneamente. Ambas terminam quando atingem o centro do espaço de busca, ou quando uma das buscas encontra os melhores atributos antes de alcançar o centro do espaço de busca;
- **Random Search - Pesquisa Aleatória:** com o propósito de evitar que a busca fique restrita a ótimos locais. Não tem uma direção específica para buscar, pois o ponto de partida da busca e o modo de adicionar ou remover atributos são decididos aleatoriamente.

Além dos sentidos de busca, existem diversas abordagens que avaliam subconjuntos de atributos e que podem remover tanto atributos irrelevantes quanto

redundantes (Parmezan et al., 2012; Liu and Motoda, 2008). A seguir, as principais abordagens:

- **Filter - Filtro:**

Com a finalidade de filtrar atributos não importantes, essa abordagem é feita antes da construção dos modelos. A ideia é simplesmente receber como entrada o conjunto de exemplos descrito utilizando somente o subconjunto de atributos importantes identificados. Ela ocorre antes do aprendizado de máquina (John et al., 1994) e utiliza-se métodos estatísticos diversos para esta seleção, como por exemplo árvores de decisão ou as “medidas de importância” que são apresentadas na próxima seção.

- **Wrapper - Empacotar:** ocorre também externamente ao algoritmo de aprendizado. Este método gera um subconjunto candidato de atributos, executa o algoritmo de aprendizado considerado somente esse subconjunto selecionado de treinamento e avalia a precisão desse classificador. Repete-se esse processo para cada subconjunto de atributos até buscar um bom modelo. Como exemplo temos a análise por árvores de decisão e florestas aleatórias (serão apresentadas mais a frente). Tem como desvantagem o custo operacional desta abordagem. Exemplo de aplicações: *Naive Bayes* e Máquina de vetores de suporte para classificação.

- **Embedded - Embutida:** é realizada internamente pelo próprio algoritmo de extração de padrões. Esta estratégia seleciona o subconjunto de atributos no processo de construção do modelo de classificação, durante a fase de treinamento, e geralmente são específicos para um dado algoritmo de aprendizado. A principal diferença dos métodos do tipo *embedded* e *wrapper*, é que em *embedded* depende em relação a um modelo preditivo específico, assim não permite a sua implementação em combinação com outros modelos (Souza, 2014).

Observação e resumo geral: Note que o que muitas vezes confunde o leitor é o excesso de categorias - que ironicamente tem o propósito de organizar e facilitar. Basicamente são estratégias diferentes com sentidos diferentes de se iniciar a busca de atributos que podem ser irrelevantes ou relevantes: antes de criar um modelo de Aprendizado de máquina; usa-se um modelo de aprendizado para selecionar os atributos antes de iniciar uma etapa de análise [pode-se até mesmo realizar outro algoritmo de aprendizado após este algoritmo de seleção] ou a própria seleção com a análise [mesmo algoritmo para selecionar e concluir]). Quando misturamos esta estratégia, denominamos de **híbridos**.

feitos/Livro ML/bookdown-demo-master/Figuras/diferenssa.bb

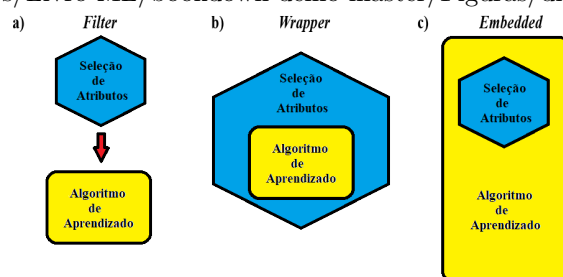


Figure 6.3: “Diferença de *Filter*, *Wrapper* e *Embedded* respectivamente (modificado de Covões (2010)).”

Chapter 7

Algoritmos de Aprendizagem - Parte I

Existe uma infinidade de algoritmos utilizados em machine learning, cada um com uma finalidade específica. Há também características que podem inviabilizar a escolha do modelo mais preciso para determinado problema, como a utilização alto poder computacional.

7.1 Naive Bayes

Antes de falarmos sobre este algoritmo, vamos para o conceito matemático. Em (1.2) tratamos do Teorema de Bayes para n atributos. Colocando-o como probabilidade condicional:

$$p(A|B_1, \dots, B_n) = p(A)p(B_1|A)p(B_2|A, B_1)p(B_3|A, B_1, B_2)\dots p(B_n|A, B_1, B_2, \dots, B_{n-1}) \quad (7.1)$$

Assumindo que cada atributo B_i é condicionalmente independente de todos os outros B_j para $j \neq i$ e $p(B_i|A, B_j) = p(B_i|A)$ o modelo poderá ser expresso como:

$$p(A_k|B_1, \dots, B_n) = p(A_k)p(B_1|A_k)p(B_2|A_k), \dots = p(A_k) \prod_i^n p(B_i|A_k) \quad k \in 1, \dots, k \quad (7.2)$$

Por fim para podermos classificar, aplicamos argumento de máxima para otimizarmos a função, assim obtém-se o classificador de Naive Bayes:

$$\text{classificador } \hat{y} = \underset{k \in 1, \dots, k}{\operatorname{argmax}} p(A_k) \prod_{i=1}^n p(B_i|A_k) \quad (7.3)$$

Lembrando que para cada atributo, a sua distribuição de probabilidades é assumida como normal.

O Naive Bayes é uma técnica de classificação baseado no teorema de Bayes com uma suposição de independência entre os preditores, ou seja, este classificador assume que a presença de uma característica particular em uma classe não está relacionada com a presença de qualquer outro fator. Por exemplo, uma fruta verde, redonda e com um tamanho de diâmetro X pode ser uma melancia, porém mesmo que estas variáveis dependam uns dos outros e de outras características, todas estas propriedades contribuem de forma independente para a probabilidade de que seja uma melancia. Este modelo é muito utilizado devido que é fácil de construir e particularmente útil para grandes volumes de dados. Porém a própria independência entre os preditores a torna desvantajosa na prática e caso haja variáveis categóricas num conjunto de dados de teste que não forem treinadas, o modelo não irá estimar estas novas variáveis.

7.1.1 Exemplo

No diagnóstico de uma nova doença e que foi feito testes em 100 pessoas aleatórias (exemplo de Orgânica Digital (2019)).

Após coletarmos a análise, descobrimos que das 100 pessoas, 20 possuíam a doença (20%) e 80 pessoas estavam saudáveis (80%), sendo que das pessoas que possuíam a doença, 90% receberam o resultado positivo no teste da doença, e 30% das pessoas que não possuíam a doença também receberam o teste positivo. Caso uma nova pessoa realizar o teste e receber um resultado positivo, qual a probabilidade de ela realmente possuir a doença?

feitos/Livro ML/bookdown-demo-master/Figuras/bayes.bb

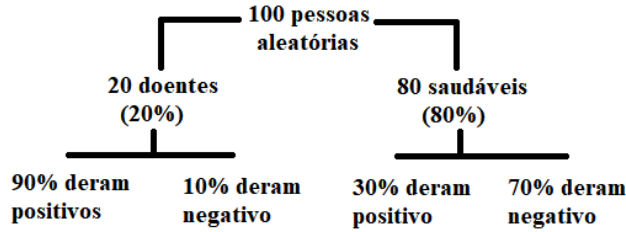


Figure 7.1: Dados coletados de uma amostra de 100 pessoas aleatórias.

Com o algoritmo de Naive Bayes, buscamos encontrar uma probabilidade da pessoa possuir a doença dado que ela recebeu um resultado positivo, multiplicando a probabilidade de possuir a doença pela probabilidade de “receber um

resultado positivo, dado que tem a doença”. De mesmo modo verificar a probabilidade de não possuir a doença dado que recebeu um resultado positivo.

Ou seja, ao caso de ter a doença dado que o resultado deu positivo:

$$P(doena|positivo) = 20\%.90\%$$

$$P(doena|positivo) = 0,2 * 0,9$$

$$P(doena|positivo) = 0,18$$

Para o caso de não ter a doença, dado que deu positivo:

$$P(no\ doena|positivo) = 80\%.30\%$$

$$P(no\ doena|positivo) = 0,8 * 0,3$$

$$P(no\ doena|positivo) = 0,24$$

Após isso precisamos normalizar os dados, para que a soma das duas probabilidades resulte 1 (100%). Como vimos em pré-processamento 6, a **Normalização por reescala** por meio de um valor mínimo e um máximo, gera um novo intervalo onde os valores de um atributo estão contidos. Um intervalo entre 0 e 1. Portanto, dividimos o resultado pela soma das duas probabilidades.

$$P(doena|positivo) = 0,18/(0,18 + 0,24) = 0,4285$$

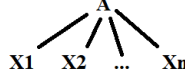
$$P(nodoena|positivo) = 0,24/(0,18 + 0,24) = 0,5714$$

Logo, podemos concluir que se o resultado do teste da nova pessoa for positivo, ela possui aproximadamente 43% (0,4285) de chance de estar doente.

Observação e resumo geral: Naive Bayes é uma técnica de classificação baseado no teorema de Bayes com uma **suposição de independência entre os preditores** diferentemente do caso em 1.2 (Teorema de Bayes), ou seja, O Naive Bayes assume que a presença de uma característica particular em uma classe não está relacionada com a presença de qualquer outro fator. Ao caso da melancia, uma fruta verde, redonda e com um tamanho de diâmetro X é possível ser ela, porém mesmo que estas variáveis dependam uma das outras e de outras características, elas contribuem de forma independente para a probabilidade de que seja uma melancia. É um modelo simples de construir e útil para grandes volumes de dados. Porém a própria independência entre os preditores a torna desvantajosa para aplicação prática e que variáveis categóricas num conjunto de dados de teste que não foram treinadas, não irá estimar essa nova variável.

Por isso *Naive* vem do significado “ingênuo”, pois como a Figura 7.2 demonstra, os atributos contribuem de forma independente para a probabilidade de A.

feitos/Livro ML/bookdown-demo-master/Figuras/naive.bb



$$p(A_k|B_1, \dots, B_n) = p(A_k)p(B_1|A_k)p(B_2|A_k), \dots = p(A_k) \prod_{i=1}^n p(B_i|A_k) \quad k \in 1, \dots, k$$

Figure 7.2: Gráfico de Probabilidade x Entropia.

7.2 Regressão

7.2.1 Análise de Regressão Linear Simples

A análise de variância, pressupõe a independência dos efeitos dos diversos tratamentos utilizados no experimento. Quando a hipótese não é verificada, precisamos refletir a dependência entre os efeitos dos tratamentos. No caso de experimentos quantitativos, frequentemente justifica a existência da equação de regressão, que une os valores dos tratamentos aos analisados. Em grande parte, trata de estimação e/ou previsão do valor médio (para população) da variável dependente com base nos valores conhecidos da variável explanatória, ela é supervisionada.

Como na prática não conseguimos analisar uma população, trabalhamos em cima de amostras e estimamos para o todo, para que possamos fazer uma aproximação. Partimos da ideia de estimarmos uma função com dados amostrais com o menor erro possível. Portanto, o Y_i (população) observado pode ser expresso como:

$$Y_i = \hat{Y}_i + \hat{\mu}_i \quad (7.4)$$

E o modelo para função de regressão amostral:

$$Y_i = \hat{\beta}_0 + \hat{\beta}_1 X_i + \hat{\mu}_i \quad (7.5)$$

em que:

\hat{Y}_i é o valor observado com i níveis de X (estimador da esperança $E(Y|Xi)$), $\hat{\beta}_0$ a constante de regressão estimado e intercepto de \hat{Y} , $\hat{\beta}_1$ o coeficiente de regressão estimado que seria a variação de \hat{Y} em função da variação de cada unidade de X , X_i com i níveis da variável independente e $\hat{\mu}_i$ é o erro associado à distância entre o valor observado e o correspondente ponto na curva. Note que os “chapéis” em cima das variáveis é utilizado quando referimos a estimações, ou seja, são variáveis de dados amostrais e não a população.

Mas como estimâmetros os parâmetros da função de forma que fique mais próxima possível e com o menor erro? Com o **Método dos Mínimos Quadrados**

(MMQ) atribuído ao Carl Friedrich Gauss - matemático alemão - torna-se possível estimar os melhores β_0 e β_1 que minimizam os erros.

Como não podemos observar a função de regressão populacional (FRP), precisamos estimá-lo por meio da função de regressão amostral:

$$Y_i = \hat{\beta}_0 + \hat{\beta}_1 X_i + \hat{\mu}_i$$

$$Y_i = \hat{Y}_i + \hat{\mu}_i$$

$$\text{Logo temos que } \rightarrow \hat{\mu}_i = Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i$$

Podemos ver que os erros $\hat{\mu}_i$ (resíduos) são basicamente as diferenças entre os valores observados e estimados de Y . Ao caso de dados com n pares de observações de Y e X , queremos encontrar a FRA que se encontra o mais próximo possível do Y observado, ou seja, escolher a FRA de modo que a soma dos resíduos $\sum \hat{\mu}_i = \sum (Y_i - \hat{Y}_i)$ seja a menor possível. Porém, como se pode ver pelo diagrama de dispersão na Figura 7.3, os erros possuem a mesma importância com variações entre sinais positivos e negativos e sua somatória será zero. Isso dificultará a possibilidade de minimizarmos.

feitos/Livro ML/bookdown-demo-master/Figuras/mmq.bb

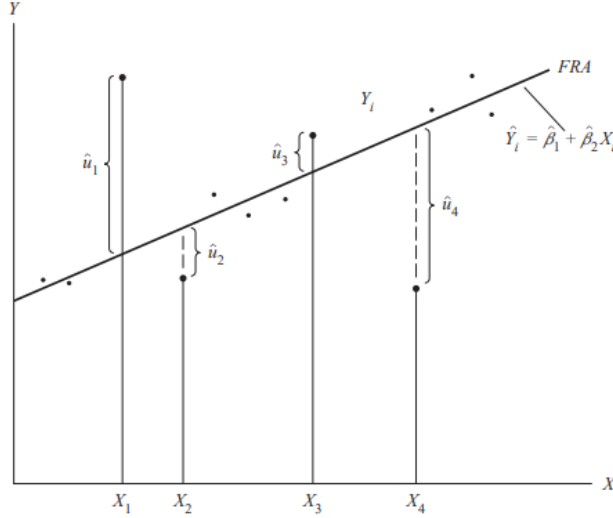


Figure 7.3: Critério do mínimos quadrados Gujarati and Porter (2011).

Para evitarmos isso, utilizamos o critério dos mínimos quadrados, de modo que elevamos os resíduos ao quadrado. Fazendo isso, o método dá mais peso aos resíduos (não irão mais se anular), podendo visualizar melhor o “tamanho” do erro total e obter propriedades estatísticas mais desejáveis.

$$\sum \hat{\mu}_i^2 = \sum (Y_i - \hat{Y}_i)^2$$

$$= \sum (Y_i - \hat{\beta}_0 - \hat{\beta}_1) X_i^2 \quad (7.6)$$

O método dos mínimos quadrados nos oferece estimativas únicas de β_0 e β_1 que proporcionam o menor valor possível (encontrando $\hat{\beta}_0$ e $\hat{\beta}_1$) de $\sum \hat{\mu}_i$. Por meio de cálculo diferenciável (recomendo o leitor interessado em se aprofundar na definição matemática buscar literaturas em foco estatístico, como por exemplo a seção 3A de Gujarati and Porter (2011)) encontra-se:

$$\sum Y_i = n\hat{\beta}_0 + \hat{\beta}_1 \sum X_i \quad (7.7)$$

$$\sum Y_i X_i = \hat{\beta}_0 \sum X_i + \hat{\beta}_1 \sum X_i^2 \quad (7.8)$$

AQUI VOU COLOCAR DO JEITO Q FIZ EM ECONOMETRIA COM AS DEFINICOES E AS DERIVADAS

Para que seja feito o modelo de regressão, ela depende das premissas: independência das variáveis erro, homogeneidade das variâncias, normalidade e relação linear entre as variáveis.

- **Coefficiente de determinação r^2 : medir a qualidade de seu ajuste**

Estimamos os parâmetros e o erro da função, agora precisamos considerar a **qualidade do ajuste** da linha de regressão ajustada a um conjunto de dados, ou seja, vamos descobrir quão “bom” o ajuste dessa linha de regressão amostral é adequada aos dados. Se todas as observações estivessem exatamente em cima da linha de regressão, seria “perfeito”, o que raramente acontece e provavelmente seria um problema de **Overfitting** (será apresentado no próximo capítulo para verificarmos a validade do modelo). O coeficiente de terminação r^2 é uma medida que diz quanto a linha de regressão amostral ajusta-se aos dados.

Para entendermos melhor, vamos visualizar por Diagrama de Venn (Kennedy, 1981). O círculo Y representa a variação da variável dependente Y e o círculo X , a variação da variável explanatória X como vimos em regressão linear. A área sombreada indica o quanto em que a variação de Y é explicada pela variação de X . Quanto maior a área sobreposta, maior a parte da variação de Y é explicada por X . O coeficiente de determinação r^2 é apenas a medida numérica dessa sobreposição. Na Figura 7.4, conforme move-se da esquerda para a direita, a sobreposição aumenta, ou seja, uma proporção cada vez maior da variação de Y é explicada por X (o r^2 aumenta). Sem sobreposição, $r^2 = 0$ e com total sobreposição, $r^2 = 1$, pois 100% da variação de Y é explicada por X . Portanto o coeficiente situa-se no intervalo entre 0 e 1.

Podemos chegar ao coeficiente de determinação apenas por manipulação algébrica:

$$\text{sabemos que: } y_i = \hat{y}_i + \hat{\mu}_i$$

feitos/Livro ML/bookdown-demo-master/Figuras/ballentine.bb

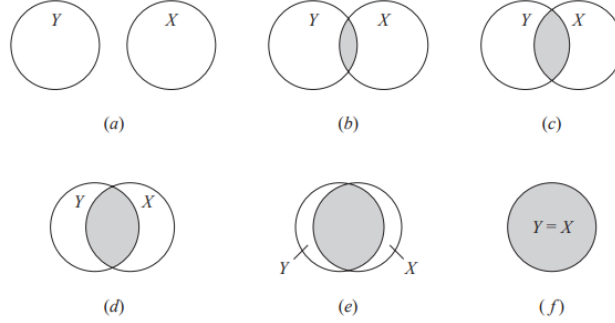


Figure 7.4: Critério do mínimos quadrados Gujarati and Porter (2011).

elevando ao quadrado e somando a amostra: $\sum y_i^2 = \sum \hat{y}_i^2 + \sum \hat{\mu}_i^2 + 2 \sum \hat{y}_i \hat{\mu}_i$

como $\sum \hat{\mu}_i = 0$, temos que: $\sum y_i^2 = \sum \hat{y}_i^2 + \sum \hat{\mu}_i^2$

$$\sum y_i^2 = \hat{\beta}_1^2 \sum x_i^2 + \sum \hat{\mu}_i^2$$

$$\text{podemos dizer } SQT = SQE + SQR \quad (7.9)$$

sendo SQT a soma total dos quadrados, SQE a soma dos quadrados explicados e SQR soma dos quadrados dos resíduos.

dividindo a equação anterior por SQT:

$$1 = \frac{SQE}{SQT} + \frac{SQR}{SQT}$$

$$\text{definindo } r^2 \text{ como: } \frac{SQE}{SQT}$$

$$\text{obtemos: } r^2 = 1 - \frac{SQR}{SQT} \rightarrow 1 - \frac{\sum \hat{\mu}_i}{\sum (Y_i - \bar{Y}_i)^2} \quad (7.10)$$

Por manipulação algébrica, podemos verificar também que $r^2 = \hat{\beta}_1^2 \left(\frac{S_x^2}{S_y^2} \right)$, sendo S_x^2 e S_y^2 as respectivas variâncias amostrais de X e Y .

Note que ao aplicarmos a raiz quadrada no coeficiente de determinação obtemos o coeficiente de correlação visto em 5.1.3, que mede o grau de associação entre duas variáveis.

$$r = \pm \sqrt{r^2}$$

AQUI VOU COLOCAR UM EXEMPLO DE REGRESSÃO PARA ENTENDER E PARTE MATEMÁTICA e falar de ANOVA

Não esqueça: dependendo das variáveis em estudo é possível que haja comportamento polinomial ao observarmos no gráfico, podendo ser quadrática, cúbica, etc. Os procedimentos são os mesmos de que linear, mas basicamente incluímos a variável e seu respectivo grau. Dependendo do comportamento muitas vezes é mais fácil ao invés de manter em exponencial (não linear), linearizarmos a função por meio dos logaritmos, semi-logarítmicos entre outros. Isso faz com que temos menos trabalho para tratarmos e estimarmos os parâmetros da função exponencial.

feitos/Livro ML/bookdown-demo-master/Figuras/explog.bb

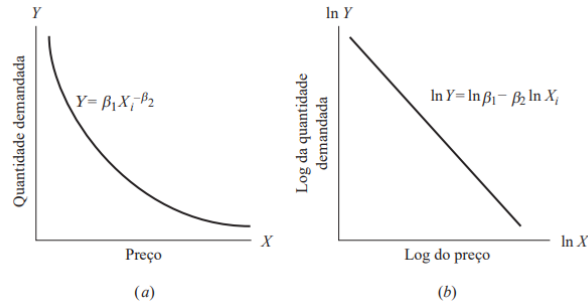


Figure 7.5: Em (a) curva de função exponencial e (b) após aplicarmos o logaritmo (Gujarati and Porter, 2011).

Atualmente é bem comum utilizarmos o modelo **log-log**, pois seu coeficiente angular β_i mede a **elasticidade** de Y em relação a X , ou seja, a variação percentual de Y correspondente a uma variação percentual em X . Por exemplo: na Figura 7.5 se Y representa a quantidade demandada de camisetas e X seu preço unitário. Em (a) temos a relação da quantidade de demanda por camisetas e o preço, mas com a transformação logarítmica teremos a estimação de $-\beta_2$ (pois é uma reta descendente) que indica a elasticidade preço (variação em $\ln(Y)$ por unidade de variação em $\ln(X)$). Portanto teríamos a variação percentual da quantidade demandada de camisetas dada uma variação percentual do preço. Atente-se: **porcentagem** (Gujarati and Porter, 2011).

Resumo geral: Em palavras, r^2 mede a proporção ou percentual da variação total de Y explicada pelo modelo de regressão.

7.2.2 Regressão Linear Múltipla

Na prática deparamos com muitas outros fatores que podem influenciar em sua variável dependente Y . Portanto são acrescentadas dentro de seu modelo de regressão mais variáveis, o que é conhecido como **Regressão Linear Múltipla**, nada mais do que uma ampliação da regressão linear simples. Num modelo, por

exemplo, com três variáveis (caso mais simples) pode ser expressa para a amostra como:

$$Y_i = \hat{\beta}_0 + \hat{\beta}_1 X_{1i} + \hat{\beta}_2 X_{2i} + \mu_i \quad (7.11)$$

Da mesma forma, Y_i a variável dependente, X_2 e X_3 as independentes explanatórias (explicativa), μ_i o erro estocástico e i para indicar i -ésima observação. Ao caso dos parâmetros, β_0 como intercepto, β_1 e β_2 os **coeficientes parciais de regressão/angulares**. β_2 mede a variação no valor médio de Y (esperança de Y), por unidade de variação em X_2 , mantendo X_3 constante, ou seja, traz o efeito “direto” de uma unidade de variação em X_2 sobre o valor médio de Y , excluindo o efeito de X_3 na média de Y . De mesmo modo, X_3 com X_2 constante.

A regressão múltipla pressupõe as mesmas hipóteses de que a regressão linear simples, porém como acréscimo - e muito importante- que as variáveis independentes devem estar **ausentes de multicolinearidade**, ou seja, não devem haver relação linear entre si. Se essa relação linear existir entre X_2 e X_3 **são colineares** ou **linearmente dependentes**, do contrário **linearmente independentes**. Caso a multicolinearidade for perfeita, os coeficientes de regressão das variáveis X serão indeterminados e seus erros padrão, infinitos. Se a multicolinearidade for menos que perfeita, serão determinados mas com grandes erros padrão (em relação aos próprios coeficientes), o que trará um modelo ruim para sua estimação.

Para medirmos a multicolinearidade é comum a análise de **correlação de pearson** entre todas as variáveis, como mencionada em **Medidas de Dependência 5.1.3**, ou analisar a ocorrência de intervalo de confiança mais amplo, verificação de razões “t” insignificantes mesmo que seu R^2 esteja alto, parâmetros estimados muito sensíveis a qualquer alteração de dados e comumente utilizado para verificar o **fator de inflação de variância (FIV)** (Montgomery et al., 2012), que pode ser expressa como:

$$VIF_j = \frac{1}{1 - r_j^2} \quad j = 1, 2, \dots, p \quad (7.12)$$

sendo r^2 o coeficiente de correlação ao quadrado e j para referir as variáveis. Por exemplo, se r_{23}^2 , refere-se ao coeficiente de correlação entre as variáveis X_2 e X_3 . Segundo, quando este indicador apresenta o valor acima de cinco, é possível a existência de multicolinearidade (Maroco, 2014).

De mesmo modo que em regressão linear simples, são estimados os MQO, Máxima verossimilhança e o **coeficiente de determinação múltiplo R^2** (mesma interpretação para regressão linear simples r^2) para que se obtenha a melhor aproximação possível.

7.2.3 Modelo de Probabilidade Linear (MPL)

Considerando um modelo típico de regressão linear simples:

$$Y_i = \beta_0 + \beta_1 X_i + \mu_i$$

em que X =sua renda e $Y = 1$ de que você compre um celular e 0 não compre. Como o regressando é binário, ou dicotômico, chamamos de probabilidade linear (MPL). Pode ser interpretada como probabilidade condicional de que o evento ocorra dado X_i , isto é, $\Pr(Y_i = 1|X_i)$. Neste caso, é a probabilidade de você comprar um celular e cuja renda é dado por X_i .

Para entender este modelo, vamos supor $E(\hat{\mu}_i) = 0$ para evitarmos estimadores tendenciosos (erros). Portanto:

$$E(Y_i|X_i) = \beta_0 + \beta_1 X_i \quad (7.13)$$

Com P_i =probabilidade de que $Y_i = 1$ (ocorrência do evento) e $(1 - P_i)$ =probabilidade de $Y_i = 0$ (não ocorrência do evento). Y_i possui a seguinte **distribuição de probabilidade de Bernoulli**:

Table 7.1:

Y_i	Probabilidade
0	$1 - P_i$
1	P_i
Total	1

Aplicando a esperança, obtemos:

$$E(Y_i) = 0(1 - P_i) + 1(P_i) = P_i \quad (7.14)$$

Igualando (7.14) com (7.13), obtemos:

$$E(Y_i|X_i) = \beta_0 + \beta_1 X_i \quad (7.15)$$

Isso verifica que a esperança condicional do modelo de regressão (7.4) pode ser interpretada como a probabilidade condicional de Y_i . Note que, como explicado em 1.2 sobre **Distribuição Bernoulli** e **Distribuição Binomial**, caso haja n observações independentes, cada um com uma probabilidade p (sucesso) e probabilidade $(1 - p)$ (fracasso) e X dessas observações representarem o número de sucessos, X então segue a distribuição binomial (com médi np e variância $np(1 - p)$). Lembrando que a probabilidade P_i situa-se entre 0 e 1 $\rightarrow 0 \leq E(Y_i|X_i) \leq 1$.

Alguns detalhes importantes:

- A hipótese de normalidade de μ_i não se verifica no caso dos modelos de probabilidade linear, pois os termos de erro assumem também apenas dois valores, seguindo a distribuição de Bernoulli. Se objetivo for a estimação pontual, a hipótese de normalidade deixa de ser necessária (Gujarati and Porter, 2011) e que conforme aumentamos o tamanho da amostra indefinidamente, os estimadores de MQO tendem geralmente a distribuir-se normalmente.
- Como sabe-se, a média e variância de uma distribuição Bernoulli possuem respectivamente p e $p(1-p)$. Logo a variância é heterocedástica $var(\mu_i) = P_i(1 - P_i)$ e portanto os estimadores de MQO não são eficientes (não possuem variância mínima). Podemos fazer a transformação para que seja homocedástico:

$$\sqrt{E(Y_i|X_i) - [1 - E(Y_i|X_i)]} = \sqrt{P_i(1 - P_i)} = \sqrt{w_i}$$

$$\frac{Y_i}{\sqrt{w_i}} = \frac{\beta_0}{\sqrt{w_i}} + \frac{\beta_1 X_i}{\sqrt{w_i}} + \frac{\mu_i}{\sqrt{w_i}} \quad (7.16)$$

Com a transformação, pode-se calcular por MQO (ponderados).

Alternativas para o MPL:

- Como mencionado, a probabilidade condicional situa-se entre 0 e 1, porém por MQO não levarem em conta esta restrição. Pode-se verificar os valores que constam entre o intervalo, considerando os valores negativos como 0 e maiores que 1 como iguais a 1 ou aplicar algum outro modelo para garanti-los dentro dos intervalos.
- O R^2 costuma-se situar muito abaixo de 1. Por ser limitado em caso de modelos binários, muitos pesquisadores buscam evitar seu uso.

Os modelos mais comuns para ser utilizado como alternativa ao MPL são o **logit** e o **probit** para evitar estes problemas.

7.2.3.1 Logit

A fim de fazer com que P_i varie entre 0 e 1 e relacione-se linearmente a X_i , a **função de distribuição logística** pode ser expressa como:

$$P_i = \frac{1}{1 + e^{-Z_i}} = \frac{e_i^Z}{1 + e_i^Z} \quad (7.17)$$

e $(1 - P_i)$ da probabilidade fracasso:

$$1 - P_i = \frac{1}{1 + e^{Z_i}} \rightarrow e^{Z_i} \quad (7.18)$$

onde $Z_i = \beta_0 + \beta_1 X_i$. Assim Z_i varia de $-\infty$ a ∞ e portanto P_i entre 0 e 1.

Para estimarmos a MQO, precisamos linearizar a função:

$$L_i = \ln\left(\frac{P_i}{1 - P_i}\right) = Z_i = \beta_0 + \beta_1 X_i \quad (7.19)$$

O modelo **logit** faz com que:

- A probabilidade varie entre 0 e 1, enquanto Z e L possam variar de $-\infty$ a ∞ ;
- Mesmo que as probabilidades não sejam lineares, L é linear em X ;
- Pode-se aplicar com mais regressores e com mesma interpretação angular medindo a variação de L para uma unidade variação em X e para o intercepto;
- Se L torna-se maior e positivo quando as chances do evento de interesse ocorrer aumenta, do contrário (maior e negativo) de não ocorrer;
- Como em MPL, o modelo Logit é heterocedástico precisa-se ponderar (Gujarati and Porter, 2011; Cox, 1970):

$$\sqrt{w_i} L_i = \beta_0 \sqrt{w_i} + \beta_1 \sqrt{w_i} X_i + \sqrt{w_i} \mu_i \quad (7.20)$$

em que, com a variância $\hat{\sigma}^2 = \frac{1}{N_i \hat{P}_i (1 - \hat{P}_i)}$, W_i é o peso $N_i \hat{P}_i (1 - \hat{P}_i)$. Por fim, aplicar o mínimos quadrados ponderados (da mesma forma que MQO, porém com a nova transformação de dados) e estimarmos os parâmetros normalmente.

Como o R^2 não é significativa nos modelos binários. É comum utilizar as **pseudo R^2** [long1997regression] - existe uma variedade delas - ou o **Count R^2** que nada mais é que o número de previsões corretas com o número total de observações. Para a hipótese nula de que todos os coeficientes angulares são simultaneamente iguais a zero, utiliza-se a **estatística da razão de verossimilhança** que segue a distribuição χ^2 que equivale ao teste F.

7.2.3.2 Probit

7.2.3.3 Tobit

7.2.4 Exemplos

7.3 Gradiente Descendente (GD)

Para a obtenção dos parâmetros de forma analítica, como regressões, muitas vezes é difícil obter os parâmetros que minimizam determinada função de interesse. Dificuldades em obter a solução do sistema na forma fechada (ou não existir) ou quando n é muito grande, o cálculo da inversa (estimando os parâmetros matricialmente) pode ser muito caro computacionalmente.

O **Gradiente Descendente (GD)** pode ser muito útil dependendo da situação, conhecido também como **máximo declive**, é um método numérico utilizado em otimização. Tem como finalidade identificar um mínimo local de uma função de modo iterativo, no qual a cada iteração toma-se a direção do gradiente. Muitas vezes serve como base para algoritmos de segunda ordem como Métodos de Newton, por exemplo.

É uma função para casos gerais, por praticidade vamos supor que temos uma função denominada custo com apenas dois parâmetros $J(\theta_0, \theta_1)$ e queremos estimar seus parâmetros que minimizam seus erros. Inicialmente atribuímos quaisquer estimativas iniciais para valores de θ_0 e θ_1 , com o GD vamos alterando os valores dos θ 's para reduzirmos $J(\theta_0, \theta_1)$ até que se chegue a um valor mínimo local.

Um exemplo que gosto muito, por NG, Andrew Y. (2019): observe a Figura 7.6 e imagine que você está em um campo, com dois montes. Mantenha sua imaginação de que está situado na cruz preta - ponto 0 - no primeiro monte vermelho. Com o GD vamos olhar 360 graus ao redor do ponto em que você está situado apenas para descobrir a resposta de que “se você fosse dar um pequeno passo em alguma direção ao seu redor com o objetivo de ir para o ponto mais baixo do campo o mais rápido possível, para qual direção você deve andar?”

Supondo que após olhar para todos os lados, com análise de GD você descobriu que seu primeiro passo será no ponto 1 da Figura 7.6. Após isso, você observa novamente para todos os lados e faz outra análise de GD para verificar aonde você vai se deslocar em seu segundo passo para chegar o mais rápido possível até concluir que será o ponto 2. Assim, sucessivamente, você vai se deslocando para os respectivos pontos 3, 4 e sucessivamente até convergir em seu objetivo Z, porém caso você iniciasse pelo ponto K, é bem possível que por meio do GD você descesse o monte por outro trajeto, encontrando outros pontos ótimos locais até chegar a outro ponto otimizado (descer por completo o monte). Esta é a ideia do Gradiente Descendente, por meio de iterações, o algoritmo vai identificando os pontos ótimos (estimadores mínimos) até convergir num ótimo local da função.

Em caso de funções simples como regressão linear, não é necessário o uso de GD. Mas em casos com muitas variáveis e ordens, pode ser bem viável.

O algoritmo pode ser expresso como:

$$\theta_j := \theta_j - \alpha \frac{d}{d\theta_j} J(\theta_0, \theta_1) \text{ com } j = \theta_0 \text{ e } j = \theta_1 \quad (7.21)$$

com j referindo-se à quantidade de observações (parâmetros que pretendemos estimar) da amostra.

O algoritmo é processado da seguinte forma: imagine na mesma Figura 7.6 que você irá dar seu primeiro passo, olhou os 360 graus e inseriu as variáveis em

feitos/Livro ML/bookdown-demo-master/Figuras/gd.bb

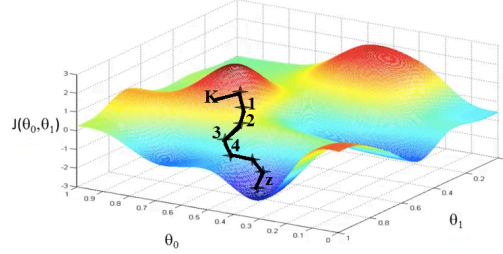


Figure 7.6: Gráfico tridimensional a exemplo de Gradiente Descendente (NG, Andrew Y., 2019).

seu algoritmo de GD e seu destino é em $Z = 10$. Seu algoritmo calcula se você passou seu destino mais do que devia ou se você está atrás de Z ainda e também verifica se precisa dar passos grandes por estar bem longe de seu destino, ou passos menores. Supondo que seu α um pouquinho alto, podemos dar um passo grande para descer o monte (1) pela diferença da observação que você inseriu com $\alpha \frac{d}{d\theta_j} J(\theta_0, \theta_1)$. Caso fosse uma taxa pequena de α , seu passo seria menor e sua derivada (taxa de variação) vai lhe dizer se você passou do ponto ótimo de Z (o quão a frente) ou está para trás (quão para trás) desse ponto ótimo.

Com o primeiro passo dado (supor passo 1 = 40), você precisa fazer o mesmo procedimento tomando agora o passo 1 como se fosse o inicial novamente, ou seja, atualizando sua função para cada θ **simultaneamente** (caso dois θ 's de entrada para a função, atualiza-se para ambos) até encontrar o novo valor ótimo do próximo passo no ponto 2 = 15. Conforme vai se aproximando de Z , seus passos vão ficando cada vez menores (de 15 para 11; de 11 para 10,50; de 10,50 para 10,10; de 10,10 para 10,05; etc) até chegar na melhor aproximação de $Z = 10$ que é o ponto ótimo da função.

Assim o algoritmo encontra os melhores parâmetros para buscar o ponto otimizado, com a estimativa dos melhores parâmetros para a aproximação com os menores erros (sim! Podemos encontrar os parâmetros dos exemplos de regressão com este algoritmo também!)

Desta forma, atribuímos (“:=”) para a própria observação de entrada da função receber ela mesma subtraída α que multiplica a derivada da função em relação a observação de entrada. Para que atualize a cada passo (iteração). α (**learning rate - taxa de aprendizagem**) é um valor fixo que controla o tamanho do passo em cada iteração: quando α for pequeno, o método fica lento, quando grande ele pode falhar na convergência e até mesmo divergir. Seu valor de-

pende muito da pesquisa e de suas fundamentações teóricas, o que recomendo o leitor quando utilizar este método verificar um valor adequado, pode ser que dependendo do valor da taxa demore muito para finalizar o algoritmo pela quantidade de iterações (tamanhos de passos muito pequenos) ou divergir (tamanho de passos muito grandes). Rendle and Schmidt-Thieme (2008) divulgaram que a fatoração de matrizes para a predição de *ratings* nos dados do desafio *Netflix* precisou de 200 iterações, usando uma taxa de aprendizagem de 0,01.

Para facilitar a compreensão do efeito da taxa de variação, observe a Figura 7.7. No primeiro gráfico você inicia seu algoritmo com o valor θ e com a derivada podemos observar que inclinação da reta tangente ao ponto é positiva ($\frac{d}{d\theta}j(\theta) \geq 0$), portanto em $\theta = \theta - \alpha$ um valor positivo, faz com que esse novo θ (segunda iteração) seja menor que o da primeira iteração, visto que terá que subtrair e deslocar-se para esquerda para tender ao ponto mínimo. Da mesma forma, ao segundo gráfico, podemos verificar que a inclinação é negativa ($\frac{d}{d\theta}j(\theta) \leq 0$), portanto $\theta = \theta - \alpha$ um valor negativo, fará com que o novo θ seja maior do que da primeira iteração, pois irá somar e deslocar-se para direita tendendo ao ponto mínimo.

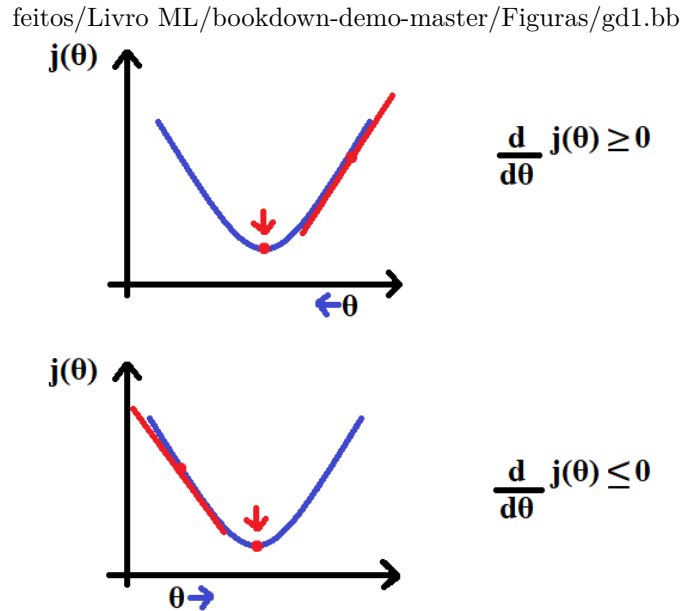


Figure 7.7: Efeito da taxa de variação no Gradiente Descendente.

Como pode-se perceber, a taxa de aprendizagem e a taxa de variação são fundamentais e complementares para o algoritmo de GD, pois elas dizem o tamanho do passo e em que posição estamos em relação ao ponto ótimo da função.

7.3.1 Exemplos

1. **Uma variável:** Vamos supor a seguinte função custo:

$$j(\theta) = \theta^2$$

Queremos minimizá-la $\min j(\theta)$. Portanto precisamos inicialmente colocar um número aleatório para nosso parâmetro - não ótimo - para que o algoritmo atualize a cada iteração. Vamos supor a taxa de aprendizagem (*learning rate*) $\alpha = 0,1$ e $\theta = 4$ para facilitar. Ou seja, $j(\theta) = 4^2 = 16$. Vamos atualizar os parâmetros:

$$\theta := \theta - \alpha \cdot \frac{d}{d\theta} j(\theta)$$

derivando a função $j(\theta) = \theta^2$ e substituindo:

$$\theta := \theta - \alpha \cdot 2\theta$$

substituindo os valores de α e θ :

$$\begin{aligned} \theta &:= 4 - 0,1 \cdot 2 \cdot 4 \\ &\rightarrow \theta := 3,2 \end{aligned}$$

Na iteração obtemos $\theta = 3,2$. Se substituirmos em $j(\theta)$ novamente, iremos obter $j(\theta) = (3,2)^2 = 10,24$. Agora atualizando novamente para a próxima iteração:

$$\theta := \theta - \alpha \cdot 2\theta$$

$$\begin{aligned} \theta &:= 3,2 - 0,1 \cdot 2 \cdot 3,2 \\ &:= 2,56 \end{aligned}$$

Portanto, $j(\theta) = (2,56)^2 = 6,55$. Sucessivamente, vamos fazendo as iterações até convergir:

θ	$j(\theta)$
4	16
3,2	10,24
2,56	6,55
2,04	4,19
1,632	2,663
.	.
.	.
.	.
0	0

Da mesma forma, se iniciarmos o algoritmo com -4:

θ	$j(\theta)$
-4	16
-3,2	10,24
-2,56	6,55
-2,04	4,19
-1,632	2,663
.	.
.	.
.	.
0	0

Note que conforme θ diminui, o custo também. Conforme mais iterações são aplicadas, mais “ótimo” será. Graficamente para -4 em vermelho e +4 em azul:

feitos/Livro ML/bookdown-demo-master/Figuras/gdx2.bb

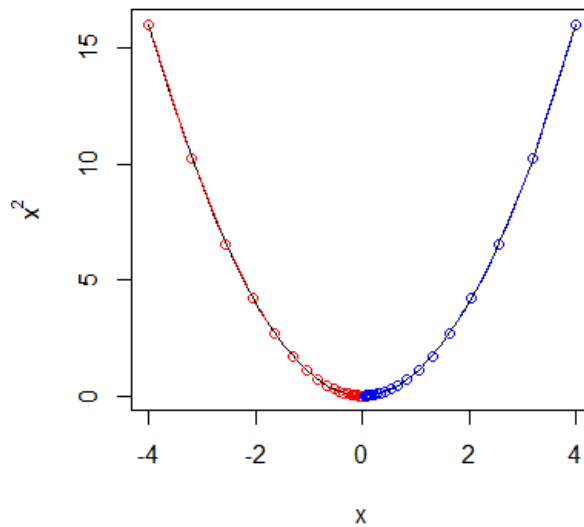


Figure 7.8: Função X^2 com valores de entrada -4 e +4.

2. **Duas variáveis:** Vamos supor a seguinte função de custo com $\alpha = 0,1$, $\theta_1 = 1$ e $\theta_2 = 2$:

$$j(\theta_1, \theta_2) = \theta_1^2 + \theta_2^2$$

$$j(\theta_1, \theta_2) = 1^2 + 2^2 = 5$$

Queremos $\min j(\theta_1, \theta_2)$ Como explicado, ao caso de haver mais de um parâmetro precisamos separar atualizar cada um simultaneamente e aplicar derivada parcial em sua função:

$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} j(\theta_1, \theta_2) \quad \text{e} \quad \theta_2 := \theta_2 - \alpha \frac{d}{d\theta_2} j(\theta_1, \theta_2)$$

calculando as derivadas parciais de $j(\theta_1, \theta_2) = \theta_1^2 + \theta_2^2$ obtemos:

$$\frac{d}{d\theta_1} j(\theta_1, \theta_2) = 2\theta_1 \quad \text{e} \quad \frac{d}{d\theta_2} j(\theta_1, \theta_2) = 2\theta_2$$

substituindo:

$$\theta_1 := \theta_1 - \alpha \cdot 2\theta_1 \quad \text{e} \quad \theta_2 := \theta_2 - \alpha \cdot 2\theta_2$$

inserindo os valores:

$$\theta_1 := 1 - 0,1 \cdot 2 \cdot 1 \quad \text{e} \quad \theta_2 := 2 - 0,1 \cdot 2 \cdot 2$$

$$\theta_1 := 0,8 \quad \text{e} \quad \theta_2 = 1,6$$

Portanto após a iteração, temos que $j(\theta_1, \theta_2) = 0,8^2 + 1,6^2 = 3,2$. Da mesma forma, para a próxima iteração temos:

$$\theta_1 := 0,8 - 0,1 \cdot 2 \cdot 0,8 \quad \text{e} \quad \theta_2 := 1,6 - 0,1 \cdot 2 \cdot 1,6$$

$$\theta_1 := 0,64 \quad \text{e} \quad \theta_2 = 1,28$$

Portanto teremos $j(\theta_1, \theta_2) = 0,64^2 + 1,28^2 = 2,048$. Assim sucessivamente:

θ_1	θ_2	$j(\theta_1, \theta_2)$
1	2	5
0,8	1,6	3,2
0,64	1,28	2,48
·	·	·
·	·	·
·	·	·
0		0

3. Erro quadrado médio (Regressão Linear Simples:) Observe a função de regressão linear:

$$f_\theta(X) = \theta_0 + \theta_1 * X$$

A função de custo:

$$j(\theta) = \frac{1}{m} \sum_{i=1}^m (f_{\theta}(x^i) - y^i)^2$$

Primeiramente vamos encontrar a derivada parcial de $j(\theta_0, \theta_1)$:

$$\frac{d}{d\theta_0} j(\theta_0, \theta_1) = \frac{d}{d\theta_0} \left(\frac{1}{m} \sum_{i=1}^m (f_{\theta}(x^i) - y^i)^2 \right) \rightarrow \frac{2}{m} \sum_{i=1}^m (f_{\theta}(x^i) - y^i)$$

$$\frac{d}{d\theta_1} j(\theta_0, \theta_1) = \frac{d}{d\theta_1} \left(\frac{1}{m} \sum_{i=1}^m (f_{\theta}(x^i) - y^i)^2 \right) \rightarrow \frac{2}{m} \sum_{i=1}^m (f_{\theta}(x^i) - y^i) x^i$$

Pode-se também multiplicar a função de custo por $\frac{1}{2}$ para que quando faz-se a derivada, facilite no cálculo e multiplicar a função de custo por um escalar não irá afetar a localização do mínimo.

$$j(\theta) = \frac{1}{2m} \sum_{i=1}^m (f_{\theta}(x^i) - y^i)^2$$

Com isso em foco de minimizarmos, basta aplicarmos o banco de dados de X e Y em seu modelo e de seus dois θ 's de entrada. Repetindo as iterações para atualizar seus valores até a convergência e identificando os parâmetros que se aproximam.

Chapter 8

Algoritmos de Aprendizagem - Parte II

8.1 Máquina de Vetores Suporte - *Support Vectors Machine*

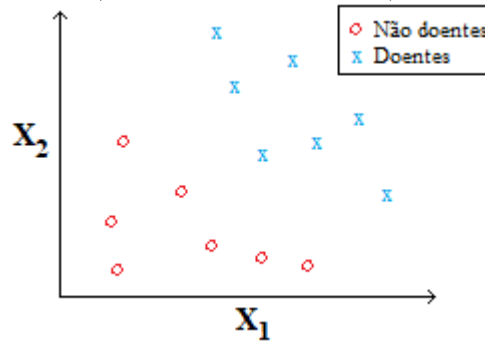
A Máquina de Vetores de Suporte (SVMs, do inglês *Support Vectors Machine*), são embasadas pela teoria de aprendizado estatística, desenvolvida por Vapnik (2013) com o propósito de resolver problemas de classificação de padrões, foi originalmente desenvolvida para classificação binária, construindo um hiperplano como superfície de decisão que separa classes linearmente separáveis (ao caso de não-linearmente separáveis utiliza-se função de mapeamento). Muitos a comparam com redes neurais pelo fato de ser eficiente em trabalhar com dados de alta dimensionalidade (Sung and Mukkamala, 2003; Ding and Dubchak, 2001). É utilizada atualmente tanto para regressão quanto para qualificação e é uma análise supervisionada.

Vamos supor um gráfico com características X_1 e X_2 e já classificados na amostra os indivíduos que estão e não estão doentes, quanto maior o valor de ambos maior a probabilidade do indivíduo ser classificado como doente.

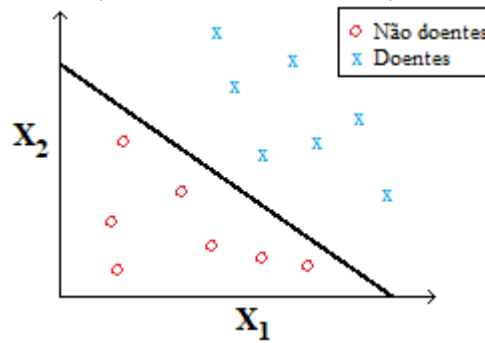
Um exemplo como esse é simples observar que podemos separar os dados traçando uma linha reta. Classificando os doentes para a direita e acima do gráfico e não doentes a esquerda e abaixo. Esta linha é o que chamamos de **hiperplano de separação**.

Lembrando que hiperplano é uma generalização de um plano: uma dimensão é um ponto, duas dimensões é uma linha e três é um plano. Quando tratamos de mais dimensões é o que denominamos hiperplano. O SVM pode trabalhar com qualquer dimensão.

feitos/Livro ML/bookdown-demo-master/Figuras/svm1.bb

Figure 8.1: Gráfico de X_1 e X_2 classificado em doentes e não doentes.

feitos/Livro ML/bookdown-demo-master/Figuras/svm2.bb

Figure 8.2: Gráfico de X_1 e X_2 classificado em doentes e não doentes por meio de um hiperplano.

8.1. MÁQUINA DE VETORES SUPORTE - SUPPORT VECTOR MACHINE 69

Podemos encontrar em uma amostra vários hiperplanos de separação e válidos para a classificação de um *dataset*. Mas não necessariamente este é o melhor. Até mesmo pode ser que classifique alguns errados.

feitos/Livro ML/bookdown-demo-master/Figuras/svm3.bb

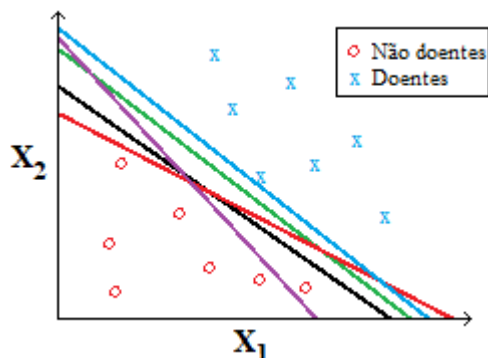


Figure 8.3: Gráfico de X_1 e X_2 com a classificação de doentes e não doentes por mais de um hiperplano de separação.

O objetivo do algoritmo de SVM é identificar um hiperplano “ideal” que busca classificar o conjunto de dados da melhor maneira possível (menor erro). Ao verificar as distâncias perpendiculares entre as observações e o hiperplano de separação, obtemos uma **Margem**. Os pontos menos distantes do hiperplano são os que a definem. Estes pontos são os **Vetores de Suporte (VS)**, que têm este nome pois eles dão suporte ao hiperplano. Caso forem movidos, a margem acompanhará o movimento.

feitos/Livro ML/bookdown-demo-master/Figuras/svm4.bb

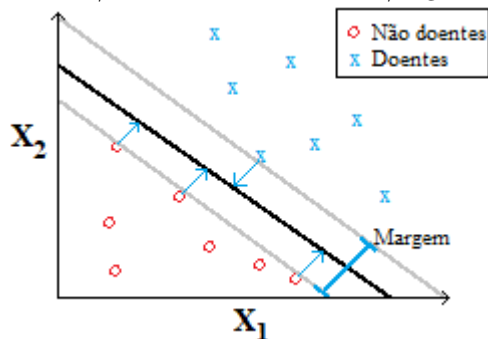


Figure 8.4: Gráfico de X_1 e X_2 com a classificação de doentes e não doentes pelo hiperplano e sua margem.

8.1.1 Classificação de Padrões Linearmente Separáveis

Uma classificação linear consiste em determinar uma função $f : X \subseteq \mathbb{R}^N \rightarrow \mathbb{R}^N$ que atribuirá um valor de $+1$ se $f(x) \geq 0$ e -1 se $f(x) < 0$. Sendo assim pelo produto interno (ver Produto Interno em 1.2):

$$f(x) = \langle \vec{w}, \vec{x} \rangle + b \quad (8.1)$$

$$= \sum_{i=1}^n \vec{w}_i \vec{x}_i + b \quad (8.2)$$

em que \vec{w} e b são popularmente conhecido como **vetor peso** e **bias** e parâmetros responsáveis em controlar a função e a regra da decisão (Lima, 2002). Os valores destes parâmetros são obtidos pelo processo de aprendizagem a partir dos dados de entrada (Gonçalves, 2008). Sendo o vetor peso \vec{w} que define uma direção perpendicular ao hiperplano e com a variação de b o hiperplano é movido paralelamente a ele mesmo.

Portanto um SVM linear procura encontrar um hiperplano ótimo que separe da melhor maneira possível os dados de cada classe (margem máxima).

feitos/Livro ML/bookdown-demo-master/Figuras/svm5.bb

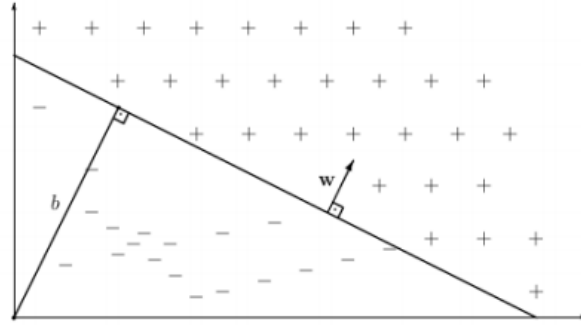


Figure 8.5: Interpretação geométrica de \vec{w} e b sobre um hiperplano (Lima, 2002; Gonçalves, 2008).

8.1.2 Hiperplano de Separação Ótima / Margem Máxima

Um hiperplano é considerado de margem máxima se separa um conjunto de vetores sem erros e a distância entre os vetores de classes diferentes mais próximos do hiperplano é a máxima possível.

Assumindo que o conjunto de dados é linearmente separável, o hiperplano ótimo é que possuir a maior margem:

feitos/Livro ML/bookdown-demo-master/Figuras/margmax.bb

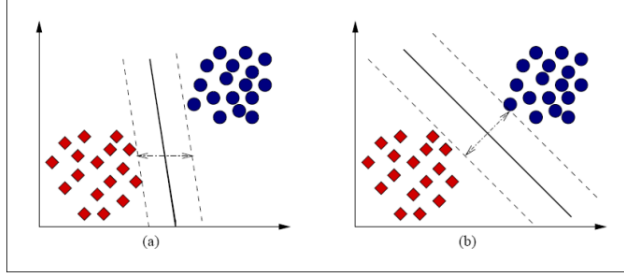


Figure 8.6: (a) Um hiperplano de separação com uma pequena margem. (b) Um hiperplano de Margem máxima (da Silva Meloni, 2009).

$$\langle \vec{w} \cdot \vec{x} \rangle + b = 0 \quad (8.3)$$

em que \vec{w} e b , o vetor peso e *bias* respectivamente.

Assumindo a restrição:

$$\begin{aligned} \langle \vec{w} \cdot \vec{x}_i \rangle + b &\geq +1, \text{ para } y_i = +1 \\ \langle \vec{w} \cdot \vec{x}_i \rangle + b &\leq -1, \text{ para } y_i = -1 \end{aligned} \quad (8.4)$$

Os classificadores lineares que separam o conjunto de dados em treinamento possuem margem positiva. Esta restrição nos mostra que não há dados entre 0 e ± 1 , tendo como a margem sempre maior que a distância entre os hiperplanos $\langle \vec{w} \cdot \vec{x}_i \rangle + b = 0$ e $|\langle \vec{w} \cdot \vec{x}_i \rangle + b| = 1$. Fazendo com que as SVMs sejam chamadas de **Margens Rígidas**, do inglês *Hard Margin*. Com isso, ao combinar ambas equações restrições:

$$y_i(\langle \vec{w} \cdot \vec{x}_i \rangle + b) \geq 1, \quad i = \{1, 2, \dots, n\} \quad (8.5)$$

Ou:

$$y_i(w^T \cdot \vec{x}_i + b) \geq 1, \quad i = \{1, 2, \dots, n\} \quad (8.6)$$

Aplicando a distância Euclidiana (d_+ e d_-) entre os vetores de suporte positivos/negativos e o hiperplano, definido a margem ρ de um hiperplano de separação como sendo a maior geométrica entre todos os hiperplano, é possível representar $\rho(d_+ + d_-)$ (Gonçalves, 2008).

$$d_i(\vec{w}, b; \vec{x}_i) = \frac{|\langle \vec{w} \cdot \vec{x}_i \rangle + b|}{\|\vec{w}\|} = \frac{y_i(|\langle \vec{w} \cdot \vec{x}_i \rangle + b|)}{\|\vec{w}\|} \quad (8.7)$$

em que $d(\vec{w}, b; \vec{x}_i)$ é a distância de um dado \vec{x}_i ao hiperplano (\vec{w}, b) (Lima, 2002). Ao levarmos em consideração a restrição de (8.6), podemos expressar:

$$d(\vec{w}, b; \vec{x}_i) \geq \frac{1}{\|\vec{w}\|} \quad (8.8)$$

Identificando $\frac{1}{\|\vec{w}\|}$ como o limite inferior da distância entre os vetores de suporte \vec{x}_i e o hiperplano (\vec{w}, b) . Logo, as distâncias serão:

$$d_+ = d_- = \frac{1}{\|\vec{w}\|} \quad (8.9)$$

A margem é sempre maior que a última instância, a minimização de $\|\vec{w}\|$ nos traz a maximização da margem. Podemos definir a margem ρ como (Gonçalves, 2008):

$$\rho = (d_+ = d_-) = \frac{2}{\|\vec{w}\|} \quad (8.10)$$

Assim teremos a distância entre hiperplanos e os vetores de suporte:

feitos/Livro ML/bookdown-demo-master/Figuras/svm6.bb

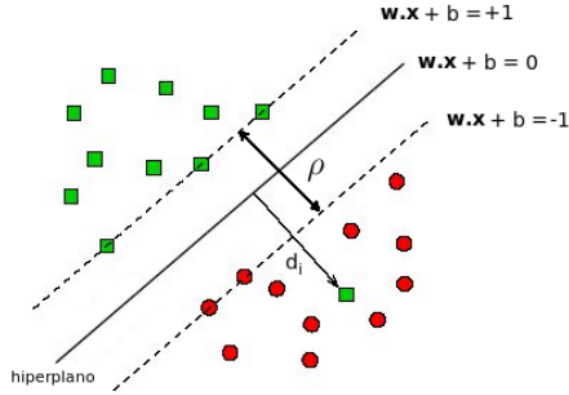


Figure 8.7: Distância entre hiperplanos e vetores de suporte (Gonçalves, 2008).

Para minimizarmos $\|\vec{w}\|$ (maximizarmos esta margem), podemos utilizar a teoria dos multiplicadores de Lagrange (ver Multiplicadores de Lagrange em 1.2):

$$L(\vec{w}, b, \alpha) = \frac{1}{2} \|\vec{w}\|^2 - \sum_{i=1}^n \alpha_i (y_i (\langle \vec{w}, \vec{x}_i \rangle + b) - 1) \quad (8.11)$$

8.1. MÁQUINA DE VETORES SUPORTE - SUPPORT VECTORS MACHINE 73

em que α_i são os multiplicadores de Lagrange. Então agora visamos minimizar $L(\vec{w}, b, \alpha)$, em relação a \vec{w} e b e a maximização dos α_i (encontrar os pontos ótimos pelas derivadas parciais iguais a zero). Portanto:

$$\frac{\partial L}{\partial b} = 0$$

$$\frac{\partial L}{\partial \vec{w}} = \left(\frac{\partial L}{\partial w_1}, \frac{\partial L}{\partial w_2}, \dots, \frac{\partial L}{\partial w_n} \right) = 0$$

Obtemos por meio delas:

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad (8.12)$$

$$\vec{w} = \sum_{i=1}^n \alpha_i y_i \vec{w}_i \quad (8.13)$$

Substituindo as equações (8.12) e (8.13) em (8.11), chegamos em:

$$L = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \quad (8.14)$$

com $\alpha_i \geq 0, i \in \{1, 2, \dots, n\}$. Serão representados os valores ótimos de (\vec{w}, b) por (\vec{w}^*, b^*) . E α_i^* assume valores positivos para os exemplos de treinamento que estão a uma distância do hiperplano ótimo igual a largura da margem, os vetores de suporte (Gonçalves, 2008). Podemos perceber que o hiperplano de separação ótimo é obtido pelos vetores de suporte e não de todo o conjunto (Lorena and de Carvalho, 2003).

Com um vetor suporte dado \vec{x}_j , obtemos valor de b^* pela condição de KKT (ver Karush-Kuhn-Tucker em 1.2):

$$b^* = y_j - \langle \vec{w}^*, \vec{x}_j \rangle. \quad (8.15)$$

Com todos os valores dos parâmetros calculados podemos, por fim, ter um novo padrão z calculando:

$$sgn(\langle \vec{w}^*, \vec{z} \rangle + b^*) \quad (8.16)$$

sendo sgn a função sinal que fornece o valor 1 se o número for positivo, valor 0 se o número for zero e -1 se for negativo.

Pode-se também utilizar as **Margens Flexíveis**, que busca não garantir todas as observações no lado certo, tolerando algumas violações. Basicamente atribui-se um erro para cada observação que viola o hiperplano proporcional o quanto passou a margem e o violou. Acrescentando na função este erro para compensar. Nos algoritmos dos *softwares* estatísticos, é comum o uso de uma constante **C** que controla a severidade do modelo, dando limite do tanto do erro que pode haver no algoritmo. Seu valor depende muito da pesquisa e de sua fundamentação teórica, pois pode ser que aumente a quantidade de vetores de suporte e até mesmo podendo causando problemas de **overfitting** (9.1).

8.1.3 Classificação de Padrões Não-Linearmente Separáveis

Nas situações reais a maioria dos padrões são mais complexos e não-lineares. O conjunto de dados é classificado como não-linearmente separável ao caso de não ser possível separar os dados com um hiperplano:

feitos/Livro ML/bookdown-demo-master/Figuras/svm7.bb

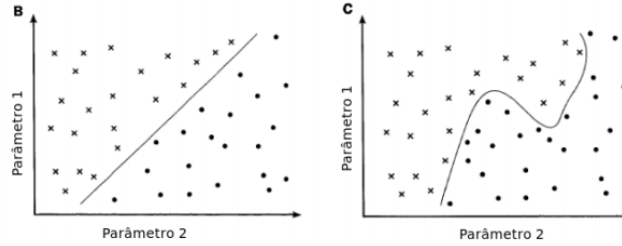


Figure 8.8: Padrões linearmente e não-linearmente separável respectivamente (Gonçalves, 2008).

Segundo Smola et al. (2000), o teorema de Cover afirma que um problema não-linear possui maior probabilidade de ser linearmente separável em um espaço de mais alta dimensionalidade. Com isso, a SVM não-linear faz uma mudança de dimensionalidade por meio das funções *Kernel* para tratarmos de uma problema de classificação linear e permitindo elaborar o hiperplano ótimo (note que ACP possui uma ideia similar dado suas propriedades).

Um conjunto de entrada X com pares $\{(x_1, y_1); (x_2, y_2), \dots, (x_n, y_n)\}$ de uma amostra de treinamento (não-linearmente separável), são mapeados por meio de uma função ϕ a fim de obter um novo conjunto de dados X' linearmente separável em um espaço de maior dimensionalidade, representado por $\{(\phi(x_1), y_1), \dots, (\phi(x_n), y_n)\}$.

Com os dados de treinamento mapeados para o espaço de características, utiliza-se os valores mapeados $\phi(x)$ ao invés de x , sendo assim o problema consiste em:

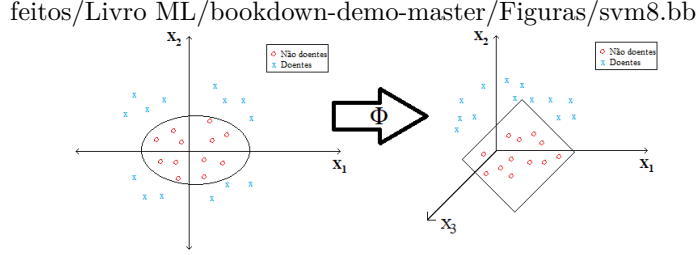


Figure 8.9: Mapeamento de um conjunto de entrada X para o espaço característica. Um novo conjunto X' .

$$L = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \langle \phi(x_i) \cdot \phi(x_j) \rangle \quad (8.17)$$

em que $\alpha_i \geq 0$. Para classificação não-linear as mesmas considerações de KKT descrito no linear. O hiperplano fica expresso como:

$$(\vec{w}\phi \cdot (\vec{x})) + b = 0 \quad (8.18)$$

Ao problema de classificação não-linear de um novo padrão \mathbf{z} :

$$\text{sgn}(\langle \vec{w}^* \cdot \phi(z) \rangle + b^*) \quad (8.19)$$

Uma função *Kernel*, pertencente a um domínio que permita calcular o produto interno para calculá-lo, recebe dois dados de entrada x_i e x_j destes dados no espaço de características.

$$\kappa = (x_i, x_j) = \langle \phi(x_i) \cdot \phi(x_j) \rangle \quad (8.20)$$

A função precisa satisfazer as condições do **Teorema de Mercer**, portanto a matriz K é positivamente definida (autovalores maiores que zero). K é obtida por:

$$K = K_{ij} = \kappa(x_i, x_j) \quad (8.21)$$

As funções *Kernel* mais utilizadas são:

Table 8.1: *Kernels* mais populares (Gonçalves, 2008).

Tipo de Kernel	Função $\kappa(x_i, x_j)$
Polinomial	$(\langle x_i \cdot x_j \rangle + 1)^p$
Gaussiano	$e^{-\frac{\ x_i - x_j\ ^2}{2\sigma^2}}$
Sigmoidal	$\tanh(\beta_0 \langle x_i \cdot x_j \rangle) + \beta_1$

Lembrando que o pesquisador precisa escolher alguns parâmetros, bem como a definição de qual algoritmo utilizar no SVM (Lorena and de Carvalho, 2003).

8.1.4 Exemplos

1. Vamos supor o seguinte conjunto de dados:

Ao observarmos no gráfico, podemos observar que os vetores de suporte são:

$$\{s_1 = (1, 2), s_2 = (4, 1), s_3 = (4, -1)\}$$

Agora, vamos inserir o valor 1 de entrada do *Bias*:

$$\{\hat{s}_1 = (1, 2, 1), \hat{s}_2 = (4, 1, 1), \hat{s}_3 = (4, -1, 1)\}$$

Precisamos agora encontrar o valor de α_i :

$$\alpha_1 \phi(s_1) \cdot \phi(s_1) + \alpha_2 \phi(s_2) \cdot \phi(s_1) + \alpha_3 \phi(s_3) \cdot \phi(s_1) = -1$$

$$\alpha_1 \phi(s_1) \cdot \phi(s_2) + \alpha_2 \phi(s_2) \cdot \phi(s_2) + \alpha_3 \phi(s_3) \cdot \phi(s_2) = +1$$

$$\alpha_1 \phi(s_1) \cdot \phi(s_3) + \alpha_2 \phi(s_2) \cdot \phi(s_3) + \alpha_3 \phi(s_3) \cdot \phi(s_3) = +1$$

E portanto:

$$\alpha_1 \hat{s}_1 \cdot \hat{s}_1 + \alpha_2 \hat{s}_2 \cdot \hat{s}_1 + \alpha_3 \hat{s}_3 \cdot \hat{s}_1 = -1$$

$$\alpha_1 \hat{s}_1 \cdot \hat{s}_2 + \alpha_2 \hat{s}_2 \cdot \hat{s}_2 + \alpha_3 \hat{s}_3 \cdot \hat{s}_2 = +1$$

$$\alpha_1 \hat{s}_1 \cdot \hat{s}_3 + \alpha_2 \hat{s}_2 \cdot \hat{s}_3 + \alpha_3 \hat{s}_3 \cdot \hat{s}_3 = +1$$

$$\left\{ \alpha_1 \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 4 \\ 1 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} + \alpha_3 \begin{pmatrix} 4 \\ -1 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} \right\} = -1$$

$$\left\{ \alpha_1 \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} \begin{pmatrix} 4 \\ 1 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 4 \\ 1 \\ 1 \end{pmatrix} \begin{pmatrix} 4 \\ 1 \\ 1 \end{pmatrix} + \alpha_3 \begin{pmatrix} 4 \\ -1 \\ 1 \end{pmatrix} \begin{pmatrix} 4 \\ 1 \\ 1 \end{pmatrix} \right\} = +1$$

$$\left\{ \alpha_1 \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} \begin{pmatrix} 4 \\ -1 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 4 \\ 1 \\ 1 \end{pmatrix} \begin{pmatrix} 4 \\ -1 \\ 1 \end{pmatrix} + \alpha_3 \begin{pmatrix} 4 \\ -1 \\ 1 \end{pmatrix} \begin{pmatrix} 4 \\ -1 \\ 1 \end{pmatrix} \right\} = +1$$

Resolvendo as matrizes, temos:

$$\alpha_1(1 + 4 + 1) + \alpha_2(4 + 2 + 1) + \alpha_3(4 - 2 + 1) = -1$$

$$\alpha_1(4 + 2 + 1) + \alpha_2(16 + 1 + 1) + \alpha_3(16 - 1 + 1) = +1$$

$$\alpha_1(4 - 2 + 1) + \alpha_2(16 - 1 + 1) + \alpha_3(16 + 1 + 1) = +1$$

E portanto:

$$6\alpha_1 + 7\alpha_2 + 3\alpha_3 = -1$$

$$7\alpha_1 + 18\alpha_2 + 16\alpha_3 = +1$$

$$3\alpha_1 + 16\alpha_2 + 18\alpha_3 = -1$$

Logo $\alpha_1 = 2,44$, $\alpha_2 = 2,83$ e $\alpha_3 = -2,06$. Encontrando $\vec{w} = \sum_{i=1}^n \alpha_i \hat{s}_i$:

$$\vec{w} = 2,44 \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} + 2,83 \begin{pmatrix} 4 \\ 1 \\ 1 \end{pmatrix} + -2,06 \begin{pmatrix} 4 \\ -1 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} 5,52 \\ 9,77 \\ 3,21 \end{pmatrix}$$

Portanto temos que $w = \begin{pmatrix} 2,44 \\ 2,83 \end{pmatrix}$ e $b = -2,06$. Com a equação $y = wx + b$ e todos os dados, podemos plotar o hiperplano.

8.2 Árvore de Decisão (*Decision Tree*)

A Árvore de Decisão - *Decision Tree* - é muito utilizada em algoritmos para classificação de dados, tem como objetivo construir classificadores que predizem classes baseadas nos valores de atributos de um *dataset* (análise supervisionada). Ela pode ser aplicada tanto em variáveis categóricas quanto contínuas de entrada e de saída. Na árvore de decisão, dividimos a população ou amostra em dois ou mais conjuntos homogêneos com base nos divisores mais significativos das variáveis de entrada. É um modelo fácil de compreensão, útil para explorar os dados e classificar os dados, sem restrição do tipo de seus dados e pode ser considerada como não paramétrica, porém precisa-se tomar cuidado em ocorrer um sobreajuste e a aplicação em variáveis contínuas pode perder informações.

Há muitos algoritmos de classificação que constroem árvores de decisão. Cada um pode ter melhor desempenho em determinada situação e outro algoritmo pode ser mais eficiente em outros tipos de situações, não há como apontar qual o melhor método. Ela é composta por:

1. Nó Raiz/Nodos: Representa a população ou uma amostra, podendo ser ainda dividido em dois ou mais conjuntos homogêneos;
2. Divisão e Arcos: É o processo de dividir um nó em dois ou mais sub-nós, gerando arcos provenientes destes nodos que recebem os valores possíveis para estes atributos;
3. Nó de Decisão: Quando um sub-nó é dividido em sub-nós adicionais;
4. Folha/Nó de Término: Os nós não divididos são denominados de Folha ou Nó de Término, representam as diferentes classes de um conjunto de treinamento;
5. Poda: O processo de remover sub-nós de um nó de decisão é chamado poda. Existe técnicas para avaliar o bom momento de podar o galho (sub-nó) da árvore.

Segue abaixo uma demonstração da ramificação da árvore:

feitos/Livro ML/bookdown-demo-master/Figuras/decisiontree.bb



Figure 8.10: Terminologia de Árvore de Decisão.

Um nó que é dividido em sub-nós é chamado de nó pai. Os sub-nós são os nós filhos do nó pai.

Em geral são determinadas regras em seu algoritmo. As regras são escritas considerando o trajeto do nó raiz até uma folha da árvore. Por elas tenderem a crescer muito, de acordo com algumas aplicações, elas são muitas vezes substituídas pelas regras. Isto acontece em virtude das regras poderem ser facilmente modularizadas. Uma regra pode ser compreendida sem que haja a necessidade de se referenciar outras regras (Ingargiola, 1996).

Sua função é de particionar recursivamente um conjunto de treinamento, até que cada subconjunto obtido deste particionamento contenha casos de uma única classe. Com esta finalidade, basicamente a árvore de decisão verifica e compara a distribuição de classes durante a construção da árvore. Após finalizar a árvore, sua saída são dados organizados de maneira compacta, que são utilizados para classificar novos casos (Holsheimer and Siebes, 1994). De início não é utilizado nenhum modelo estatístico nela, apenas classifica os atributos de acordo com as amostras provenientes. Mas atualmente utilizam técnicas estatísticas para aperfeiçoar o modelo e avaliar seus resultados (Shiba et al., 2005).

Geralmente utilizam-na para classificação (variáveis categóricas), mas também é possível para a Análise de Regressão (variáveis contínuas) que já vimos. Para as árvores de regressão são utilizadas quando a variável dependente é contínua. O valor obtido pelos nós de término nos dados de treinamento é o valor médio das observações. Para classificação o obtido pelos nós de término nos dados de treinamento é a moda, ou seja, a observação mais recorrente no conjunto de dados. Ambos processos continuam o processo de divisão até atingir algum critério fornecido pelo pesquisador. Mas como fazemos esta divisão para poder-

mos classificar?

A decisão de como fazer estas divisões dos nós pode influenciar muito na precisão do algoritmo e portanto, seus resultados. Pode-se utilizar pelo qui-quadrado (ver 5.2), por ganho de informação (ver 5.1.1), Índice de Gini, redução de variância, entre diversos outros.

Ao Índice de Gini, aplicado no sistema *Classification and Regression Trees* (CART)(Breiman et al., 1984), mede a impureza de uma partição de dados, ela basicamente nos mostra que se selecionarmos aleatoriamente dois atributos de uma população ou amostra, ambos devem ter a mesma classe e a soma da probabilidade será se esta amostra/população for pura.

É utilizada portanto para variáveis categóricas como “Sucesso” e “Fracasso”, ou seja, aplica-se em divisões binárias. Quanto maior for este índice, mais homogêneo será. Foi criado como uma medida de variância para dados categóricos (Light and Margolin, 1971), é expresso como:

$$\begin{aligned} G(p_1, p_2, \dots, p_j) &= \sum_j p_j \sum_{i \neq j} p_i \\ &= \sum_j p_j (1 - p_j) \\ &= 1 - \sum_j p_j^2 \end{aligned} \tag{8.22}$$

Para evitarmos problemas como **overfitting** por exemplo, que ocorre quando o seu modelo aprendeu tão bem as relações existentes dos conjuntos de dados para treino que acabou apenas decorando esses dados (será apresentado em 9.1). Existe diversos meios que variam de acordo com propostas de diferentes pesquisas. Pode-se definir um número mínimo de amostras que são necessárias em um nó para se fazer a divisão, ou delimitar amostras mínimas para o nó terminal e o máximo de nós terminais, determinar a profundidade máxima da árvore (o quanto ela vai ramificar e expandir-se), atentar ao quanto de recurso será utilizado para treinarmos o modelo e quanto irão para serem testados (visto que não pode ser o mesmo conjunto de dados para ambos pois ela apenas iria decorar e replicar).

A poda (pós-poda) de uma árvore é importante para que se verifique a melhor divisão e chegue até a condição de parada especificada. Um exemplo que gosto muito e creio que facilitará para a compreensão da poda (ANALYTICS VIDHYA, 2016) : suponha que há duas pistas, você está em seu veículo verde na pista da direita com uma certa quantidade de carros em sua frente movendo a aproximados $80km/h$ cada. Na pista da esquerda encontra-se dois caminhões de entrega de encomendas à apenas $30km/h$ cada. Caso você vá pela esquerda irá alcançar o carro à frente, podendo passar até chegar atrás do caminhão e irá manter seu veículo a $30km/h$ procurando desesperadamente encontrar alguma oportunidade de voltar para a direita. Entretanto todos os outros carros ultrapassam você.

Seria uma ótima escolha caso você precisasse ultrapassá-los em poucos segundos. Mas a um prazo maior poderia ser uma escolha bem ruim. Esta é a diferença entre a árvore de decisão sem e com a poda. O algoritmo de árvore de decisão com restrições não irá visualizar os caminhões a frente e adotará o trajeto interessante naquele momento e iria optar pela esquerda. Porém quando utilizamos a poda, estamos observando alguns passos à frente antes de tomarmos decisões de que lado iríamos. Ao observar que para a esquerda é ruim, poda-se este galho.

feitos/Livro ML/bookdown-demo-master/Figuras/poda.bb

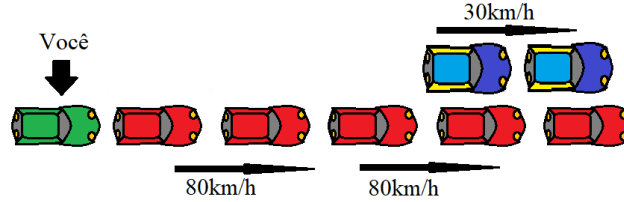


Figure 8.11: Desenho de veículos em uma pista. Adaptado de (ANALYTICS VIDHYA, 2016).

Para que se faça a poda, inicia-se o algoritmo de árvore de decisão até uma grande profundidade em sua ramificação; analisa-se a parte inferior da árvore (os filhos e seus resultados) removendo folhas que estão dando retornos negativos quando comparadas ao topo (comparando o erro de cada nó e a soma dos erros dos nós descendentes, ou algum outro método estatístico similar). Tem também algoritmos de pré-poda que buscam não particionar mais o conjunto de treinamento com algum determinado critério como não ultrapassar a uma determinada variação de ganho de informação, parar se todas as instâncias pertencem à mesma classe, valores de atributos iguais, significância estatística, redução de erro, etc. Os algoritmos de “pós-poda” são mais lentos e pode haver um custo bem maior que o de “pré-poda”, mas são mais confiáveis.

Um dos cálculos mais utilizados para a poda da árvore é a **taxa de erro** que representa a razão entre o número de casos com classificação errada (c_e) e o número de casos classificados corretamente (cc) pela partição, caso a taxa de erro aumente conforme a ramificação, irá podar:

$$E(T) = \frac{c_e}{c_e + cc} \quad (8.23)$$

Na seção

8.2.1 Exemplos

Aos exemplos de Árvore de Decisão, vamos tomar como base da literatura de (ANALYTICS VIDHYA, 2016).

Vamos supor uma amostra com 30 alunos com duas características cada: Sexo (meninos e meninas), Classe (I e II). Temos como propósito elaborar um modelo de árvore de decisão para prevermos quais alunos iriam jogar tênis durante o intervalo. Portanto precisamos classificar estes alunos com base nestas duas características. Supondo valores pré-estabelecidos dos alunos, a árvore segregará os alunos com base nestas variáveis e identificará a variável que cria os melhores conjuntos homogêneos e heterogêneos entre si.

feitos/Livro ML/bookdown-demo-master/Figuras/decision.bb

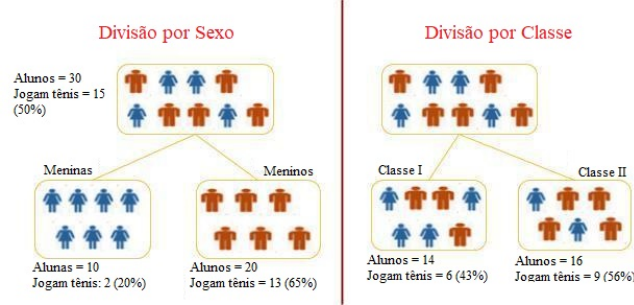


Figure 8.12: Divisão de alunos que jogam tênis, traduzido de (ANALYTICS VIDHYA, 2016).

1. Índice de Gini - Usando o Índice de Gini, vamos verificar qual divisão produz sub-nós mais impuros. Dentro de 10 meninas, apenas duas jogam tênis no intervalo, ou seja 20%; aos 20 meninos, 13 que equivale a 65%. Ressalta-se que os dados foram arredondados em duas casas para facilitar a exemplificação.

$$\text{sub-nó Meninas} = 1 - (0,2^2 + 0,8^2) = 0,320$$

$$\text{sub-nó Meninos} = 1 - (0,65^2 + 0,35^2) = 0,455$$

A medição de impureza para as Meninas é 0,320. Como são duas possibilidades (Menina *versus* Menino), podemos dividir por 0,5 para uma compreensão mais intuitiva, obteremos 0,64. Caso fosse 0,5/0,5 = 1, significaria que o agrupamento é o mais impuro possível, pois é muito distribuído e não apenas uma variável predominante no conjunto amostral. Ao caso dos Meninos (0,455), obtemos um valor de 0,91 (bem impuro) ao dividirmos por 0,5.

Vamos analisar agora o valor de Gini dado que foi selecionado 10 meninas e 20 meninos de uma amostra de 30 alunos. Um valor ponderado:

$$Gini = \frac{10}{30} \cdot (0,20^2 + 0,80^2) + \frac{20}{30} \cdot (0,65^2 + 0,35^2) = 0,59$$

$$\text{impureza: } 1 - 0,59 = 0,41$$

Da mesma forma, calculamos para a classe I e II:

$$\text{sub-nó Classe I} = 1 - (0,43^2 + 0,57^2) = 0,49$$

$$\text{sub-nó Classe II} = 1 - (0,56^2 + 0,44^2) = 0,49$$

O valor ponderado:

$$Gini = \frac{14}{30} \cdot (0,43^2 + 0,57^2) + \frac{16}{30} \cdot (0,56^2 + 0,44^2) = 0,51$$

$$\text{impureza: } 1 - 0,51 = 0,49$$

Como Sexo possui um Índice de Gini maior que da Classe, ou uma impureza menor, o algoritmo irá fazer com que a divisão do nó ocorra em gênero. Caso houvesse mais variáveis como Altura, iria comparar entre as três. O de maior valor de Índice seria a nova referência de ramificação (se tornando um nó de divisão) caso haja outras características para mais ramificações.

2. Ganho de Informação - Inicialmente precisamos calcular a entropia do nó pai e em seguida calcular a entropia de cada nó individual da divisão e a média ponderada de todos os sub-nós disponíveis na divisão. Sabemos que de 30 alunos, 15 irão jogar tênis e 15 não, logo:

$$\text{Entropia para o nó pai: } H(A) = - \sum p(A) \log_2(p(A))$$

$$H(A) = - \left(\frac{15}{30} \log_2(15/30) + \frac{15}{30} \log_2(15/30) \right) = 1$$

Portanto o nó é totalmente impuro, o que faz sentido, pois ele é exatamente 50% dos dados distribuído em jogar tênis e outros 50% de não jogar no intervalo (maior distribuição possível entre as duas possibilidades).

Para o nó feminino, 2 que irão jogar e 8 que não irão jogar tênis num total de 10 meninas a entropia será $-((2/10)\log_2(2/10) + (8/10)\log_2(8/10)) = 0,72$ e para nó masculino com 13 que irão e 7 não no total de 20 meninos temos: $-((13/20)\log_2(13/20) + (7/20)\log_2(7/20)) = 0,93$.

Portanto o Ganho de Informação será:

$$\text{Ganho de Informação}(D, T) = \text{entropia}(D) - \sum_{i=1}^k \frac{|D_i|}{|D|} \cdot \text{entropia}(D_i)$$

$$\begin{aligned} & \text{Entropia}_{\text{pai}}(D) - \sum_{i=1}^k \text{peso} \cdot \text{Entropia}_{\text{filho}} \\ &= 1 - ((10/30) \cdot 0,72 + (20/30) \cdot 0,93) = 0,14 \end{aligned}$$

Do mesmo modo, vamos calcular em Classe I, $-((6/14)\log_2(6/14) + (8/14)\log_2(8/14)) = 0,99$ e para nó Classe II, $-((9/16)\log_2(9/16) + (7/16)\log_2(7/16)) = 0,99$.

E o Ganho de Informação:

$$G.I = 1 - ((14/30).0,99 + (16/30).0,99) = 0,01$$

Como o Ganho de Informação de Sexo é maior que o de Classe (menos impuro), a árvore irá iniciar sua ramificação a partir da característica Sexo. Caso houvesse mais variáveis como Altura, iria comparar entre as três o G.I e a de maior G.I seria considerado a nova “Pai” para que se possa recalculer caso haja outras características para mais ramificações.

3. Exemplo numérico - Adilson tem uma lanchonete e recebe cerca de 300 clientes por mês. Cada cliente gasta em média R\$100,00. Uma concorrente vai abrir uma nova unidade próximo de seu estabelecimento, o que reduzirá seu número de clientes a não ser que Adilson amplie seu comércio. Considerando que está apertado financeiramente, está na dúvida se vale este investimento contabilizando num prazo de 5 meses. A análise foi: caso ele investisse R\$40.000,00, as chances de ele obter 330 clientes por mês é de 30% e de obter 380 é 70%. Se Adilson não optar expandir, não irá gastar nada porém com a concorrente, irá ter uma probabilidade de 60% de atender 250 clientes por mês e de 40% de atender 290. Qual seria sua decisão, tomando como base o algoritmo de Árvore de Decisão?

feitos/Livro ML/bookdown-demo-master/Figuras/decisionlanche.bb

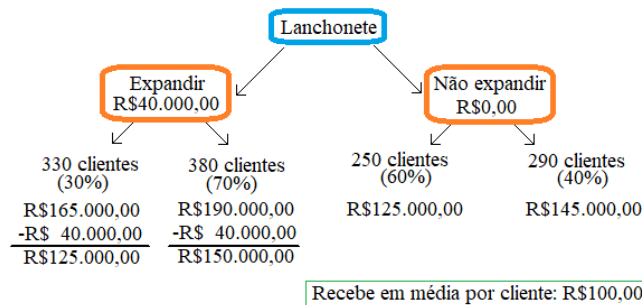


Figure 8.13: Árvore de Decisão sobre o lucro da lanchonete.

Pode-se calcular pelo Valor Esperado (Média). Para a primeira situação, gastando R\$40.000,00, em 5 meses seu valor esperado será:

$$V.E = (0,3 \cdot 125.000,00) + (0,7 \cdot 150.000,00) = 142.500,00$$

Ao segundo caso, em que Adilson não irá optar o investimento em seu estaleci-

mento:

$$V.E = (0,6 \cdot 125.000,00) + (0,4 \cdot 145.000,00) = 133.000,00$$

Portanto Adilson tem uma probabilidade maior de escolher investir os R\$40.000,00 para ampliar sua lanchonete. Lembrando que foi utilizado o critério de médias, podendo haver diversos outros.

8.3 Elastic Net

8.4 KNN

8.5 Análise de Componentes Principais

A Análise de Componentes Principais, popularmente conhecida como ACP ou PCA (*Principal Component Analysis*), em inglês, foi introduzida por Pearson (1901) e fundamentada no artigo de Hotelling (1933). É uma **análise multi-variada** que tem como objetivo explicar a estrutura de variância e covariância de um vetor aleatório, composto por p -variáveis aleatórias, através da construção de combinações lineares das variáveis originais que são chamadas de componentes principais e não correlacionadas entre si (Mingoti, 2007). É uma técnica bastante utilizada em diversas áreas do conhecimento, como a biologia, a agronomia, a zootécnica, a ecologia, a engenharia florestal, a medicina, a economia, entre outras áreas. Muitos sugerem o seu uso quando o volume de dados ou variáveis é grande possibilitando reduzir a dimensão da matriz de dados que compõem o conjunto de variáveis resposta com apenas poucos componentes, ou seja, p variáveis originais substituídas por k (sendo $k < p$) componentes principais não correlacionadas.

Vamos supor um conjunto de dados em apenas duas dimensões (x, y) e que pode ser plotado em um plano cartesiano. Podemos verificar pelo seu comportamento que possuem alta correlação positiva.

Mas se quisermos descobrir a variação do conjunto de dados, o ACP busca encontrar um novo sistema de coordenadas em que cada ponto tem um novo valor (x, y) . Os eixos não representam algo físico, mas representam combinações de x e y que denominamos “**componentes principais**”, escolhidas para analisar a variação do eixo. Observe que rotacionamos o gráfico na Figura 8.15 e que após a ACP, podemos verificar a possibilidade de descartar a componente referente ao eixo y , visto que a componente do eixo x explica 99,30% da variação total dos dados, ou seja, o primeiro componente tem uma maior dispersão (variância). Possibilitando pela componente principal do eixo x , analisar e até mesmo classificar as observações, como por exemplo, a observação 1 e 2 como um conjunto e a 3, 4 e 5 como um segundo conjunto.

Com mais dimensões, o ACP torna-se ainda mais útil pois possibilita observarmos o conjunto de dados num melhor ângulo.

feitos/Livro ML/bookdown-demo-master/Figuras/pca1.bb

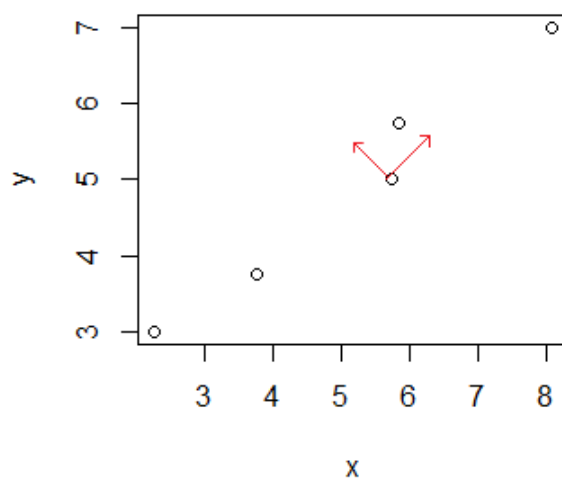


Figure 8.14: Gráfico bidimensional x por y .

feitos/Livro ML/bookdown-demo-master/Figuras/pca2.bb

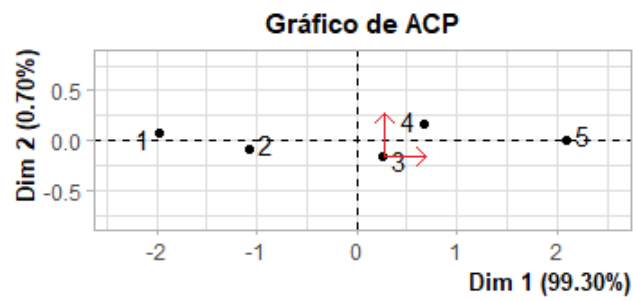


Figure 8.15: Gráfico de x por y rotacionado.

feitos/Livro ML/bookdown-demo-master/Figuras/pca3.bb

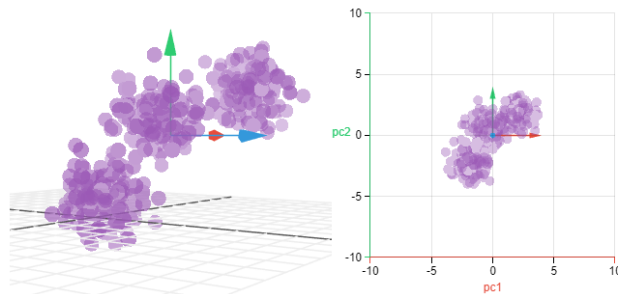


Figure 8.16: Gráfico tridimensional, em Powell, Victor and Lehe, Lewis (2014).

Portanto, a ACP assume que os dados originais estão representados por características (variáveis) correlacionadas com o objetivo de transformar essas variáveis em novas (componentes principais) por meio de mudança de base do espaço vetorial que não sejam correlacionadas entre si e que estas novas variáveis (menores que as originais) retenha a maior parte da variação apresentada pelas originais, tornando possível a classificação.

A suposição de normalidade não é requisito para sua técnica, mas ainda sim é conveniente padronizar (6.2.2) cada variável, permitindo que todas as variáveis tenham o mesmo peso para evitarmos viés de escala (Hongyu et al., 2016). A padronização das variáveis do vetor pelas respectivas médias e desvios padrões, gera novas variáveis centradas em zero e com variâncias iguais a 1. Assim, as componentes principais são determinadas a partir da matriz de covariâncias das variáveis originais padronizadas (Mingoti, 2007).

Agora que sabemos o que é ACP, vamos apresentar alguns conceitos de Álgebra Linear e Estatísticas para compreendermos como é aplicado este método.

8.5.1 Autovalores e Autovetores

Caso ainda não tenha muito contato com a Álgebra Linear, recomendo buscar algumas literaturas a respeito. Em 1.2 encontra-se sobre Escalar, Vetores, Espaço Vetorial e Transformação Linear que serão tratadas neste tópico.

Dado uma matriz $A_{m \times n}$ que define uma transformação linear (não muda sua dimensão), existem vetores onde sua orientação não é afetada por esta transformação, os **autovetores**. Na Figura 8.17, u é um autovetor e V não.

feitos/Livro ML/bookdown-demo-master/Figuras/autovetor.bb

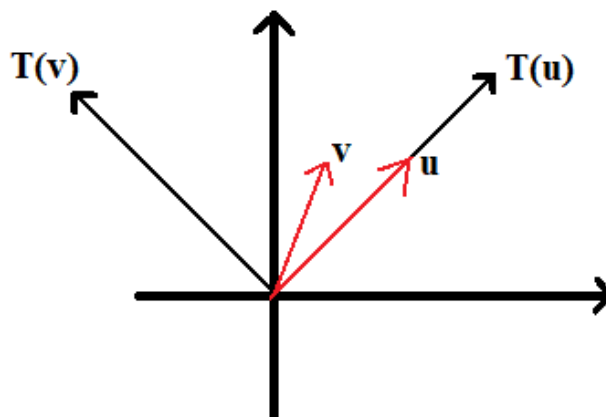


Figure 8.17: u é um autovetor de T , porém V não.

Um vetor é dito ser autovetor da matriz $A_{m \times n}$ se a transformação linear deste vetor $T(u)$ é colinear a este vetor, ou seja, $A_{m \times n} \vec{u} = \lambda \vec{u}$. Sendo que λ é um

escalar e chamado de **autovalor** da matriz correspondente ao autovetor. Para encontrarmos o autovetor:

$$\begin{aligned} A_{m \times n} \vec{u} &= \lambda \vec{u} \\ A_{m \times n} \vec{u} - \lambda \vec{u} &= 0 \\ (A_{m \times n} - \lambda I) \vec{u} &= 0 \end{aligned} \quad (8.24)$$

esta equação tem solução trivial, ou seja, diferentes da nula ($\vec{v} \neq 0$) se e somente se, seu determinante é zero. Conhecido como **Equação característica** e sua solução são os **autovalores**:

$$\text{Eq. Característica } \det(A_{m \times n} - \lambda I) = 0 \quad (8.25)$$

Note também que toda transformação linear (matriz) em um espaço vetorial complexo (números imaginários) tem, pelo menos, um autovetor (real ou complexo).

8.5.1.1 Exemplo

1. Vamos considerar um operador linear $T : R^2 \rightarrow R^2$. Com $T(x, y) = (4x + 5y, 2x + 2y)$. Quais são os autovalores a matriz $A = \begin{bmatrix} 4 & 5 \\ 2 & 2 \end{bmatrix}$?

Vamos resolver a equação característica $\det(A_{m \times n} - \lambda I) = 0$.

$$\det(A_{m \times n} - \lambda I) = \begin{vmatrix} 4 & 5 \\ 2 & 2 \end{vmatrix} - \lambda \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} = \begin{vmatrix} 4 - \lambda & 5 \\ 2 & 2 - \lambda \end{vmatrix}$$

Com $\det(A_{m \times n} - \lambda I) = 0$:

$$(4 - \lambda)(2 - \lambda) - 10 = 0$$

$$\lambda^2 - 6\lambda - 2 = 0 \text{ resolvendo a equação: } \lambda_1 \approx 6,32 \text{ e } \lambda_2 \approx -0,32$$

2. Considere a matriz $A = \begin{bmatrix} 1 & 0 \\ 1 & -2 \end{bmatrix}$.

Vamos resolver a equação característica $\det(A_{m \times n} - \lambda I) = 0$.

$$\det(A_{m \times n} - \lambda I) = \begin{vmatrix} 1 & 0 \\ 1 & -2 \end{vmatrix} - \lambda \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} = \begin{vmatrix} 1 - \lambda & 0 \\ 1 & -2 - \lambda \end{vmatrix}$$

Resolvendo este sistema, obtemos os autovalores:

$$\lambda_1 = 1 \quad \lambda_2 = -2$$

Vamos encontrar agora seus respectivos autovetores, lembrando que $A_{m \times n} \vec{u} = \lambda \vec{u}$:

Primeiro encontrar os autovetores de $\lambda_1 = 1$:

$$A_{m \times n} \vec{u} = \lambda \vec{u}$$

$$\begin{bmatrix} 1 & 0 \\ 1 & -2 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = 1 \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

$$x = x$$

$$x - 2y = y$$

$$\text{logo: } x = 3y$$

Portanto em $\lambda_1 = 1$ será $X = \begin{bmatrix} 3y \\ y \end{bmatrix}$, com o autovetor de $y = 1$ e $x = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$.

Agora para o segundo autovalor $\lambda_2 = -2$:

$$A_{m \times n} \vec{u} = \lambda \vec{u}$$

$$\begin{bmatrix} 1 & 0 \\ 1 & -2 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = -2 \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

$$x = -2x$$

$$x - 2y = -2y$$

$$\text{logo: } x = 0$$

Portanto em $\lambda_2 = -2$ será $y = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ e $x = 0$.

Logo o primeiro autovetor será $X = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$ e o segundo $y = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

8.5.2 Estatísticas

Alguns conceitos de Estatísticas são fundamentais para que se entenda a ACP:

- **Covariância x Correlação:** como apresentado em 1.2, a covariância é semelhante à correlação (ver 6.2.2) entre duas variáveis, no entanto, elas diferem que os coeficientes de correlação são padronizados. Isso faz com que um relacionamento linear varie entre $-1 \leq \rho \leq 1$. A correlação mede tanto a força como a direção da relação linear entre duas variáveis. Ao caso da covariância os valores não são padronizados. Assim, a covariância pode variar de $-\infty \leq Cov(x, y) \leq \infty$ demonstrando quanto x e y mudam juntas. Portanto o valor para uma relação linear ideal depende muito dos dados. Como os dados não são padronizados, é difícil determinar a força da relação entre as variáveis. Note que o coeficiente de correlação é uma função da covariância:

$$\rho_{x,y} = \frac{\text{cov}(x,y)}{\sigma_x \sigma_y}$$

Uma covariância positiva sempre resulta em uma correlação positiva e uma covariância negativa sempre resulta em uma correlação negativa.

Quando temos um vetor de n variáveis em vez de apenas duas, iremos obter uma matriz de covariâncias ou correlação. Contendo em sua diagonal a variância σ^2 , pois $\text{cov}(x_i, x_i) = \sigma^2(x_i)$, por exemplo:

$$\begin{bmatrix} \text{cov}_{1,1} & \text{cov}_{1,2=2,1} & \text{cov}_{1,3=3,1} \\ \text{cov}_{1,1=2,1} & \text{cov}_{2,2} & \text{cov}_{2,3=3,2} \\ \text{cov}_{3,1=1,3} & \text{cov}_{2,3=3,2} & \text{cov}_{3,3} \end{bmatrix} = \begin{bmatrix} \text{var}_1 & \text{cov}_{1,2=2,1} & \text{cov}_{1,3=3,1} \\ \text{cov}_{1,1=2,1} & \text{var}_2 & \text{cov}_{2,3=3,2} \\ \text{cov}_{3,1=1,3} & \text{cov}_{2,3=3,2} & \text{var}_3 \end{bmatrix}$$

8.5.3 A ACP

Agora que compreendemos alguns conceitos importantes, podemos entender melhor a metodologia da ACP. Assumindo que os dados originais estão representados por variáveis correlacionadas (etapa de pré-processamento), ou seja, não independentes. Vamos ao objetivo de transformar essas p variáveis em outras novas k (com $k < p$) de ordem decrescente de variabilidade e que não sejam correlacionadas e que as primeiras novas variáveis retenham a maior parte da variação apresentadas pelas originais a fim de podermos classificá-los.

Dado um vetor $X = (X_1, X_2, \dots, X_p)'$ aleatório de p variáveis originais com vetor de médias $\mu = (\mu_1, \mu_2, \dots, \mu_p)$ e matriz de covariâncias $A_{m \times n}$ e $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p$ os autovalores da matriz de covariâncias, com seus respectivos autovetores normalizados $e'_i = e_1, e_2, \dots, e_p$. O primeiro componente principal y_1 , como dito que deve ser ordem decrescente de variabilidade, será uma combinação linear do vetor aleatório X de forma que a variância $\text{var}(y_1) = \sigma_{y_1}^2$ seja a máxima (maior possível), ou melhor, precisamos encontrar um vetor e_1 tal que $y_1 = (e_1)^T X$ e $\text{var}(y_1 = (e_1)^T X)$ seja máxima. De mesmo modo para y_2 e um vetor e_2 e assim sucessivamente para p variáveis em seu banco de dados.

Portanto a matriz dos autovetores normalizados da matriz de covariância $A_{m \times n}$ é:

$$O_{m \times n} = \begin{bmatrix} e_{11} & e_{21} & \cdot & \cdot & \cdot & e_{p1} \\ e_{12} & e_{22} & \cdot & \cdot & \cdot & e_{p2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ e_{1p} & e_{2p} & \cdot & \cdot & \cdot & e_{pp} \end{bmatrix} = [e_1, e_2, \dots, e_p] \quad (8.26)$$

E dos autovalores:

$$\Lambda_{mxn} = \begin{bmatrix} \lambda_1 & & 0 \\ & \lambda_2 & \\ 0 & & \lambda_n \end{bmatrix} \quad (8.27)$$

Portanto, as variáveis aleatórias que constituem o vetor Y não são correlacionadas entre si. Com isso, a ideia de utilizar combinações lineares em Y como forma de representar a estrutura de covariâncias do vetor X torna-se interessante, a fim de reduzir o espaço de variáveis de p para $k < p$ dimensões. Os vetores X e Y terão a mesma variância total e generalizada, com Y de vantagem de não haver variáveis correlacionadas e facilitando na interpretação conjunta delas (análise multivariada). Portanto:

$$Y_j = e'_j X = e_{j1}X_1 + e_{j2}X_2 + \dots + e_{jp}X_p \quad (8.28)$$

A esperança e variância:

$$E[Y_j] = e'_j \mu = e_{j1}\mu_1 + e_{j2}\mu_2 + \dots + e_{jp}\mu_p \quad (8.29)$$

$$Var[Y_j] = e'_j A_{mxn} e_j = \lambda_j \quad (8.30)$$

Lembrando que $Cov[Y_j, Y_k] = 0$, $j \neq k$ e que cada autovalor λ_j representa a variância de uma componente principal Y_j . A primeira componente terá a maior variabilidade e a última menor.

Para calcularmos a correlação estimada entre a j -ésima componente principal e a variável aleatória X , podemos expressar:

$$r_{Y_j, X_i} = \frac{e_{ji} \sqrt{\lambda_j}}{\sqrt{\sigma}} \quad (8.31)$$

De mesmo modo para tratarmos de amostras, são trabalhados com \hat{Y}_j e \hat{X}_j e seus respectivos, autovetores, autovalores, matriz de covariância amostral e correlação.

Os maiores autovalores são os que orientam o sinal, os demais podem ser descartados. Porém quantos componentes principais devemos utilizar? Precisamos verificar a proporção da variação total dos dados originais que uma componente pode explicar, a partir disso selecionarmos. Lembrando que cada autovalor λ_i refere-se a $var(y_i)$.

Para calcularmos a variação total, expressa-se pela somatória de todos os autovalores:

$$\sum_j \lambda_j \quad (8.32)$$

Portanto, para analisar cada i componente, ou seja, cada autovalor (variação “explicada” por cada componente):

$$p_i = \frac{\lambda_j}{\sum_j \lambda_j} \quad (8.33)$$

Sendo geralmente escolhido as componentes com seus respectivos autovalores que explicam entre 70%-90% segundo alguns pesquisadores. Outros como Kaiser (1960), propõe aceitar, observando diretamente, somente os autovalores iguais ou superiores à unidade.

Importante: sobre utilizar matriz de covariância ou de correlação depende muito das fundamentações teóricas e recomendações dos pesquisadores. Em geral, utiliza-se a matriz de correlação (quando padronizamos e elaboramos a matriz) ao caso de padronizar escalas distintas que podem viesar, como por exemplo, medidas de distância e de peso.

Caso esteja utilizando software para a análise, dependendo do software utilizado com seu determinado modelo de formulação de componentes principais, pode ocorrer essa troca de sinal que nada mais é do que uma reflexão em relação ao eixo, uma rotação em seu espaço vetorial n -dimensional em torno da origem, poderá ocasionar uma “rotação” em torno do eixo. Tratando de álgebra linear e suas combinações lineares, a combinação poderá possuir soluções diferentes que diferem apenas o sinal.

Resumo geral: É comum o pesquisador trabalhar com um volume de dados muito grande e que estão muitas vezes correlacionadas. A Análise de Componentes principais busca explicar a estrutura de variância e covariância de um vetor aleatório com p variáveis, possibilitando por meio da combinação linear deste vetor aleatório novas componentes (denominada componente principal) com menos variáveis ($k < p$) que o conjunto de dados original e não correlacionadas de modo que estas componentes principais retenha a maior parte da variação apresentada pelas originais, possibilitando classificarmos o conjunto de dados e até mesmo descartar variáveis que podem ser redundantes ou não importantes. Utiliza-se de fundamentações teóricas de Autovetores e Autovalores para que se encontre um novo sistema de coordenadas com novos pontos a partir das originais, pode-se dizer que rotacionamos para que se visualize num “novo ângulo”, para descobrir e avaliar em ordem decrescente a variação (matriz de covariância do vetor aleatório) deste conjunto de dados. Os passos:

1. Calcular a Matriz de Correlação amostral $R_{m \times n}$ ou Covariância amostral $S_{m \times n}$ do vetor aleatório de p variáveis.
2. Encontrar λ_i autovalores da matriz.
3. Encontrar seus respectivos e_i autovetores.
4. Aplicar outras análises caso necessite (como correlação da componente e a variável), interpretar os dados e selecionar as novas variáveis.

8.5.4 Exemplos

Tomando como base exemplos de Mingoti (2007).

1. Matriz de covariância amostral

A Tabela apresenta dados relativos as 12 empresas no que se refere a 3 variáveis (medidas em unidades monetárias): ganho bruto (X_1), ganho líquido (X_2) e o patrimônio acumulado (X_3):

Empresas	Ganho Bruto(X_1)	Ganho Líquido(X_2)	Patrimônio Líquido(X_3)
E1	9893	564	17689
E2	8776	389	17359
E3	13572	1103	18597
E4	6455	743	8745
E5	5129	203	14397
E6	5432	215	3467
E7	3807	385	4679
E8	3423	187	6754
E9	3708	127	2275
E10	3294	297	6754
E11	5433	432	5589
E12	6287	451	8972

Após calcularmos suas covariâncias (recomendo o leitor calcular e verificar e atentar que por ser exemplificação, passível de ocorrência de arredondamento dos valores), obtemos a matriz de covariância amostral:

	Ganho Bruto (X_1)	Ganho Líquido (X_2)	Patrimônio Líquido (X_3)
Ganho Bruto (X_1)	9550608,6	706121,1	14978232,5
Ganho Líquido (X_2)	706121,1	76269,5	933915,1
Patrimônio Líquido (X_3)	14978232,5	933915,1	34408113,0

Para calcularmos os autovalores:

$$det(A_{m \times n} - \lambda I) = 0$$

$$\begin{bmatrix} 9550608,6 - \lambda & 706121,1 & 14978232,5 \\ 706121,1 & 76269,5 - \lambda & 933915,1 \\ 14978232,5 & 933915,1 & 34408113,0 - \lambda \end{bmatrix} = 0$$

Resolvendo o sistema, obtemos os seguintes autovalores das componentes principais:

$$\lambda_1 = 38018192,2 \quad \lambda_2 = 2327881,5 \quad \lambda_3 = 19334,8$$

Para encontrarmos a porcentagem da variância explicada por cada auto valor:

$$\% \lambda_1 = \frac{38018192,2}{38018192,2 + 2327881,5 + 19334,8} \cdot 100\% = 94,2\%$$

$$\% \lambda_2 = \frac{2327881,5}{38018192,2 + 2327881,5 + 19334,8} \cdot 100\% = 5,77\%$$

$$\% \lambda_3 = \frac{19334,8}{38018192,2 + 2327881,5 + 19334,8} \cdot 100\% = 0,048\%$$

Portanto, podemos descartar o segundo e o terceiro componente principal, pois o primeiro explica cerca de 94,2%.

Por fim os autovetores podem ser calculados:

$$A_{m \times n} \vec{u} = \lambda \vec{u}$$

Com $A_{m \times n}$ a matriz de covariância amostral, u o autovetor e λ os respectivos autovalores dos autovetores.

$$\begin{bmatrix} \vec{u}_1 \\ \vec{u}_2 \\ \vec{u}_3 \end{bmatrix} \begin{bmatrix} 9550608,6 & 706121,1 & 14978232,5 \\ 706121,1 & 76269,5 & 933915,1 \\ 14978232,5 & 933915,1 & 34408113,0 \end{bmatrix} = \lambda_i \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

substituindo os autovalores:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \begin{bmatrix} 9550608,6 & 706121,1 & 14978232,5 \\ 706121,1 & 76269,5 & 933915,1 \\ 14978232,5 & 933915,1 & 34408113,0 \end{bmatrix} = \begin{bmatrix} 0,942 & 0 & 0 \\ 0 & 0,0577 & 0 \\ 0 & 0 & 0,0048 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

Teremos os autovetores:

	Autovetor Ganho Bruto (u_1)	Autovetor Ganho Líquido (u_2)	Autovetor Patrimônio Líquido (u_3)
Autovetor Ganho Bruto (u_1)	0,425	0,900	-0,099
Autovetor Ganho Líquido (u_2)	0,028	0,096	0,995
Autovetor Patrimônio Líquido (u_3)	0,905	-0,426	0,016

Com os autovetores, podemos elaborar as três componentes principais:

$$\hat{y}_1 = 0,425(\text{GanhoBruto}) + 0,028(\text{GanhoLiquido}) + 0,905(\text{PatrimnioLiquido})$$

$$\hat{y}_2 = 0,900(\text{GanhoBruto}) + 0,096(\text{GanhoLiquido}) - 0,429(\text{PatrimnioLiquido})$$

$$\hat{y}_3 = -0,099(\text{GanhoBruto}) + 0,995(\text{GanhoLiquido}) + 0,016(\text{PatrimnioLiquido})$$

Determinada as componentes principais, podemos obter seus valores numéricos (**escores**) para cada elemento amostral. Basicamente substituímos os valores originais nas funções encontradas de componentes principais (\hat{y}_1, \hat{y}_2 e \hat{y}_3):

Empresas	CP_1	CP_2	CP_3
E1	8857,59	-165,27	-90,18
E2	8079,36	-1046,65	-158,93
E3	11257,93	2810,25	96,18
E4	-690,80	566,19	284,23
E5	3844,09	-3084,94	-30,40
E6	-5915,42	1841,62	-224,93
E7	-5504,97	-119,93	124,81
E8	-3796,38	-1367,83	-0,64
E9	-7729,15	789,46	-160,88
E10	-3848,18	-1473,28	121,59
E11	-3989,16	960,15	25,13
E12	-564,92	290,23	14,02

Podemos observar que a empresa E9 possui o menor desempenho, e as E1, E2 e E3 os melhores. Entenda que não necessariamente o sinal de negativo é sempre ser um pior valor, isso depende da pesquisa e da interpretação do sinal ou como em caso de autovetores, indica a rotação. Para analisarmos por gráfico não é recomendável utilizar neste caso, devido que são valores bem grandes para serem inseridos. No caso de Matriz de correlação, que serão padronizados os dados, podemos visualizar melhor.

E a correlação entre as componentes principais e as variáveis originais:

	CP 1	CP 2	CP 3
Ganho Bruto (X_1)	0,8859	0,4639	-0,0047
Ganho Líquido (X_2)	0,6450	0,5569	0,5232
Patrimônio Líquido (X_3)	0,9933	-0,1156	0,0004

Por meio da observação de seus resultados podemos analisar que:

- A primeira componente possui alta correlação-positiva com todas as três

variáveis, podemos analisar como um índice de desempenho global da empresa. Pelo autovetor, podemos ver que o patrimônio possui o maior peso e de menor o ganho líquido. Podemos verificar que quanto maior for os valores das variáveis, maior será dessa componente, ou melhor, maior será o desempenho global da empresa. Esta ocupa, observando pelos autovalores, 94,20% de toda variação explicada, dependendo da pesquisa pode-se descartar as outras componentes.

- A segunda componente que ocupa 5,77% de toda variação explicada (autovetor), possui o ganho bruto e patrimônio de maior variância amostral (analisando o tabela de covariância amostra). Pelos autovetores, podemos verificar que o ganho bruto é a variável dominante com segunda maior variância amostral. Com a componente próximo a zero, entende-se que haverá um certo equilíbrio entre ganho bruto e patrimônio acumulado, o que na verdade o aumento do ganho bruto eleva-se esta componente e o patrimônio contrário. Note que há correlação bem menor entre elas.
- A terceira componente com pouca variância total explicada, referente ao ganho líquido de menor variância amostral, possui pouca importância. Apenas o ganho líquido possui alta correlação, visto que às outras duas são próximas de zero.

2. Matriz de correlação

No exemplo anterior, vimos que as componentes principais foram obtidas a partir de matriz de covariâncias e que são influenciadas pelas variáveis com maior variância. Porém em casos onde existe muita discrepância entre essas variâncias por motivos de unidades de medidas distintas entre as variáveis. Podemos amenizar essa discrepância por meio de transformação dos dados originais de modo a equilibrar as variâncias ou colocar todos os dados em mesma escala de medida. Uma muito usual é a padronização que gera novas variáveis centradas em zero e com variâncias iguais a 1. Em caso de dúvida, veja em 6.2.2. Tomando como base o mesmo conjunto de dados do exercício anterior, padronizando e elaborando a matriz de correlação amostral, obtemos:

	Ganho Bruto (X_1)	Ganho Líquido (X_2)	Patrimônio Líquido
Ganho Bruto (X_1)	1,00	0,827	0,826
Ganho Líquido (X_2)	0,827	1,00	0,576
Patrimônio Líquido (X_3)	0,826	0,576	1,00

Com o mesmo procedimento do exemplo anterior ao caso de matriz de covariância, obtemos os respectivos autovalores e autovetores:

$$\lambda_1 = 2,493 \quad \lambda_2 = 0,423 \quad \lambda_3 = 0,084$$

$$\% \lambda_1 = 83,084\% \quad \% \lambda_2 = 14,117\% \quad \% \lambda_3 = 2,799\%$$

	Autovetor Ganho Bruto (u_1)	Autovetor Ganho Líquido (u_2)	Autovetor Patrimônio Líquido (u_3)
Autovetor Ganho Bruto (u_1)	0,617	-0,001	-0,787
Autovetor Ganho Líquido (u_2)	0,557	-0,706	0,437
Autovetor Patrimônio Líquido (u_3)	0,556	0,708	0,435

Por meio dos autovalores, podemos verificar que a variância total explicada pela primeira componente é aproximadamente 83,1%, pela segunda 14,1% e pela terceira 2,8%. As duas primeiras componentes explicam juntas 97,2% aproximadamente da variância total do vetor original padronizado. Note que o processo de análise é da mesma forma que o exemplo anterior. A primeira componente é um índice de desempenho global padronizado da empresa. A segunda componente representa uma comparação entre ganho líquido e patrimônio padronizados (verifique pelo autovetor da segunda componente que o ganho bruto possui um valor de coeficiente muito pequeno em relação aos outros). Por fim, a terceira componente compara-se o ganho bruto com às outras duas variáveis.

Suas componentes principais são:

$$\hat{y}_1 = 0,617(\text{GanhoBruto}) + 0,557(\text{GanhoLiquido}) + 0,556(\text{PatrimnioLiquido})$$

$$\hat{y}_2 = -0,001(\text{GanhoBruto}) - 0,706(\text{GanhoLiquido}) + 0,708(\text{PatrimnioLiquido})$$

$$\hat{y}_3 = -0,787(\text{GanhoBruto}) + 0,437(\text{GanhoLiquido}) + 0,435(\text{PatrimnioLiquido})$$

Note que em relação ao exemplo anterior, seus coeficientes de ponderação estão numericamente mais equilibrados que no caso de matriz de covariâncias amostral. Todas as variâncias iguais a um, sem dominância direta de nenhuma variável.

Determinada as componentes principais, podemos obter seus valores numéricos (escores) para cada elemento amostral. Podendo ser obtidas com técnicas estatísticas usuais como análise de variância e análise de regressão, entre outras. Usando dados nas três componentes principais, obtemos:

Empresas	CP_1	CP_2	CP_3
E1	1,85	0,65	-0,11
E2	1,22	1,07	-0,13

Empresas	CP_1	CP_2	CP_3
E3	3,84	-0,68	-0,13
E4	0,62	-0,96	0,41
E5	-0,23	1,20	0,31
E6	-1,22	-0,21	-0,60
E7	-1,08	-0,51	0,21
E8	-1,38	0,28	0,14
E9	-1,89	-0,13	-0,38
E10	-1,17	-0,02	0,36
E11	-0,56	-0,53	0,08
E12	0,00	0,15	-0,01

Da mesma forma que o exemplo anterior é importante reforçar novamente que pode haver troca de sinal de acordo com a formulação pelo software e a sua rotação em torno do eixo, visto que tratando de combinações lineares pode haver soluções com sinais diferentes. Pode resultar em valores numéricos diferentes e o pesquisador deve atentar em sua interpretação dos resultados obtidos. Pelos resultados nos leva a verificar que a empresa E9 possui o menor desempenho, e as E1, E2 e E3 os melhores.

Note que se compararmos os Escores do primeiro exemplo com matriz de covariância amostral e os Escores do exemplo com matriz de correlação amostral, foram concordantes a indicação das três empresas com melhor desempenho global e na de pior desempenho. Porém, algumas discordaram em algumas posições. Houve 5 concordância em 12 classificações (41,7%). Como as componentes principais foram obtidas pela decomposição espectral de matrizes diferentes, houve esta diferença. E que a matriz de correlação amostral leva em consideração a média do conjunto de 12 empresas de cada variável original, a matriz de covariância amostral não. Importante notar que neste caso, as primeiras componentes obtidas possuem a mesma interpretação, verificando consistência nas análises de ambos os métodos.

Com valores padronizados, fica mais fácil a visualização e interpretação da ACP por um gráfico, que denominamos de biplot. Basicamente colocamos no eixo X a primeira componente e no eixo Y a segunda componente principal (as que mais explicam a variância total). No qual as observações em análise são os Escores das observações (no caso as empresas) e os respectivos vetores das variáveis (Ganho Bruto, Ganho Líquido e Patrimônio Líquido).

A primeira componente principal (Dim1), as variáveis que se referem ao patrimônio, ganho bruto e ganho líquido possuem cargas positivas (todas tendendo fortemente à direita), reafirmando a análise anterior de que a primeira componente refere-se ao desempenho global padronizado da empresa. Com E1, E2 e E3 os maiores em desempenho. Para a segunda componente principal (Dim2), note que Patrimônio e Ganho Líquido estão em sentidos opostos, o que reafirma nossa análise anterior de ser uma comparação entre elas.

feitos/Livro ML/bookdown-demo-master/Figuras/biplotpadro.bb

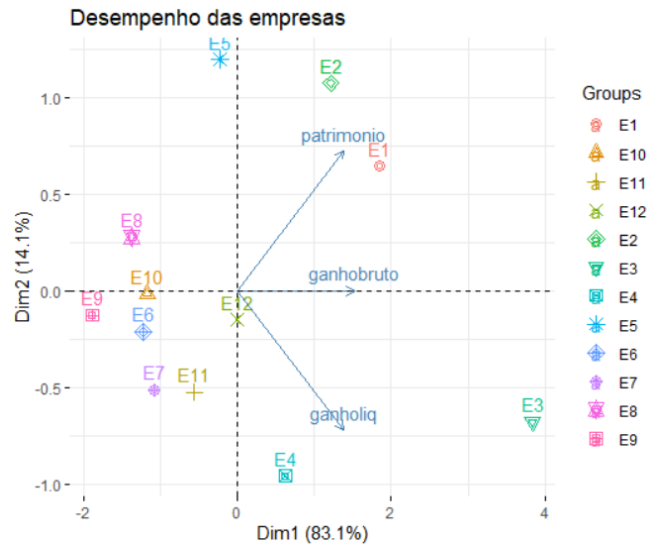


Figure 8.18: Gráfico de biplot, em X a primeira componente principal e Y a segunda componente.

8.6 Análise de Agrupamentos - Clusters

A Análise de Agrupamentos, também conhecido como Análise de Conglomerados, classificação ou *Clusters*, tem como propósito dividir os elementos de uma amostra (ou população) em grupos de modo que os elementos pertencentes a estes grupos tenham características similares entre si e heterôgenos com os outros grupos (Mingoti, 2007). Esta classificação vai de acordo com a medida e o método de classificação. Este tipo de análise é muito comum seu uso em diversas áreas como segmentação de clientes de acordo com perfis de consumo (Punj and Stewart, 1983), perfis de personalidade em psicologia (Speece et al., 1985), classificação de cidades, etc. Para o agrupamento de *clusters*, tomaremos como base a literatura de Mingoti (2007).

É muito importante o critério que o pesquisador utilizará para delimitar até que ponto os elementos podem ser considerados semelhantes em suas características ou não, por isso precisa-se de medidas apropriadas para classificar. Cada elemento amostral têm informações de p variáveis dentro de um vetor e por meio de medidas matemáticas, como as medidas de distância pode ser possível compararmos as observações dentro de seu banco de dados. Calculando a distância entre os vetores das observações da amostra e agrupando de acordo com suas distância (agrupar os de menores distâncias entre si). Aqui entramos com a aplicação de Medida de Distância, em 5.1.2. Qualquer medida de distância pode ser utilizada em variáveis quantitativas pode ser transformada num coeficiente

de similaridade (Mingoti, 2007).

A Análise de Clusters são frequentemente classificadas em **Hierárquicas (aglomerativas e divisivas)** que comumente utilizada para identificar possíveis agrupamentos e um provável valor da quantidade de grupos g e **Não hierárquicas** que necessita um número de grupos pré-estabelecido pelo pesquisados que a aplica.

feitos/Livro ML/bookdown-demo-master/Figuras/hierarq.bb

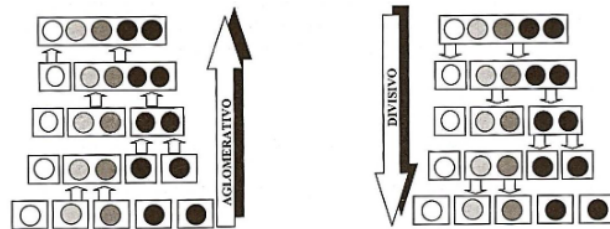


Figure 8.19: Esquema geral de procedimentos hierárquicos aglomerativos e divisivos (Mingoti, 2007).

8.6.1 Técnicas Hierárquicas Aglomerativas

Nesta técnica, inicia-se com n conglomerados como se cada elemento do banco e dados fosse um conglomerado isolado. No algoritmo, a cada passo os elementos amostrais vão sendo agrupados, formando novos conglomerados até que todos os elementos considerados estejam num único grupo. No início, tratando-se de variabilidade, tem-se a partição de menor dispersão interna já que todos os conglomerados possui apenas um único elemento (variância σ^2 zero). Ao final dos estágios, encontra-se a maior dispersão interna possível, pois todos os elementos amostrais estão num único *cluster* (Mingoti, 2007). Conforme Mingoti (2007), os passos fundamentais são:

1. Cada elemento possui um *cluster* de tamanho 1, logo n *clusters*;
2. Em cada estágio do algoritmo de agrupamento, os pares de conglomerados mais “similares” vão combinando-se passando a constituir um único conglomerado. Apenas um novo conglomerado pode ser formada a cada passo, ou seja, a cada etapa o número de conglomerados irá diminuir;
3. Como mostra-se na Figura 8.19, em cada estágio do algoritmo, cada novo conglomerado formado é um agrupamento de conglomerados de estágios anteriores. Se dois elementos amostrais aparecem juntos num mesmo *cluster* em alguma etapa, estarão juntos em todos os outros;
4. Por estarmos trabalhando com conglomerados em hierarquia, podemos construir um gráfico denominado Dendograma, ou Dendrograma, que representa a história do agrupamento. É um gráfico em forma de árvore tal

que a escala vertical indica o nível de similaridade (ou dissimilaridade) e na horizontal os elementos amostrais em ordem relacionada à história do agrupamento. Sua altura representa ao nível em que os elementos foram considerados semelhantes entre si (distância do agrupamento ou nível de similaridade).

Existe alguns métodos para que se escolha o número final dos grupos g , mas em geral é subjetivo com base em fundamentações empíricas. Vamos agora para os métodos mais comuns e utilizados em muitos *softwares* estatísticos.

8.6.1.1 Método de Ligação Simples (*Simple Linkage*)

A similaridade entre dois conglomerados é definida pelos dois elementos mais parecidos entre si (Sneath, 1957). Por exemplo, num determinado momento do algoritmo, encontra-se dois grupos: $C_1 = \{X_1, X_3, X_7\}$ e $C_2 = \{X_2, X_6\}$. A distância entre esses dois grupos será definida por:

$$d(C_1, C + 2) = \min\{d(X_l, X_k, l \neq k, l = 1, 3, 7 \text{ e } k = 2, 6)\} \quad (8.34)$$

é a distância entre os vizinhos mais próximos (elementos mais parecidos com os conglomerados. Em cada estágio, os dois conglomerados que são mais similares com relação à distância são combinados em um único *cluster*. Como ilustrado abaixo, a distância entre 6 e 1 caracteriza a distância entre os grupos, pelo método de ligação simples.

feitos/Livro ML/bookdown-demo-master/Figuras/ligsimples.bb

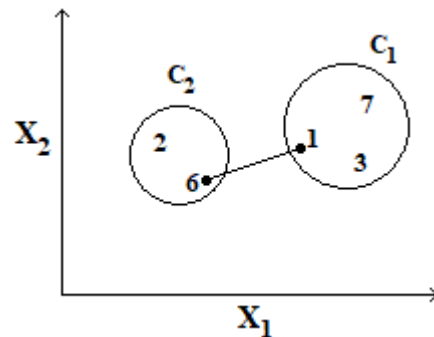


Figure 8.20: Método de ligação simples, adaptado de Mingoti (2007).

Vamos aproveitar o exemplo que utilizamos de distância Euclidiana, em 5.1.2.1. Com o seguinte banco de dados:

Table 8.10: Renda e Idade de 6 indivíduos, abordado em 5.1.2.1 sobre distância Euclidiana (Mingoti, 2007)

Renda	9,6	8,4	2,4	18,2	3,9	6,4
Idade	28	31	42	38	25	41

A matriz de distância calculada entre os seis elementos amostrais é dada por:

$$D_{6 \times 6} = \begin{bmatrix} & A & B & C & D & E & F \\ A & 0 & & & & & \\ B & 3,23 & 0 & & & & \\ C & 15,74 & 12,53 & 0 & & & \\ D & 13,19 & 12,04 & 16,29 & 0 & & \\ E & 6,44 & 7,50 & 17,06 & 19,33 & 0 & \\ F & 13,39 & 10,19 & 4,12 & 12,18 & 16,19 & 0 \end{bmatrix}$$

Sabemos que o menor valor observado na Matriz é 3,23 (distância entre os elementos A e B) nas duas variáveis medidas. Portanto este dois indivíduos são aglomerados fazendo com que a amostra de seis elementos passe a ser cinco. Lembrando que permanece as distâncias mínimas dos elementos com o novo conglomerado $\{A, B\}$ pois queremos os mais próximos.

$$d(\{A, B\}, \{C\}) = \min(d\{A, C\}, \{B, C\}) = \min(\{15, 74\}, \{12, 53\}) = 12, 53$$

$$d(\{A, B\}, \{D\}) = \min(d\{A, D\}, \{B, D\}) = \min(\{13, 19\}, \{12, 04\}) = 12, 04$$

$$d(\{A, B\}, \{E\}) = \min(d\{A, E\}, \{B, E\}) = \min(\{6, 44\}, \{7, 50\}) = 6, 44$$

$$d(\{A, B\}, \{F\}) = \min(d\{A, F\}, \{B, F\}) = \min(\{13, 39\}, \{10, 19\}) = 10, 19$$

$$D_{5 \times 5} = \begin{bmatrix} & \{A, B\} & C & D & E & F \\ \{A, B\} & 0 & & & & \\ C & 12,53 & 0 & & & \\ D & 12,04 & 16,29 & 0 & & \\ E & 6,44 & 17,06 & 19,33 & 0 & \\ F & 10,19 & 4,12 & 12,18 & 16,19 & 0 \end{bmatrix}$$

Nesta nova matriz, será a distância entre C e F (4,12), da mesma forma que anteriormente, fazendo com que fique quatro grupos:

$$D_{4 \times 4} = \begin{bmatrix} & \{A, B\} & \{C, F\} & \{D\} & \{E\} \\ \{A, B\} & 0 & & & \\ \{C, F\} & 10, 19 & 0 & & \\ \{D\} & 12, 04 & 12, 18 & 0 & \\ \{E\} & 6, 44 & 16, 19 & 19, 33 & 0 \end{bmatrix}$$

Agora a menor distância encontra-se entre $\{A, B\}$ e $\{E\}$ com 6,44:

$$D_{3 \times 3} = \begin{bmatrix} & \{A, B, E\} & \{C, F\} & \{D\} \\ \{A, B, E\} & 0 & & \\ \{C, F\} & 10, 19 & 0 & \\ \{D\} & 12, 04 & 12, 18 & 0 \end{bmatrix}$$

O próximo valor mínimo será 10,19 sobrando $C_1 = \{A, B, E, C, F\}$ e $C_2 = \{D\}$ e por fim reduz-se um único *cluster* $C_1 = \{A, B, C, D, E, F\}$ com o nível de junção igual a 12,04.

Portanto, o o histórico do agrupamento e seu respectivo dendograma será:

Table 8.11: Histórico do agrupamento por meio da Ligação Simples.

Passo	Número de Grupos	Fusão	Distância
1	5	$\{A\}$ e $\{B\}$	3,23
2	4	$\{C\}$ e $\{F\}$	4,12
3	3	$\{A, B\}$ e $\{E\}$	6,44
4	2	$\{A, B, E\}$ e $\{C, F\}$	10,19
5	1	$\{A, B, E, C, F\}$ e $\{D\}$	12,04

8.6.1.2 Método de Ligação Completa (*Complete Linkage*)

A similaridade entre dois conglomerados é definida pelos elementos que são menos semelhantes entre si (Sneath, 1957). Por exemplo, vamos considerar os conjuntos $C_1 = \{X_1, X_3, X_7\}$ e $C_2 = \{X_2, X_6\}$. A distância entre eles então será:

$$d(C_1, C_2) = \max\{d(X_l, X_k), l \neq k, l = 1, 3, 7 \text{ e } k = 2, 6\} \quad (8.35)$$

Em cada estágio calcula-se para os pares de grupos, combinando num único aqueles que estiverem com o menor valor da distância. Segue abaixo uma ilustração.

Aproveitando o mesmo exemplo que utilizamos no método anterior, sobre a distância Euclidiana, em 5.1.2.1. A matriz de distância calculada entre os seis

feitos/Livro ML/bookdown-demo-master/Figuras/dendsimples.bb

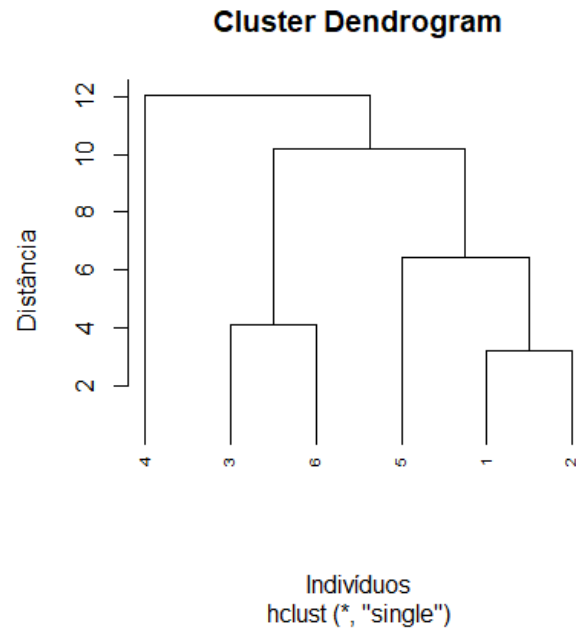


Figure 8.21: Dendrograma do agrupamento. Método de ligação simples.

feitos/Livro ML/bookdown-demo-master/Figuras/ligcompleta.bb

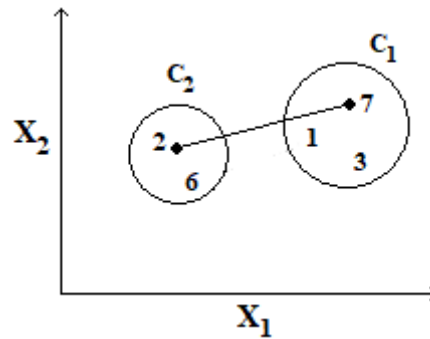


Figure 8.22: Método de ligação completa, adaptado de Mingoti (2007).

elementos amostrais é dada por:

$$D_{6 \times 6} = \begin{bmatrix} & A & B & C & D & E & F \\ A & 0 & & & & & \\ B & 3,23 & 0 & & & & \\ C & 15,74 & 12,53 & 0 & & & \\ D & 13,19 & 12,04 & 16,29 & 0 & & \\ E & 6,44 & 7,50 & 17,06 & 19,33 & 0 & \\ F & 13,39 & 10,19 & 4,12 & 12,18 & 16,19 & 0 \end{bmatrix}$$

Sabemos que o menor valor observado na Matriz é 3,23 (distância entre os elementos A e B) nas duas variáveis medidas. Portanto este dois indivíduos são aglomerados fazendo com que a amostra de seis elementos passe a ser cinco. Lembrando que permanece as distâncias máximas dos elementos com o novo conglomerado $\{A, B\}$ pois queremos os mais afastados.

$$d(\{A, B\}, \{C\}) = \max(d\{A, C\}, \{B, C\}) = \min(\{15, 74\}, \{12, 53\}) = 15, 74$$

$$d(\{A, B\}, \{D\}) = \max(d\{A, D\}, \{B, D\}) = \max(\{13, 19\}, \{12, 04\}) = 13, 19$$

$$d(\{A, B\}, \{E\}) = \max(d\{A, E\}, \{B, E\}) = \max(\{6, 44\}, \{7, 50\}) = 7, 50$$

$$d(\{A, B\}, \{F\}) = \max(d\{A, F\}, \{B, F\}) = \max(\{13, 39\}, \{10, 19\}) = 13, 39$$

$$D_{5 \times 5} = \begin{bmatrix} & \{A, B\} & C & D & E & F \\ \{A, B\} & 0 & & & & \\ C & 15,74 & 0 & & & \\ D & 13,19 & 16,29 & 0 & & \\ E & 7,50 & 17,06 & 19,33 & 0 & \\ F & 13,39 & 4,12 & 12,18 & 16,19 & 0 \end{bmatrix}$$

Nesta nova matriz, será a distância entre C e F (4,12), da mesma forma que anteriormente, fazendo com que fique quatro grupos e analisando pela máxima:

$$D_{4 \times 4} = \begin{bmatrix} & \{A, B\} & \{C, F\} & \{D\} & \{E\} \\ \{A, B\} & 0 & & & \\ \{C, F\} & 15,74 & 0 & & \\ \{D\} & 13,19 & 16,29 & 0 & \\ \{E\} & 7,50 & 17,06 & 19,33 & 0 \end{bmatrix}$$

Agora a menor distância encontra-se entre $\{A, B\}$ e $\{E\}$ com 7,50:

$$D_{3 \times 3} = \begin{bmatrix} & \{A, B, E\} & \{C, F\} & \{D\} \\ \{A, B\} & 0 & & \\ \{C, F\} & 17,06 & 0 & \\ \{D\} & 19,33 & 16,29 & 0 \end{bmatrix}$$

O próximo valor mínimo será 16,29, unindo os grupos $\{C, F\}$ e $\{D\}$, tornando os conglomerados $C_1 = \{A, B, E\}$ e $C_2 = \{C, F, D\}$ e por fim a distância máxima entre os dois grupos é 19,33.

Portanto, o o histórico do agrupamento por meio da Ligação Completa e seu respectivo dendograma será:

Table 8.12: Histórico do agrupamento por meio da Ligação Completa.

Passo	Número de Grupos	Fusão	Distância
1	5	$\{A\}$ e $\{B\}$	3,23
2	4	$\{C\}$ e $\{F\}$	4,12
3	3	$\{A, B\}$ e $\{E\}$	7,5
4	2	$\{C, F\}$ e $\{D\}$	16,29
5	1	$\{A, B, E\}$ e $\{C, F\}$	19,33

8.6.1.3 Método da Média das Distâncias (*Average Linkage*)

Neste caso a distância entre os conglomerados é com base nas médias. Se C_1 tem n_1 elementos e C_2 tem n_2 elementos, a distância será expressa como:

$$d(C_1, C_2) = \sum_{l \in C_1} \sum_{k \in C_2} \frac{1}{n_1 n_2} d(X_l, X_k) \quad (8.36)$$

Portanto, a distância entre $C_1 = \{X_1, X_3, X_7\}$ e $C_2 = \{X_2, X_6\}$ será:

$$d(C_1, C_2) = \frac{1}{6} [d(X_1, X_2) + d(X_1, X_6) + d(X_3, X_2) + d(X_3, X_6) + d(X_7, X_2) + d(X_7, X_6)]$$

Vamo considerar novamente a matriz inicial como exemplificação:

feitos/Livro ML/bookdown-demo-master/Figuras/dendcomplete.bb

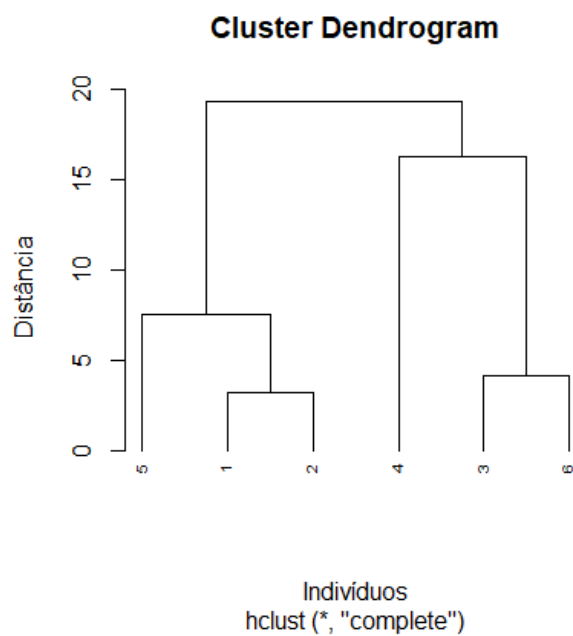


Figure 8.23: Dendrograma do agrupamento. Método de ligação completa.

$$D_{6 \times 6} = \begin{bmatrix} & A & B & C & D & E & F \\ A & 0 & & & & & \\ B & 3,23 & 0 & & & & \\ C & 15,74 & 12,53 & 0 & & & \\ D & 13,19 & 12,04 & 16,29 & 0 & & \\ E & 6,44 & 7,50 & 17,06 & 19,33 & 0 & \\ F & 13,39 & 10,19 & 4,12 & 12,18 & 16,19 & 0 \end{bmatrix}$$

Sabemos que o menor valor observado na Matriz é 3,23 (distância entre os elementos A e B) nas duas variáveis medidas. Portanto este dois indivíduos são aglomerados fazendo com que a amostra de seis elementos passe a ser cinco. Dessa vez são calculados em relação às médias das distância dos conglomerados $\{A, B\}$ com os outros.

$$d(\{A, B\}, \{C\}) = [d\{A, C\} + \{B, C\}]/2 = [\{15, 74\} + \{12, 53\}]/2 = 14, 13$$

$$d(\{A, B\}, \{D\}) = [d\{A, D\} + \{B, D\}]/2 = [\{13, 19\} + \{12, 04\}]/2 = 12, 62$$

$$d(\{A, B\}, \{E\}) = [d\{A, E\} + \{B, E\}]/2 = [\{6, 44\} + \{7, 50\}]/2 = 6, 97$$

$$d(\{A, B\}, \{F\}) = [d\{A, F\} + \{B, F\}]/2 = [\{13, 39\} + \{10, 19\}]/2 = 11, 79$$

$$D_{5 \times 5} = \begin{bmatrix} & \{A, B\} & C & D & E & F \\ \{A, B\} & 0 & & & & \\ C & 14, 13 & 0 & & & \\ D & 16, 62 & 16, 29 & 0 & & \\ E & 6, 97 & 17, 06 & 19, 33 & 0 & \\ F & 11, 79 & 4, 12 & 12, 18 & 16, 19 & 0 \end{bmatrix}$$

A próxima distância será entre C e F (4,12) novamente, fazendo com que fique quatro grupos. Repetindo o processo pela média.:

$$D_{4 \times 4} = \begin{bmatrix} & \{A, B\} & \{C, F\} & \{D\} & \{E\} \\ \{A, B\} & 0 & & & \\ \{C, F\} & 12, 96 & 0 & & \\ \{D\} & 12, 62 & 14, 24 & 0 & \\ \{E\} & 6, 97 & 16, 62 & 16, 19 & 0 \end{bmatrix}$$

Atente-se ao cálculo das distâncias médias, como por exemplo $\{A, B\}$ e $\{C, F\}$ foram quatros valores: $d(\{A, B\}, \{C, F\}) = [d(A, C) + d(A, F) + d(B, C) + d(B, F)]/4$.

Agora, teremos $\{A, B\}$ e $\{E\}$ com 6,97:

$$D_{3 \times 3} = \begin{bmatrix} & \{A, B, E\} & \{D\} & \{C, F\} \\ \{A, B\} & 0 & & \\ \{D\} & 14,85 & 0 & \\ \{E\} & 14,18 & 14,24 & 0 \end{bmatrix}$$

O próximo valor mínimo será 14,18, unindo os grupos $\{A, B, E\}$ e $\{C, F\}$, tornando os conglomerados $C_1 = \{A, B, C, E, F\}$ e $C_2 = \{D\}$ e por final será 14,61 a distância entre eles.

Portanto, o o histórico do agrupamento e seu respectivo dendograma será:

Table 8.13: Histórico do agrupamento por meio da Ligação Média.

Passo	Número de Grupos	Fusão	Distância
1	5	$\{A\}$ e $\{B\}$	3,23
2	4	$\{C\}$ e $\{F\}$	4,12
3	3	$\{A, B\}$ e $\{E\}$	6,97
4	2	$\{A, B, E\}$ e $\{C, F\}$	14,19
5	1	$\{A, B, E, C, F\}$ e $\{D\}$	14,61

8.6.1.4 Método do Centróide (*Centroid Method*)

A distância entre dois grupos é medida com a distância entre os vetores de médias, também denominado como centróides. Com $C_1 = \{X_1, X_3, X_7\}$ e $C_2 = \{X_2, X_6\}$, pode-se calcular:

$$\text{vetor de médias de } C_1 = \overline{X}_1 = \frac{1}{3}[X_1 + X_3 + X_7]$$

$$\text{vetor de médias de } C_2 = \overline{X}_2 = \frac{1}{2}[X_2 + X_6]$$

E a distância entre C_1 e C_2 , é a distância Euclidiana ao quadrado entre os vetores de médias amostral (também pode ser usado com a usual entre os vetores):

$$d(C_1, C_2) = (\overline{X}_1 - \overline{X}_2)'(\overline{X}_1 - \overline{X}_2) \quad (8.37)$$

Para cada passo do algoritmo, os conglomerados que possuem o menor valor de distância são agrupados. Vamos manter a matriz euclidiana (podemos manter a usual ou utilizar calcular as distâncias euclidianas ao quadrado)

feitos/Livro ML/bookdown-demo-master/Figuras/dendaverage.bb

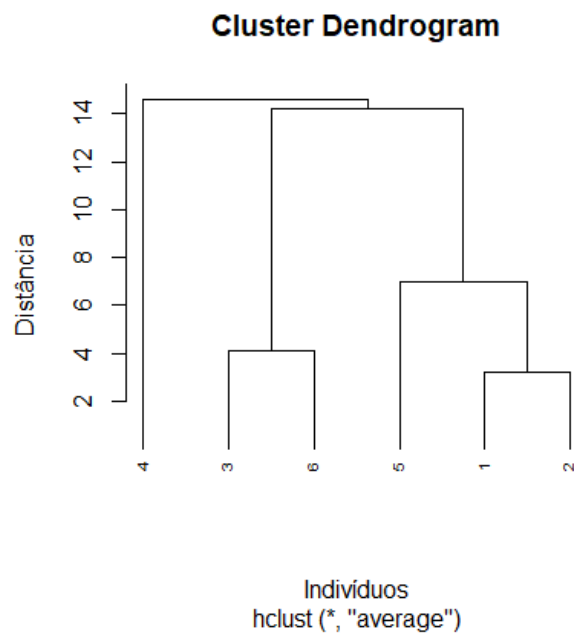


Figure 8.24: Dendrograma do agrupamento. Método da média das distâncias.

$$D_{6 \times 6} = \begin{bmatrix} & A & B & C & D & E & F \\ A & 0 & & & & & \\ B & 3,23 & 0 & & & & \\ C & 15,74 & 12,53 & 0 & & & \\ D & 13,19 & 12,04 & 16,29 & 0 & & \\ E & 6,44 & 7,50 & 17,06 & 19,33 & 0 & \\ F & 13,39 & 10,19 & 4,12 & 12,18 & 16,19 & 0 \end{bmatrix}$$

Lembrando que os valores originais das variáveis são:

Table 8.14: Renda e Idade de 6 indivíduos, abordado em 5.1.2.1 sobre distância Euclidiana (Mingoti, 2007).

Renda	9,6	8,4	2,4	18,2	3,9	6,4
Idade	28	31	42	38	25	41

Temos o menor o valor de 3,23 (distância entre os elementos A e B) nas duas variáveis medidas. Portanto este dois indivíduos são aglomerados fazendo com que a amostra de seis elementos passe a ser cinco. Lembrando que agora calcula-se a média entre os vetores, já que queremos o centróide. Em $\{A, B\}$ obtemos:

$$\{A, B\} = \left(\frac{9,6 + 8,4}{2}\right); \left(\frac{28 + 31}{2}\right) = (9; 29, 5)$$

Portanto calculando a distância com a nova coordenada $\{A, B\}$:

$$d(\{A, B\}, \{C\}) = \sqrt{(9 - 2,4)^2 + (29,5 - 42)^2} = 14,135$$

$$d(\{A, B\}, \{D\}) = \sqrt{(9 - 18,2)^2 + (29,5 - 38)^2} = 12,525$$

$$d(\{A, B\}, \{E\}) = \sqrt{(9 - 3,9)^2 + (29,5 - 25)^2} = 6,800$$

$$d(\{A, B\}, \{F\}) = \sqrt{(9 - 6,4)^2 + (29,5 - 41)^2} = 11,700$$

$$D_{5 \times 5} = \begin{bmatrix} & \{A, B\} & C & D & E & F \\ \{A, B\} & 0 & & & & \\ C & 14,13 & 0 & & & \\ D & 12,52 & 16,29 & 0 & & \\ E & 6,80 & 17,06 & 19,33 & 0 & \\ F & 11,70 & 4,12 & 12,18 & 16,19 & 0 \end{bmatrix}$$

A próxima distância será entre C e F (4,12) novamente, fazendo com que fique quatro grupos. Repetindo o processo sucessivamente de retornar aos dados

originais, recalculando e identificando o novo menor valor (o que dependendo do banco de dados exige tempo computacional) chegaremos aos seguintes resultados:

Table 8.15: Histórico do agrupamento pelo Método de Centróide.

Passo	Número de Grupos	Fusão	Distância
1	5	{A} e {B}	3,2
2	4	{C} e {F}	4,1
3	3	{A,B} e {E}	6,8
4	2	{A,B,E} e {C,F}	13,8
5	1	{A,B,E,C,F} e {D}	12,9

Lembrando que dependendo do arredondamento, pode haver pequena variação e que é possível fazer com o quadrado da distância euclidiana. Note também que o nível de fusão no passo 5 foi menor que o do passo 4. É possível esta ocorrência no método de centróide pois em algum passo do algoritmo de agrupamento houver empates entre valores da matriz de distâncias, quanto maior for o número de elementos amostrais, menor será a chance dessa ocorrência (Mingoti, 2007). A partição do dendograma será muito parecida com os anteriores, não será necessário apresentar.

8.6.1.5 Método de Ward (*Ward's Method*)

Vimos nos métodos anteriores que ao aumentarmos o estágio k para $k + 1$, a qualidade da partição decresce (com excessão de centróide) pois o nível de fusão e portanto o nível de similaridade também. Então percebe-se que a variação entre grupos diminui e a variação dentro dos grupos aumenta. Ward Jr (1963) propôs um método fundamental na mudança de variação entre os grupos em formação e entre cada passo do processo de agrupamento. É também conhecido como “mínima variância” por ter como objetivo a minimização da soma de quadrados dentro dos grupos.

Inicialmente cada elemento é considerado como um único elemento conglomerado e em cada passo do algoritmo de agrupamento é calculada a soma de quadrados dentro de cada conglomerado. Portanto, o agrupamento é feito a partir das somas de quadrados dos desvios entre acessos ou do quadrado da distância Euclidiana.

$$SS_i = \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_i)'(X_{ij} - \bar{X}_i) \quad (8.38)$$

em que n_i é o número de elementos no conglomerado C_i , quando se está no passo k ; X_i é o vetor de observações do j -ésimo elemento amostral e pertence ao i -ésimo conglomerado; \bar{X}_i , o centróide do conglomerado; e SS_i , a soma de

quadrados correspondente do conglomerado C_i . No passo k então, a soma de quadrados total é expressa como:

$$SS_i = \sum_{i=1}^{g_k} SS_i \quad (8.39)$$

onde g_k é o valor de grupos existentes quando se está no passo k . A Distância entre os conglomerados C_l e C_i é definida pela soma dos quadrados entre os *clusters* C_l e C_i :

$$d(C_l, C_i) = \left[\frac{n_l n_i}{n_l + n_i} \right] (X_l - \bar{X}_i)' (X_l - \bar{X}_i) \quad (8.40)$$

em que cada passo do algoritmo de agrupamento, os dois conglomerados que minimizam a distância são combinados.

Note que é muito semelhante com o método de centróide, porém o método de Ward leva em consideração a diferença dos tamanhos dos conglomerados que estão sendo comparados, ele possui como fator de ponderação $\left[\frac{n_l n_i}{n_l + n_i} \right]$ que quanto maior forem os valores de n_l e n_i e a discrepância entre eles, maior será o fator e portanto, a distância entre os centróides dos conglomerados comparados (Mingoti, 2007).

O processo de calcular é bem simples, semelhante ao método anterior, porém para as distâncias entre os conglomerados utiliza-se a equação (8.40) e acaba que sendo bem trabalhosa e consumindo muito tempo do pesquisador, por isso temos atualmente diversos *softwares* que possam auxiliar neste processo. Com o mesmo conjunto de dados de Renda e Idade do exemplos anteriores podemos chegar no seguinte histórico de agrupamento:

Table 8.16: Histórico do agrupamento pelo Método de Ward.

Passo	Número de Grupos	Fusão	Distância
1	5	{A} e {B}	10,44
2	4	{C} e {F}	17,00
3	3	{A,B} e {E}	61,68
4	2	{A,B,E} e {C,F}	270,25
5	1	{A,B,E,C,F} e {D}	465,00

Importante lembrar que de mesmo modo em outras literaturas, os métodos descritos fazem o agrupamento de elementos amostrais com base em algum critério pré-estabelecido, então nem sempre segue a divisão dos dados amostrais de ordem “natural” entre os n elementos amostrais ou populacional. Pode variar um pouco dependente do *Software* e sua simulação.

Atente-se pois com os exemplos utilizados, deram resultados bem próximos. Dependendo do tamanho do conjunto de dados pode nem sempre ocorrer assim, mas claro, esperamos uma consistência entre os diferentes métodos.

8.6.2 Número final de grupos

Como escolher o número final de grupos g ? Em qual passo k ? Isso varia muito com sua pesquisa, sua fundamentação teórica e até mesmo a experiência. Existe alguns critérios que pesquisadores utilizam para avaliar e tomar a decisão, as principais são:

1. **Análise do comportamento do nível de fusão:** sabemos que a medida que o passo aumenta, a similaridade entre os conglomerados vai decrescendo. Portanto muitos elaboram um gráfico de passo pelo nível de distância. Se há “pontos de salto” grandes em relações aos outros pontos de distância, pode indicar parar. Ao caso do exemplo Método de Ligação Simples ficaria:

feitos/Livro ML/bookdown-demo-master/Figuras/steps.bb

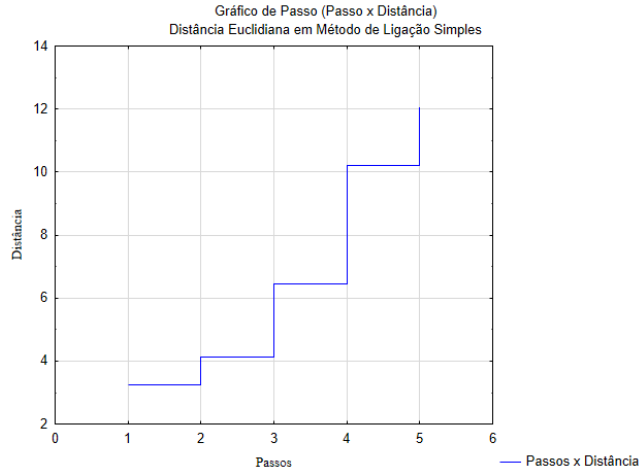


Figure 8.25: Gráfico de Passo a partir do exemplo de Método de Ligação Simples. Passos *versus* Distância.

2. **Análise do Comportamento do nível de similaridade:** em vez de observarmos o comportamento da distância em cada estágio, como no critério anterior. utiliza-se o seguinte cálculo para C_i e C_l unidos em certa etapa:

$$S_{il} = \left(1 - \frac{d_{il}}{\max\{d_{jk}, j, k = 1, 2, \dots, n\}}\right) \cdot 100 \quad (8.41)$$

sendo $d_{il}\{\max\{d_{jk}, j, k - 1, 2, \dots, n\}\}$ a maior distância entre os n elementos amostrais na matriz do primeiro estágio. Tem como objetivo encontrar pontos onde há decréscimo acentuado na similaridade dos conglomerados unidos (ao encontrar, finaliza o algoritmo). Segundo Felix (2004), geralmente valores acima de 90% resulta em quantidade de grupos muito elevado.

3. Análise da soma de quadrados entre grupos, o coeficiente R^2 :

Em cada k passo, podemos calcular a soma de quadrados entre os grupos e dentro dos grupos.

Para X_{ij} vetor de medidas observadas para o j -ésimo elemento amostral do i -ésimo grupo \bar{X} e partição dos dados amostrais em g grupos. A Soma de Quadrados Total corrigida para a média global em cada variável:

$$SST_c = \sum_{i=1}^g \sum_{j=1}^{n_i} (X_{ij} - \bar{X})'(X_{ij} - \bar{X}) \quad (8.42)$$

A Soma dos Quadrados Total dentro dos grupos da partição, que equivale ao residual:

$$SSR = \sum_{i=1}^g \sum_{j=1}^{n_i} = \sum_{i=1}^g SS_i \quad (8.43)$$

E a Soma de Quadrados Total entre os g grupos da partição:

$$SSB = \sum_{i=1}^g n_i (\bar{X}_i - \bar{X})'(\bar{X}_i - \bar{X}) \quad (8.44)$$

Por fim, o coeficiente R^2 é expresso por:

$$R^2 = \frac{SSB}{SST_c} \quad (8.45)$$

Quanto maior for seu valor, maior será a soma de quadrados SSB e menor o residual SSR. Elaborando um gráfico com os passos do agrupamento e o R^2 . E parar o algoritmo no “ponto de salto” grande em relação aos demais.

Observação: Há diversos critérios para o leitor se aprofundar, como **Estatística Pseudo F**, **Estatística Pseudo T**, **Correlação Semiparcial** que pode-se aplicar em método de Ward, **Estatística Cubic Clustering Criterium**, etc. Cabe o leitor interessado se aprofundar em seus estudos de acordo com sua pretensão. Os métodos hierárquicos são muito utilizados nas pesquisas atuais e ainda estão em constante desenvolvimento e combinações com outros modelos para aperfeiçoar suas pesquisas. Muitos usam o método hierárquico também,

da mesma forma que Análise de Componentes Principais e pré-processamento, para selecionar variáveis que possam ser utilizadas em sua pesquisa.

8.6.3 Técnicas Não Hierárquicas

As Técnicas Não Hierárquicas são técnicas que têm como propósito identificar diretamente uma partição de n elementos em k *clusters*, de modo que a partição satisfaça: coesão/semelhança interna e isolamento/separação dos *clusters* formados. Há muitas partições possíveis de ordem k e não é plausível criar todas possíveis (provavelmente nem será possível). Portanto deve-se utilizar alguns meios que possam investigar algumas partições viáveis e próxima da ótima.

Diferentemente do Hierárquico, esta técnica necessita que o pesquisador especifique previamente o número de *clusters* desejado. Em cada passo do agrupamento, os novos grupos podem ser formados através da divisão ou junção de grupos formados em etapas anteriores, ou seja, não necessariamente os mesmos elementos num mesmo conglomerado estarão juntos no final. Então, não será possível o uso de dendogramas e geralmente são processos iterativos com maior capacidade de dados. Vamos observar alguns métodos utilizados.

8.6.3.1 Método da K-Médias (*K-Means*)

Por Hartigan and Wong (1979), o método k-Médias é um dos mais conhecidos e utilizados em Análise de *Clusters*. Nele, cada elemento amostral é alocado ao *cluster* no qual o centróide (vetor de médias amostral) é o mais próximo de valores observados para o respectivo elemento. É composto pelos passos (Mingoti, 2007):

1. Selecione k centróides (conhecido como sementes ou protótipos), para iniciar a etapa de partição. Para selecionar varia também pelo método aplicado: pode-se selecionar de forma aleatória simples sem reposição; utilizar técnicas hierárquicas aglomerativas para se obter os grupos iniciais e calcular o vetor de médias; escolher a partir de uma variável aleatória de maior variância; selecionar por análise estatística elementos discrepantes no conjunto de dados; escolher prefixada ou os primeiros valores do *dataset*, etc;
2. Cada elemento do conjunto de dados é comparado com cada centróide inicial, por meio de alguma medida de distância (geralmente Euclidiana). O de menor distância é alocado ao grupo;
3. Após aplicar em cada n elemento amostral, recalcula-se os valores dos centróides para cada novo grupo formado e repete-se a etapa 2, considerando os centróides desse novo grupo;
4. Os passos 2 e 3 serão repetidos até que nenhuma realocação de elementos seja necessária e o pesquisador verificar e analisar de acordo com sua demanda.

Não é recomendável para o experimento quando k primeiros elementos amostrais são similares entre si.

Vamos a um exemplo:

Table 8.17: Valores dos índices de desenvolvimento de países.

Países	Expectativa de Vida	Educação	PIB	Estabilidade Política
Reino Unido	0,88	0,99	0,91	1,10
Austrália	0,90	0,99	0,93	1,26
Canadá	0,90	0,98	0,94	1,24
Estados Unidos	0,87	0,98	0,97	1,18
Japão	0,93	0,93	0,93	1,20
França	0,89	0,97	0,92	1,04
Cingapura	0,88	0,87	0,91	1,41
Argentina	0,81	0,92	0,80	0,55
Uruguai	0,82	0,92	0,75	1,05
Cuba	0,85	0,90	0,64	0,07
Colômbia	0,77	0,85	0,69	-1,36
Brasil	0,71	0,83	0,72	0,47
Paraguai	0,75	0,83	0,63	-0,87
Egito	0,70	0,62	0,60	0,21
Nigéria	0,44	0,58	0,37	-1,36
Senegal	0,47	0,37	0,45	-0,68
Serra Leoa	0,23	0,33	0,27	-1,26
Angola	0,34	0,36	0,51	-1,98
Etiópia	0,31	0,35	0,32	-0,55
Moçambique	0,24	0,37	0,36	0,20
China	0,76	0,80	0,61	0,39

Fonte: ONU, 2002, site: www.undp.org/hdro. Relatório de Desenvolvimento Humano.

Como dito, este método é muito dispendioso ao pesquisador calcular manualmente, portanto recomendo-o utilizar algum *software* estatístico para o seu cálculo e verificar o algoritmo do mesmo. Particularmente, utilizo para a escolha das sementes iniciais a seleção aleatória ou alguma técnica Hierárquica, pois ela evita com que há influências pessoais na seleção.

Após aplicarmos o método k -Médias com a escolha aleatória das sementes iniciais obtemos:

Table 8.18: Resultado descritivo dos *clusters* formados.

Grupos	SQ	Países	Média Expectativa de Vida	Média Educação	Média PIB	Média Estabilidade Política
$n_1 = 3$	0,0257	Reino Unido, França, Uruguai	0,8633	0,960	0,860	1,063
$n_2 = 7$	2,187	Colômbia, Paraguai, Nigéria, Senegal, Serra Leoa, Angola, Etiópia	0,473	0,524	0,463	-1,154
$n_3 = 6$	0,748	Argentina, Cuba, Brasil, Egito, Moçambique, China	0,678	0,740	0,622	0,315
$n_4 = 5$	0,047	Austrália, Canadá, Estados Unidos, Japão, Cingapura	0,896	0,950	0,936	1,258

Table 8.19: Análise da qualidade dos grupos formados.

SSR (Soma de Quadrados Residual, soma dos grupos)	3,008
SSB (Soma de Quadrados entre os g grupos)	22,757
SST (Soma de Quadrados Total)	25,765
$R^2 = SSB/SST$	88,3%

Podemos avaliar um bom modelo pelo seu R^2 , vale lembrar que o algoritmo foi aplicado para a escolha aleatória de centróides. O que pode variar o resultado de acordo com o processo. Por isso muitas vezes, os pesquisadores utilizam mais de um método para classificações para que se possa comparar e ter consistência em sua pesquisa. Caso o pesquisador verifique e valide estas classificações, pode-se aplicar novos estudos por exemplo, para cada conjunto de países, desde análise de regressão à diversos outros métodos.

8.6.3.2 Método de Fuzzy C-Médias (*C-Means*)

O método de Fuzzy (Bezdek, 1981) também é um método iterativo que requer do pesquisador pré-estabelecer do número de grupos, como *K-means*. Este método procura a partição que minimiza a função objetivo, expressa por:

$$J = \sum_{i=1}^c \sum_{j=1}^n (u_{ij}^m d(X_j, V_i)) \quad (8.46)$$

em que V_i é a semente (protótipo ou centróide ponderado) do conglomerado $i = 1, 2, \dots, c$, $m > 1$ é o parâmetro Fuzzy, quanto mais alto for, mais difuso será o cluster no final (geralmente usam-se $m = 2$); u_{ij} é a probabilidade de que o elemento X_j pertença ao conglomerado com a semente V_i e d é a distância (método escolhido pelo pesquisador, geralmente Euclidiana).

A função J (8.46) é minimizada quando as probabilidades u_{ij} e a semente V_i são definidas como:

$$u_{ij} = \left[\sum_{k=1}^c \left(\frac{d(X_j, V_i)}{d(X_j, V_k)} \right)^{2/(m-1)} \right]^{-1} \quad (8.47)$$

$$V_i = \frac{\sum_{j=1}^n (u_{ij})^m X_j}{\sum_{j=1}^n (u_{ij})^m} \quad (8.48)$$

em que $i = 1, 2, \dots, c$ e $j = 1, 2, \dots, n$. Com u_{ij} seguindo uma distribuição entre 0 e 1 e os protótipos vão se modificando a cada iteração. O algoritmo é interrompido quando a distância entre os protótipos de uma iteração em relação à anterior é menor ou igual a um erro μ estabelecido pelo pesquisador, ou seja, os vetores V_t e V_{t+1} da iterações t e $t+1$ que guardam as sementes precisam que: $d(V_t, V_{t+1}) < \mu$.

Para cada elemento amostral, este método estima uma probabilidade de que o este elemento pertença a cada um dos *clusters* c da partição. Podemos então encontrar elementos amostrais que se assemelham a mais de um dos c grupos. Alguns pesquisadores utilizam como critério de seleção para qual *cluster* irá pertencer de acordo com o que tenha a maior probabilidade.

Utilizando o mesmo conjunto de dados utilizado no método K-Médias e aplicarmos o método de Fuzzy C-Médias, supondo $m = 2$ e com *cluster* pré-estabelecido, obtemos seus resultados e alocando-os com base no critério de maior probabilidade.

Table 8.20: Resultado obtido pelo método de C-Médias. Utilizando como critério de alocar ao grupo pela maior probabilidade.

Países	Prob. C_1	Prob. C_2	Prob. C_3	Prob. C_4	Prob. C_5
Reino Unido	0,864	0,026	0,063	0,027	0,020
Austrália	0,776	0,044	0,098	0,046	0,035
Canadá	0,802	0,039	0,087	0,041	0,031
Estados Unidos	0,842	0,031	0,071	0,032	0,024
Japão	0,836	0,032	0,073	0,033	0,025
França	0,767	0,043	0,110	0,046	0,034
Cingapura	0,625	0,076	0,158	0,080	0,061
Argentina	0,228	0,098	0,500	0,103	0,071
Uruguai	0,636	0,066	0,177	0,070	0,051
Cuba	0,135	0,158	0,447	0,160	0,100
Colômbia	0,071	0,310	0,103	0,184	0,332
Brasil	0,123	0,068	0,685	0,075	0,049
Paraguai	0,048	0,557	0,080	0,162	0,152
Egito	0,120	0,121	0,533	0,144	0,082
Nigéria	0,024	0,103	0,035	0,081	0,757
Senegal	0,035	0,165	0,060	0,631	0,108
Serra Leoa	0,057	0,196	0,083	0,207	0,457
Angola	0,084	0,221	0,113	0,196	0,386
Etiópia	0,054	0,172	0,093	0,547	0,134
Moçambique	0,149	0,177	0,285	0,253	0,136
China	0,061	0,041	0,823	0,046	0,029

Portanto ficará:

Table 8.21: Quantidade de Países por grupo e Soma de Quadrados por grupo.

Grupos	Países	SQ
$n_1 = 8$	Reino Unido, Austrália, Canadá, Estados Unidos, Japão, França, Cingapura, Uruguai	0,157
$n_2 = 1$	Paraguai	0,000
$n_3 = 6$	Argentina, Cuba, Brasil, Egito, Moçambique, China	0,748

Grupos	Países	SQ
$n_4 = 2$	Senegal, Etiópia	0,030
$n_5 = 4$	Colômbia, Nigéria, Serra Leoa, Angola	0,763

Table 8.22: Análise da qualidade dos grupos formados.

SSR (Soma de Quadrados Residual, soma dos grupos)	1,698
SSB (Soma de Quadrados entre os g grupos)	20,983
SST (Soma de Quadrados Total)	22,681
$R^2 = SSB/SST$	0,925%

Tivemos um resultado interessante com um bom valor de R^2 e próximo ao do exemplo anterior, entretanto seria importante dar uma atenção maior em alguns países, visto que suas probabilidades para a seleção de *cluster* são bem semelhantes para Colômbia em $n_2(0,310)$ e $n_5(0,332)$ e Moçambique $n_3 = 0,285$ e $n_4 = 0,253$. Poderíamos testar com novas estratégias, novas quantidades de *clusters* ou alguns métodos de avaliação para que se avalie e torne mais consistente a análise. Em caso de variáveis com alta probabilidade não teremos dúvidas sobre sua alocação. O início da seleção de centróides foi formulado de forma aleatória para este exemplo. Recomendo-o o leitor retornar ao exemplo de K-médias e comparar a este ou até mesmo com Análise de Componentes Principais. Entenda que são metodologias diferentes com combinações de estratégias diferentes (desde medidas de distância como Euclidiana, método de análise multivariada, tipo de seleção de centróides, etc), podemos combinar e comparar todas estas técnicas para termos consistências em nossas pesquisas.

Em 9 serão apresentados outros métodos para medir o desempenho e validar seu modelo.

8.7 modelos nivel III

8.8 grad boosting -> estudar boosting e bagging dentro de emseamble

8.9 Redes Neurais

Chapter 9

Validação de um modelo

9.1 *Overfitting, Underfitting*

Sendo **muito importantes** nesta área, o Underfitting (sub-ajustado) e Overfitting (sobre-ajustado) são dois termos que temos que estar sempre atentos. Um bom modelo não deve sofrer de nenhum deles (Silver, 2013).

- **Overfitting:** Um cenário de overfitting ocorre quando, nos dados de treino, o seu modelo ML tem um desempenho excelente, porém quando utilizamos os dados em novos bancos de dados, seu resultado é ruim. Nesta situação, seu modelo aprendeu tão bem as relações existentes dos conjuntos de dados para treino que acabou apenas decorando esses dados. Portanto ao receber as informações das variáveis preditoras aos novos dados, o modelo tenta aplicar as mesmas regras decoradas, porém com estes novos dados (diferentes do treino) esta regra não tem validade e seu desempenho é afetado.

As principais causas e soluções de um Overfitting são:

1. Algoritmo muito complexo para os dados: caso for possível, pode-se simplificar o modelo utilizado por um algoritmo mais simples, com menos parâmetros. Permitindo reduzir as chances do modelo sofrer overfitting.
2. Poucos dados para treinar: dependendo da quantidade de dados utilizados para treinar, pode ser que seja uma amostra pequena, com isso recomenda-se aumentar seu tamanho coletando mais dados.
3. Ruídos nos dados de treinamento: é comum dentro do banco de dados existir algum tipo de ruído, isto é, *outlier* (valores extremos ou até mesmo valores incorretos nos dados). Esses ruídos podem fazer com que o modelo aprenda sobre ele, levando ao overfitting. Seria

recomendado pré-processamento adequado para tratar essa interferência.

9.1.1 Underfitting: No cenário underfitting, o desempenho já é ruim no próprio treinamento de seu algoritmo.

As principais causas e soluções de um Underfitting são:

1. Algoritmo inadequado: bem provável que o modelo estatístico proposto pelo pesquisa pode não ter sido adequado ao comportamento dos dados. Por exemplo aplicar um algoritmo para funções de primeiro grau (linear) em um conjunto de dados com comportamento exponencial (função de segundo grau). Recomendável o pesquisador substituir o algoritmo escolhendo outro com outros parâmetros para solucionar o underfitting.
2. Características não representativas: há possibilidade de que as características que estamos utilizando para treinar o modelo não sejam representativas, ou seja, não possuem relação entre si ou não sejam importantes para o modelo aplicado.
3. Modelo com muitos parâmetros de restrição: o modelo torna-se inflexível, restrito, e não consegue se ajustar de forma adequada aos dados.

Segue abaixo a Figura 9.1 demonstrando os dois casos anteriores e um modelo adequado.

feitos/Livro ML/bookdown-demo-master/Figuras/graficofit.bb

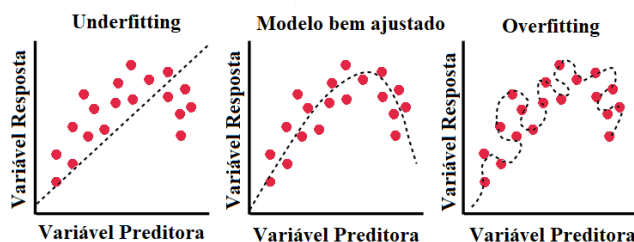


Figure 9.1: “Gráfico representando um *Underfitting*, um Modelo bem ajustado e um *Overfitting* respectivamente.”

9.2 Validação Cruzada

A fim de que não haja previsões desastrosas geradas pelo modelo, para medirmos o desempenho real do modelo criado, é necessário que realizemos testes com ele, utilizando dados diferentes dos que foram apresentados no início. Portanto uma das técnicas mais utilizadas é a **Cross-validation (Validação Cruzada)**.

Após a realização do pré-processamento (analisar), iremos separar a totalidade dos dados históricos existentes em dois grupos, sendo o primeiro responsável pelo aprendizado do modelo, e o segundo por realizar os testes. Seguindo o mesmo exemplo de bons ou mau pagadores, usualmente separamos o conjunto de dados dos clientes em duas amostras. Uma com

9.3 Como escolher um bom modelo?

9.4 AOC e ROC

Bibliography

- ANALYTICS VIDHYA (2016). Tree based algorithms: A complete tutorial from scratch (in r & python).
- AQUARELA (2017). Otimizando agendamentos médicos com inteligência artificial. *AQUARELA*.
- Assunção, F. (2012). *Estratégias para tratamento de variáveis com dados faltantes durante o desenvolvimento de modelos preditivos*. PhD thesis, Universidade de São Paulo.
- Banzatto, D. A. and Kronka, S. d. N. (1992). Experimentação agrícola. *Jaboticabal: Funep*, 2.
- Batista, G. E. d. A. P. et al. (2003). *Pré-processamento de dados em aprendizado de máquina supervisionado*. PhD thesis, Universidade de São Paulo.
- Bezdek, J. C. (1981). Objective function clustering. In *Pattern recognition with fuzzy objective function algorithms*, pages 43–93. Springer.
- Bobrow, D. G. (1967). Problems in natural language communication with computers. *IEEE Transactions on Human Factors in Electronics*, (1):52–55.
- Bolfarine, H. and Sandoval, M. C. (2001). *Introdução à inferência estatística*, volume 2. SBM.
- Box, G. E. and Jenkins, G. M. (1976). Time series analysis: Forecasting and control san francisco. *Calif: Holden-Day*.
- Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.
- Buchanan, B., Sutherland, G., and Feigenbaum, E. (1969). Heuristic dendral: A program for generating explanatory hypotheses. *Organic Chemistry*.
- Buchanan, B. G. and Shortliffe, E. H. (1984). Rule-based expert systems: the mycin experiments of the stanford heuristic programming project.
- CARVALHO, A., Faceli, K., LORENA, A., and Gama, J. (2011). Inteligência artificial—uma abordagem de aprendizado de máquina. *Rio de Janeiro: LTC*.

- Casella, G. and Berger, R. L. (2010). *Inferência estatística. São Paulo: Cengage Learning.*
- Cordeiro, G. M. (1999). *Introdução a teoria assintótica.* IMPA.
- Covões, T. F. (2010). *Seleção de atributos via agrupamento.* PhD thesis, Universidade de São Paulo.
- Cox, D. (1970). *Analysis of binary data* london: Methuen &co.
- Cross, S. E. and Walker, E. (1994). Dart: applying knowledge-based planning and scheduling to crisis action planning. *Intelligent Scheduling. Morgan Kaufmann.*
- da Silva Meloni, R. B. (2009). *Classificação de Imagens de Sensoriamento Remoto usando SVM.* PhD thesis, PUC-Rio.
- da Silveira, J. A. P. (2013). Searle e dennett: duas perspectivas de estudo da mente. *Problemata: Revista Internacional de Filosofia*, 4(2):238–258.
- de Andrade, D. F., Borgatto, A. F., Araujo, P. H., and Schmitt, J. (2019). *Caderno de Pesquisa 1: Técnicas de imputação de dados na análise de questionários contextuais.* Cebraspe, Brasília. ISBN 978-85-5656-010-0.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.
- Dennett, D. C. (2009). The part of cognitive science that is philosophy. *Topics in Cognitive Science*, 1(2):231–236.
- Ding, C. H. and Dubchak, I. (2001). Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics*, 17(4):349–358.
- Dougherty, J., Kohavi, R., and Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. In *Machine learning proceedings 1995*, pages 194–202. Elsevier.
- Enders, C. K. (2010). *Applied missing data analysis.* Guilford press.
- Evans, T. G. (1964). A program for the solution of a class of geometric-analogy intelligence-test questions. Technical report, AIR FORCE CAMBRIDGE RESEARCH LABS LG HANSCOM FIELD MASS.
- Fahlman, S. E. (1974). A planning system for robot construction tasks. *Artificial intelligence*, 5(1):1–49.
- Felix, F. N. (2004). Aplicando bootstrap para determinação de intervalos de confiança para o número de grupos no procedimento hierárquico aglomerativo de ward. *Dissertação-Mestrado.*
- Fisher, R. A. (1912). On an absolute criterion for fitting frequency curves. *Messenger of Mathematics*, 41:155–156.

- Freund, J. E. (2009). *Estatística Aplicada-: Economia, Administração e Contabilidade*. Bookman Editora.
- Garcia, S., Luengo, J., Sáez, J. A., Lopez, V., and Herrera, F. (2012). A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 25(4):734–750.
- Gonçalves, A. R. (2008). Máquina de vetores suporte. *Universidade Estadual de Londrina*, 21.
- Guimarães, R. R. C. (2019). A inteligência artificial e a disputa por diferentes caminhos em sua utilização preditiva no processo penal. *Revista Brasileira de Direito Processual Penal*, 5(3):1555–1588.
- Gujarati, D. N. and Porter, D. C. (2011). *Econometria básica-5*. Amgh Editora.
- Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, 28(1):100–108.
- Hartley, R. V. (1928). Transmission of information 1. *Bell System technical journal*, 7(3):535–563.
- Hebb, D. O. (1949). *The organization of behavior: a neuropsychological theory*. J. Wiley; Chapman & Hall.
- Henri, T. (1978). *Introduction to econometrics*. Prentice Hall, Englewood Cliffs, New Jersey.
- Holsheimer, M. and Siebes, A. P. (1994). *Data mining: The search for knowledge in databases*. Centrum voor Wiskunde en Informatica.
- Hongyu, K., Sandanielo, V. L. M., and de Oliveira Junior, G. J. (2016). Análise de componentes principais: resumo teórico, aplicação e interpretação. *E&S Engineering and Science*, 5(1):83–90.
- Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417.
- Huffman, D. A. (1971). Impossible object as nonsense sentences. *Machine intelligence*, 6:295–324.
- Ingargiola, G. (1996). Building classification models: Id3 and c4. 5. Disponível por WWW em: <http://www.cis.temple.edu/~ingargio/cis587/readings/id3-c45.html>.
- John, G. H., Kohavi, R., and Pfleger, K. (1994). Irrelevant features and the subset selection problem. In *Machine Learning Proceedings 1994*, pages 121–129. Elsevier.
- Kaiser, H. F. (1960). The application of electronic computers to factor analysis. *Educational and psychological measurement*, 20(1):141–151.

- Kennedy, P. E. (1981). The “ballentine”: a graphical aid for econometrics. *Australian Economic Papers*, 20(37):414–416.
- Langley, P. et al. (1994). Selection of relevant features in machine learning. In *Proceedings of the AAAI Fall symposium on relevance*, volume 184, pages 245–271. Citeseer.
- Lee, H. D. (2005). *Seleção de atributos importantes para a extração de conhecimento de bases de dados*. PhD thesis, Universidade de São Paulo.
- Light, R. J. and Margolin, B. H. (1971). An analysis of variance for categorical data. *Journal of the American Statistical Association*, 66(335):534–544.
- Lima, A. R. G. (2002). Máquinas de vetores suporte na classificação de impressões digitais. *Universidade Federal do Ceará, Departamento de Computação, Fortaleza-Ceará*.
- Little, R. J. and Rubin, D. B. (2019). *Statistical analysis with missing data*, volume 793. John Wiley & Sons.
- Liu, H. and Motoda, H. (1998). *Feature extraction, construction and selection: A data mining perspective*, volume 453. Springer Science & Business Media.
- Liu, H. and Motoda, H. (2008). Computational methods of feature selection (chapman & hall/crc data mining and knowledge discovery series).
- Liu, H. and Motoda, H. (2012). *Feature selection for knowledge discovery and data mining*, volume 454. Springer Science & Business Media.
- Lorena, A. C. and de Carvalho, A. C. (2003). Introdução as máquinas de vetores suporte. *Relatório Técnico do Instituto de Ciências Matemáticas e de Computação (USP/Sao Carlos)*, 192:11.
- Mahalanobis, P. C. (1936). On the generalized distance in statistics. National Institute of Science of India.
- Maroco, J. (2014). Análise estatística com o spss. *Statistics*, 6.
- McCarthy, J. (1968). Programs with common sense’in minsky m (ed) semantic information processing.
- McCarthy, J., Minsky, M. L., Rochester, N., and Shannon, C. E. (2006). A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI magazine*, 27(4):12–12.
- McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.
- Meehl, P. E. (1954). Clinical versus statistical prediction: A theoretical analysis and a review of the evidence minneapolis: University of minnesota press.[reprinted with new preface. In *In Proceedings of the 1955 Invitational Conference on Testing Problems*. Citeseer.

- Mingoti, S. A. (2007). Análise de dados através de métodos estatística multivariada: uma abordagem aplicada. In *Análise de dados através de métodos estatística multivariada: uma abordagem aplicada*, pages 295–295.
- Minsky, M. L. and Papert, S. (1969). Perceptrons: an introduction to. *Computational Geometry*.
- Montgomery, D. C., Peck, E. A., and Vining, G. G. (2012). *Introduction to linear regression analysis*, volume 821. John Wiley & Sons.
- MOSER, J. d. M. (2006). O golem. *Estudos em homenagem a Margarida Llosa*, pages 323–336.
- Moser, S. M. and Chen, P.-N. (2012). *A student's guide to coding and information theory*. Cambridge University Press.
- Newell, A. and Shaw, J. (1959). A variety of intelligent learning in a general problem solver. *RAND Report P-1742, dated July, 6*.
- NG, Andrew Y. (2019). Gradient descent algorithm.
- Nyquist, H. (1924). Certain factors affecting telegraph speed. *Transactions of the American Institute of Electrical Engineers*, 43:412–422.
- Orgânica Digital (2019). Algoritmo de classificação naive bayes.
- Parmezan, A. R. S., Lee, H. D., Spolaôr, N., and Chung, W. F. (2012). Avaliação de métodos para seleção de atributos importantes para aprendizado de máquina supervisionado no processo de mineração de dados.
- Paviotti, J. R. and Magossi, C. J. (2019). Considerações sobre o conceito de entropia na teoria da informação.
- Pearson, K. (1901). Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.
- Pereira, S. G. (2019). Inserção de dados faltantes não aleatórios para estimativa de variável geometalúrgica.
- Powell, Victor and Lehe, Lewis (2014). Análise do componente principal.
- Punj, G. and Stewart, D. W. (1983). Cluster analysis in marketing research: Review and suggestions for application. *Journal of marketing research*, 20(2):134–148.
- Rendle, S. and Schmidt-Thieme, L. (2008). Online-updating regularized kernel matrix factorization models for large-scale recommender systems. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 251–258.
- Rosenthal, R. (1994). Science and ethics in conducting, analyzing, and reporting psychological research. *Psychological Science*, 5(3):127–134.

- Rubin, D. B. (2004). *Multiple imputation for nonresponse in surveys*, volume 81. John Wiley & Sons.
- Russel, S. and NORVIG, P. (2004). Inteligência artificial. 2^a. edição. *Rio de Janeiro: Campus*.
- RUSSEL, S. and Norvig, P. (2013). Inteligência artificial. tradução de regina célia smile. *Rio de Janeiro: Campus Elsevier*.
- Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3):210–229.
- Schafer, J. L. (1999). Multiple imputation: a primer. *Statistical methods in medical research*, 8(1):3–15.
- Schafer, J. L. and Graham, J. W. (2002). Missing data: our view of the state of the art. *Psychological methods*, 7(2):147.
- Searle, J. R. (1980). Minds, brains, and programs, from the behavioral and brain sciences, vol. 3. *Cambridge University Press* [http://members. aol. com/NeoNoetics/MindsBrainsPrograms. html](http://members.aol.com/NeoNoetics/MindsBrainsPrograms.html) From, 23:2004.
- Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.
- Shannon, C. E. (1950). Xxii. programming a computer for playing chess. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 41(314):256–275.
- Shelley, M. (1818). Frankenstein or the modern prometheus. *London: Printed for Lackington, Hughes, Harding, Mayor & Jones*.
- Shiba, M. H., Santos, R. L., Quintanilha, J. A., and Kim, H. Y. (2005). Classificação de imagens de sensoriamento remoto pela aprendizagem por árvore de decisão: uma avaliação de desempenho. *Simpósio Brasileiro de Sensoriamento Remoto*, 12:4319–4326.
- Silver, N. (2013). *O sinal e o ruído*. Editora Intrínseca.
- Simon, P. (2013). *Too big to ignore: the business case for big data*, volume 72. John Wiley & Sons.
- Slagle, J. R. (1963). A heuristic program that solves symbolic integration problems in freshman calculus. *Journal of the ACM (JACM)*, 10(4):507–520.
- Smola, A. J., Bartlett, P., Schölkopf, B., and Schuurmans, D. (2000). Introduction to large margin classifiers.
- Sneath, P. H. (1957). The application of computers to taxonomy. *Microbiology*, 17(1):201–226.
- Souza, F. A. d. (2014). *Computational Intelligence Methodologies for Soft Sensors Development in Industrial Processes*. PhD thesis.

- Speece, D. L., McKinney, J. D., and Appelbaum, M. I. (1985). Classification and validation of behavioral subtypes of learning-disabled children. *Journal of Educational Psychology*, 77(1):67.
- Sung, A. H. and Mukkamala, S. (2003). Identifying important features for intrusion detection using support vector machines and neural networks. In *2003 Symposium on Applications and the Internet, 2003. Proceedings.*, pages 209–216. IEEE.
- TURING, I. B. A. (1950). Computing machinery and intelligence-am turing. *Mind*, 59(236):433.
- Vapnik, V. (2013). *The nature of statistical learning theory*. Springer science & business media.
- Waltz, D. (1975). Understanding line drawings of scenes with shadows. In *The psychology of computer vision*. Citeseer.
- Ward Jr, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301):236–244.
- Winograd, T. (1972). Understanding natural language. *Cognitive psychology*, 3(1):1–191.
- Winston, P. H. (1970). Learning structural descriptions from examples.