Estatística Multivariada 2

Victor Coscrato

2025-07-31

Índice

Prefácio

Seja Bem-vindo! Este livro aborda uma variedade de técnicas de análise multivariada, essenciais para a compreensão de dados complexos em diversas áreas do conhecimento.

Este material foi elaborado especialmente para estudantes tendo um primeiro contato com técnicas estatísticas multivariadas. São cobertos os métodos a seguir:

- Análise de Componentes Principais
- Análise Fatorial
- Análise de Agrupamento
- Análise Discriminante
- Análise de Correspondência
- Análise de Correlação Canônica

O objetivo é fornecer uma base sólida e prática para a aplicação dessas técnicas. antes, temos uma breve introdução dos conceitos fundamentais que norteiam a análise multivariada e algumas definições e resultados vetores e matrizes que são importantes para o acompanhamento do livro.

Parte I

Parte I: Introdução

1 Introdução

A análise multivariada é o campo da estatística dedicado a compreender conjuntos de dados com múltiplas variáveis inter-relacionadas. Em vez de analisar cada variável isoladamente, seu foco é examinar simultaneamente as relações entre três ou mais variáveis para extrair padrões e estruturas que de outra forma permaneceriam ocultos.

Cada observação em um estudo — seja um paciente descrito por indicadores de saúde, um consumidor por hábitos de compra, ou uma empresa por métricas financeiras — pode ser representada como um **vetor de observações**. A análise multivariada nos fornece as ferramentas para entender a estrutura de dependência e interdependência dentro desses vetores.

1.1 Por que usar Análise Multivariada?

A análise multivariada é motivada pela necessidade de extrair informações significativas de conjuntos de dados complexos. Ao invés de analisar variáveis de forma isolada, essas técnicas permitem uma compreensão mais profunda e realista dos dados. Os principais objetivos são:

- **Simplificação Estrutural:** Reduzir a dimensionalidade dos dados, identificando as principais fontes de variação e eliminando redundâncias. Isso facilita a visualização e a interpretação de dados complexos, revelando a estrutura subjacente de forma mais clara.
- **Agrupamento e Classificação:** Organizar as observações em grupos homogêneos (agrupamento) ou atribuir observações a categorias predefinidas (classificação). O objetivo é identificar padrões que permitam segmentar os dados de maneira significativa.
- **Investigação de Estruturas de Dependência:** Explorar e quantificar as relações entre variáveis. Isso inclui desde a análise de correlações simples até a modelagem de interações complexas entre múltiplos conjuntos de variáveis.
- **Predição:** Construir modelos para prever o valor de uma ou mais variáveis com base em outras.
- **Inferência:** Realizar testes de hipóteses e inferências estatísticas sobre as relações em um contexto multivariado.

Nos próximos capítulos, construiremos a base teórica para atingir esses objetivos, começando pelo conceito de vetor aleatório e seus parâmetros, para depois explorarmos como as amostras de dados nos permitem estimar e analisar essas estruturas.

1.2 Visão Geral das Técnicas Multivariadas

As técnicas de análise multivariada podem ser classificadas com base em seus objetivos e na natureza das relações entre as variáveis. Uma distinção fundamental é entre **técnicas de dependência**, que analisam a relação entre variáveis dependentes e independentes, e **técnicas de interdependência**, que exploram as relações em um único conjunto de variáveis.

- **Técnicas de Dependência:** Analisam a relação entre uma ou mais variáveis dependentes e um conjunto de variáveis independentes. O objetivo é prever ou explicar o valor das variáveis dependentes.
- **Técnicas de Interdependência:** Exploram as relações entre todas as variáveis de um conjunto, sem fazer distinção entre dependentes e independentes. O foco é entender a estrutura geral dos dados.

Além disso a escolha de uma determinada técnica depende também dos tipos de variáveis em questão.

- **Variáveis Categóricas (Qualitativas):** Representam categorias ou grupos (e.g., gênero, tipo de produto).
- **Variáveis Métricas (Quantitativas):** Representam quantidades numéricas (e.g., idade, altura, renda, temperatura).

Com o objetivo de classificar os métodos a serem apresentados nesse livro e posteriormente auxiliar na escolha da técnica mais adequada para o tratamento de um conjunto de dados, apresentamos a seguir uma tabela com algumas características de cada método e na sequência um fluxograma de decisão.

Tabela 1.1: Principais técnicas abordadas neste livro.

Técnica	Objetivo Principal	Tipo de Variável	Tipo de Análise
Componentes Principais (PCA)	Redução de dimensionalidade	Quantitativas	Interdependência
Análise Fatorial (FA)	Identificação de fatores latentes	Quantitativas	Interdependência
Análise de Agrupamento	Formação de grupos homogêneos	Quantitativas/Qualitativas	Interdependência
Análise Discriminante	Classificação de observações	Mista (Quali/Quanti)	Dependência
Correlação Canônica	Relação entre conjuntos de variáveis	Quantitativas	Dependência
Análise de Correspondência	Relação entre variáveis categóricas	Qualitativas	Interdependência

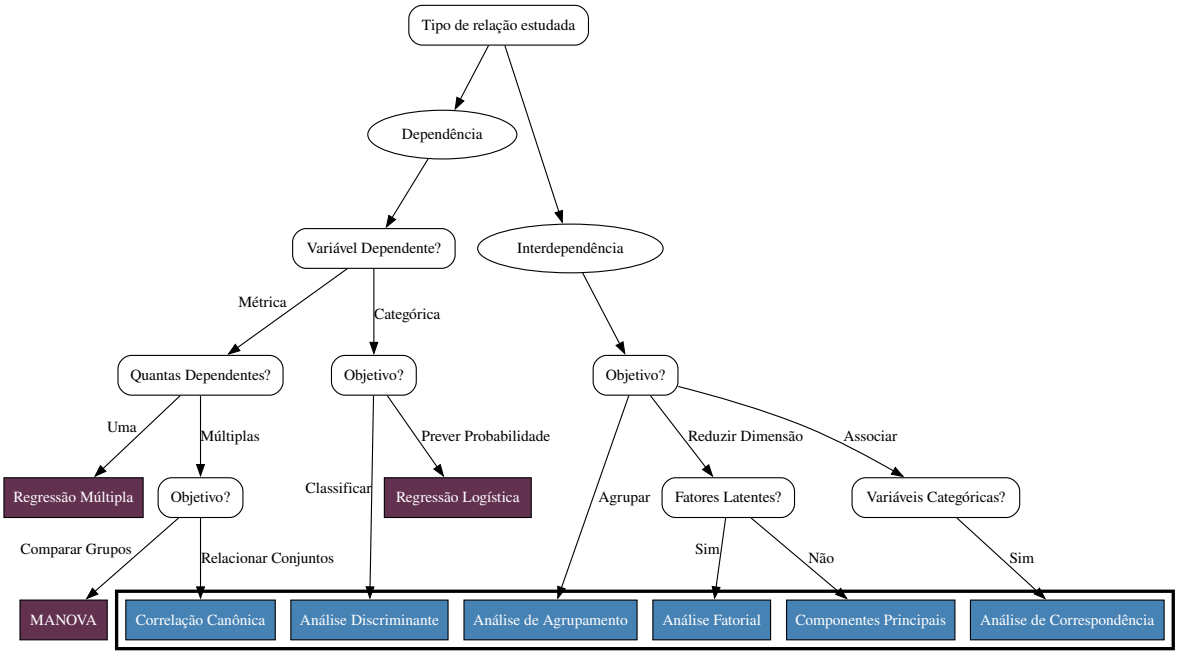

Figura 1.1: Diagrama de decisão para escolha de técnica de Análise Multivariada. Nós enfatizados fundo azul escuro indicam as técnicas de análise multivariada abordadas neste livro. Importante: Este diagrama é um guia simplificado para auxiliar na escolha da técnica mais adequada com base nas características dos dados e nos objetivos da análise. Ele não é exaustivo e serve apenas para posicionar as técnicas discutidas neste livro. A escolha final da técnica deve sempre considerar o contexto específico do problema e as características detalhadas dos dados

2 Vetores Aleatórios e Suas Características

No capítulo anterior, estabelecemos a motivação para a análise multivariada: a necessidade de entender sistemas complexos onde múltiplas variáveis interagem. Para fazer isso de maneira formal e rigorosa, precisamos primeiro definir o objeto matemático central de nosso estudo. Em vez de começar com uma tabela de dados, começamos com o conceito que gera esses dados: o **vetor aleatório**.

2.1 O Vetor Aleatório

Imagine que, para uma população de interesse (e.g., todos os estudantes de uma universidade), associamos a cada membro um conjunto de p características que nos interessam (e.g., nota em matemática, nota em história, horas de estudo). Antes de observarmos um membro específico, os valores dessas características são incertos. Podemos modelar essa incerteza tratando cada característica como uma variável aleatória.

Definição 2.1. Um **vetor aleatório** $\mathbf{x}$ é um vetor-coluna cujos componentes são p variáveis aleatórias, $X_1, X_2, \ldots, X_p$.

$$\mathbf{x} = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{pmatrix}$$

Este vetor é a representação matemática de uma "observação multivariada" em nível populacional. Toda a teoria da análise multivariada se baseia na compreensão das propriedades e da estrutura de distribuição deste vetor.

> Cuidado com a Notação
>
> - Uma letra minúscula em negrito (e.g., $\mathbf{x}$) denota um **vetor aleatório**.
> - Uma letra maiúscula comum (e.g., X_j) denota uma **variável aleatória** escalar, o j-ésimo componente do vetor.
> - Mais adiante, uma letra maiúscula em negrito (e.g., $\mathbf{X}$) será usada para a **matriz de dados** (amostral).
> - O apóstrofo ($'$) denota **transposição matricial**. Por exemplo, $\mathbf{A}'$ é a transposta de $\mathbf{A}$.

2.2 Parâmetros Populacionais

Assim como variáveis aleatórias escalares são caracterizadas por parâmetros como a média (expectativa) e a variância, os vetores aleatórios também o são. Esses parâmetros descrevem a tendência central, a dispersão e as inter-relações das variáveis que compõem o vetor.

Definição 2.2. O **vetor de médias populacional**, denotado por $\boldsymbol{\mu}$, é o vetor das expectativas de cada uma de suas variáveis componentes.

$$\boldsymbol{\mu} = E[\mathbf{x}] = \begin{pmatrix} E[X_1] \\ E[X_2] \\ \vdots \\ E[X_p] \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{pmatrix}$$

Geometricamente, $\boldsymbol{\mu}$ representa o **centróide** (centro de massa) da distribuição de probabilidade no espaço p-dimensional.

Definição 2.3. A **matriz de covariâncias populacional**, denotada por $\boldsymbol{\Sigma}$, é uma matriz simétrica $p \times p$ cujo elemento (j, k) é a covariância entre a j-ésima e a k-ésima variável aleatória, $\sigma_{jk} = \text{Cov}(X_j, X_k) = E[(X_j - \mu_j)(X_k - \mu_k)]$.

$$\boldsymbol{\Sigma} = \text{Cov}[\mathbf{x}] = E[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})'] = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{pmatrix}$$

- **Diagonal (σ_{jj}):** As **variâncias**, $\text{Var}(X_j)$, medem a dispersão de cada variável.
- **Fora da Diagonal (σ_{jk}):** As **covariâncias**, medem a tendência de associação linear entre as variáveis X_j e X_k.
- **Simetria:** A matriz é simétrica, pois $\text{Cov}(X_j, X_k) = \text{Cov}(X_k, X_j)$, o que implica $\sigma_{jk} = \sigma_{kj}$.

Definição 2.4. A **matriz de correlações populacional**, denotada por $\mathbf{P}$, é uma versão reescalada da matriz de covariâncias, com elementos $\rho_{jk} = \frac{\sigma_{jk}}{\sqrt{\sigma_{jj}}\sqrt{\sigma_{kk}}}$.

$$\mathbf{P} = \begin{pmatrix} 1 & \rho_{12} & \cdots & \rho_{1p} \\ \rho_{21} & 1 & \cdots & \rho_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{p1} & \rho_{p2} & \cdots & 1 \end{pmatrix}$$

Seus elementos ρ_{jk} variam de -1 a 1, fornecendo uma medida de associação linear livre de escala.

2.3 Propriedades de $\boldsymbol{\mu}$ e $\boldsymbol{\Sigma}$

As propriedades de combinações lineares são generalizações diretas dos resultados univariados. Seja $\mathbf{x}$ um vetor aleatório p-dimensional com média $\boldsymbol{\mu}$ e covariância $\boldsymbol{\Sigma}$. Sejam $\mathbf{c}$ um vetor de constantes $p \times 1$ e $\mathbf{A}$ uma matriz de constantes $q \times p$.

1. **Esperança de uma Combinação Linear:**

$$E[\mathbf{A}\mathbf{x} + \mathbf{c}] = \mathbf{A}E[\mathbf{x}] + \mathbf{c} = \mathbf{A}\boldsymbol{\mu} + \mathbf{c}$$

2. **Covariância de uma Combinação Linear:**

$$\text{Cov}[\mathbf{A}\mathbf{x} + \mathbf{c}] = \mathbf{A}\text{Cov}[\mathbf{x}]\mathbf{A}' = \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}'$$

No próximo capítulo, veremos como, na prática, não temos acesso a esses parâmetros populacionais ($\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, $\mathbf{P}$), mas podemos usar dados observados para obter estimativas confiáveis deles.

3 Amostra e Estimação de Parâmetros

No capítulo anterior, introduzimos os conceitos teóricos que descrevem uma população multivariada: o vetor de médias $\boldsymbol{\mu}$ e a matriz de covariâncias $\boldsymbol{\Sigma}$. Esses parâmetros são construções ideais que existem no nível populacional. Na prática, quase nunca temos acesso a toda a população para calculá-los diretamente.

O nosso trabalho como estatísticos e analistas de dados é fazer inferências sobre esses parâmetros desconhecidos com base em um conjunto limitado de dados. Fazemos isso através da **amostragem**.

3.1 Da População à Amostra

Assumimos que coletamos uma **amostra aleatória** de n observações da população. Cada observação, $\mathbf{x}_i$ (com $i = 1, \ldots, n$), é uma **realização** independente do vetor aleatório $\mathbf{x}$ que definimos no capítulo anterior.

A coleção de todas essas observações forma o nosso conjunto de dados. É aqui que, finalmente, introduzimos a **matriz de dados**, $\mathbf{X}$, uma estrutura central em toda a análise multivariada aplicada.

A matriz $\mathbf{X}$ é uma matriz de dimensão $n \times p$, onde cada linha é uma observação multivariada e cada coluna representa uma variável.

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1' \\ \mathbf{x}_2' \\ \vdots \\ \mathbf{x}_n' \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix} \tag{3.1}$$

O elemento x_{ij} representa o valor da j-ésima variável para a i-ésima observação. Com esta matriz em mãos, nosso objetivo é calcular quantidades que sirvam como boas **estimativas** para os parâmetros populacionais $\boldsymbol{\mu}$ e $\boldsymbol{\Sigma}$.

3.2 Estimadores Amostrais

As quantidades que calculamos a partir da amostra são chamadas de **estatísticas amostrais** ou **estimadores**, e são as contrapartes amostrais dos parâmetros populacionais.

Definição 3.1. O estimador de $\boldsymbol{\mu}$ é o **vetor de médias amostral**, $\bar{\mathbf{x}}$, cujos componentes $\bar{x}_j$ são a média das observações para a j-ésima variável.

$$\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}, \quad \text{resultando em} \quad \bar{\mathbf{x}} = \begin{pmatrix} \bar{x}_1 \\ \vdots \\ \bar{x}_p \end{pmatrix}$$

Definição 3.2. O estimador de $\boldsymbol{\Sigma}$ é a **matriz de covariâncias amostral**, $\mathbf{S}$. Seus elementos são a variância amostral (s_{jj}) e a covariância amostral (s_{jk}).

$s_{jk} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)$

A matriz resultante é:

$$\mathbf{S} = \begin{pmatrix} s_{11} & s_{12} & \cdots & s_{1p} \\ s_{21} & s_{22} & \cdots & s_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ s_{p1} & s_{p2} & \cdots & s_{pp} \end{pmatrix}$$

> **i** Por que dividir por $n-1$?
>
> A divisão por $n-1$ (graus de liberdade) em vez de n é feita para garantir que s_{jk} seja um estimador não-viesado de σ_{jk}, ou seja, $E[s_{jk}] = \sigma_{jk}$.

Definição 3.3. O estimador de $\mathbf{P}$ é a **matriz de correlações amostral**, $\mathbf{R}$, cujos elementos r_{jk} são obtidos padronizando a covariância amostral.

$r_{jk} = \frac{s_{jk}}{\sqrt{s_{jj}}\sqrt{s_{kk}}} = \frac{\sum_{i=1}^{n}(x_{ij}-\bar{x}_j)(x_{ik}-\bar{x}_k)}{\sqrt{\sum_{i=1}^{n}(x_{ij}-\bar{x}_j)^2}\sqrt{\sum_{i=1}^{n}(x_{ik}-\bar{x}_k)^2}}$

A matriz resultante é:

$$\mathbf{R} = \begin{pmatrix} 1 & r_{12} & \cdots & r_{1p} \\ r_{21} & 1 & \cdots & r_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ r_{p1} & r_{p2} & \cdots & 1 \end{pmatrix}$$

A matriz $\mathbf{R}$ é uma matriz simétrica com 1s na diagonal.

Em resumo: Neste capítulo, fizemos a ponte entre a teoria e a prática. - No **nível populacional**, temos parâmetros teóricos e não observáveis ($\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$). - No **nível amostral**, temos dados observáveis na matriz $\mathbf{X}$, a partir da qual calculamos estatísticas ($\bar{\mathbf{x}}$, $\mathbf{S}$) que **estimam** esses parâmetros.

A maior parte das técnicas que veremos neste livro opera sobre as matrizes $\mathbf{S}$ ou $\mathbf{R}$ para fazer inferências sobre a estrutura da população.

4 A Distribuição Normal Multivariada

Até agora, discutimos os parâmetros de um vetor aleatório ($\boldsymbol{\mu}$ e $\boldsymbol{\Sigma}$) sem assumir uma forma específica para sua distribuição de probabilidade. No entanto, para desenvolvermos uma teoria de inferência estatística robusta e compreendermos o funcionamento de muitas técnicas clássicas, precisamos de um modelo de distribuição de referência. Na análise multivariada, esse papel é desempenhado pela **distribuição Normal Multivariada (NMV)**.

A NMV é uma generalização da distribuição normal (Gaussiana) para o caso de p variáveis. Ela é, de longe, a distribuição mais importante da análise multivariada, por várias razões: 1. Muitos fenômenos naturais podem ser aproximados pela NMV. 2. O Teorema do Limite Central, em sua forma multivariada, garante que a média de vetores aleatórios de (quase) qualquer distribuição tende a se comportar como uma NMV para amostras grandes. 3. Suas propriedades matemáticas são extremamente convenientes e bem compreendidas, o que facilita muito a derivação de resultados teóricos.

4.1 A Função de Densidade

Um vetor aleatório $\mathbf{x}$ de dimensão p segue uma distribuição Normal Multivariada com vetor de médias $\boldsymbol{\mu}$ e matriz de covariâncias $\boldsymbol{\Sigma}$ (positiva definida), denotado por $\mathbf{x} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, se sua função de densidade de probabilidade (FDP) for dada por:

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

Onde: - $|\boldsymbol{\Sigma}|$ é o determinante da matriz de covariâncias. - $\boldsymbol{\Sigma}^{-1}$ é a inversa da matriz de covariâncias.

Apesar de parecer intimidante, a estrutura da FDP é bastante lógica. O termo no expoente, $(\mathbf{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})$, é uma forma quadrática que mede a "distância" do ponto $\mathbf{x}$ ao centro $\boldsymbol{\mu}$, ponderada pela estrutura de covariância $\boldsymbol{\Sigma}$. Essa distância é chamada de **distância de Mahalanobis**. Quanto maior essa distância, menor o valor da função de densidade, o que faz todo o sentido.

4.2 Propriedades da Distribuição Normal Multivariada

A popularidade da NMV vem de suas propriedades elegantes:

1. **Combinações Lineares:** Se $\mathbf{x} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, então qualquer combinação linear de suas componentes, $\mathbf{a}'\mathbf{x} = a_1X_1 + \cdots + a_pX_p$, segue uma distribuição normal univariada. Mais geralmente, se $\mathbf{A}$ é uma matriz de constantes, então $\mathbf{Ax}$ segue uma distribuição Normal Multivariada.
2. **Distribuições Marginais:** Qualquer subconjunto de variáveis de um vetor Normal Multivariado também segue uma distribuição Normal Multivariada. Por exemplo, se $\mathbf{x} = [X_1, X_2, X_3]'$ é NMV, então o vetor $[X_1, X_3]'$ também é NMV.
3. **Independência e Covariância Zero:** Para a maioria das distribuições, covariância zero não implica independência. No entanto, para a NMV, essa implicação é verdadeira. Se um subconjunto de variáveis em um vetor NMV tem covariância zero com outro subconjunto, então esses dois subconjuntos são **independentes**. Esta é uma propriedade extremamente poderosa.

4.3 Visualizando a Normal Bivariada

Para ganhar intuição, é útil visualizar o caso bivariado ($p = 2$). A função de densidade forma uma superfície em forma de sino no espaço 3D. Os contornos de densidade constante, quando projetados no plano (x_1, x_2), formam elipses.

$$(\mathbf{x} - \boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}) = c^2$$

A forma e a orientação dessas elipses são inteiramente determinadas pela matriz de covariâncias $\boldsymbol{\Sigma}$.

- **Se** $\sigma_{12} = 0$ **e** $\sigma_{11} = \sigma_{22}$**:** As variáveis são não correlacionadas (e, portanto, independentes) e têm a mesma variância. Os contornos são **círculos**.
- **Se** $\sigma_{12} = 0$ **e** $\sigma_{11} \neq \sigma_{22}$**:** As variáveis são não correlacionadas, mas com variâncias diferentes. Os contornos são **elipses alinhadas com os eixos** de coordenadas.
- **Se** $\sigma_{12} \neq 0$**:** As variáveis são correlacionadas. Os contornos são **elipses rotacionadas**. A direção do eixo principal da elipse é determinada pelos autovetores de $\boldsymbol{\Sigma}$, e o comprimento dos eixos é determinado pelos autovalores.

Essa conexão entre a álgebra da matriz $\boldsymbol{\Sigma}$ e a geometria da distribuição de dados é um dos temas mais importantes da análise multivariada e será a base para a técnica de Componentes Principais, que exploraremos mais adiante.

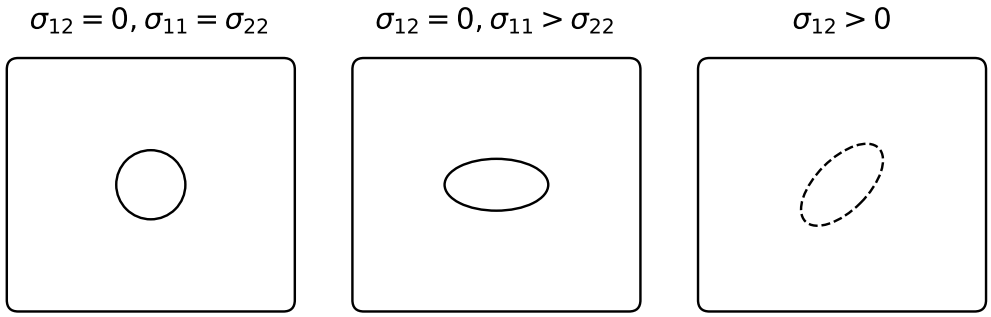

Figura 4.1: Contornos de densidade para uma distribuição Normal Bivariada, ilustrando o efeito da matriz de covariância.

5 Fundamentos Matemáticos: Álgebra Matricial

Com os conceitos estatísticos fundamentais estabelecidos, voltamos nossa atenção para as ferramentas matemáticas necessárias para manipular esses objetos. A linguagem da análise multivariada é a álgebra linear.

Neste capítulo, revisaremos conceitos-chave — formas quadráticas, matrizes positiva-definidas e a decomposição espectral — que são a base para muitas das técnicas que veremos, como a Análise de Componentes Principais (PCA).

5.1 Formas Quadráticas

Uma forma quadrática é uma função polinomial de várias variáveis que contém apenas termos de grau dois. Para um vetor $\mathbf{x}$ de dimensão $p \times 1$ e uma matriz simétrica $\mathbf{A}$ de dimensão $p \times p$, a forma quadrática é expressa como:

$$Q(\mathbf{x}) = \mathbf{x}'\mathbf{A}\mathbf{x} = \sum_{i=1}^{p}\sum_{j=1}^{p} a_{ij}x_i x_j$$

Um exemplo fundamental que já encontramos é a distância de Mahalanobis ao quadrado, $(\mathbf{x} - \boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})$, que aparece no expoente da distribuição normal multivariada. Esta forma quadrática define as elipses de contorno de densidade constante da distribuição.

5.2 Matrizes Positiva-Definidas

O conceito de positividade para um número escalar é estendido para matrizes através das formas quadráticas. Uma matriz simétrica $\mathbf{A}$ é dita:

- **positiva-definida** se $\mathbf{x}'\mathbf{A}\mathbf{x} > 0$ para todos os vetores não-nulos $\mathbf{x}$.
- **positiva-semidefinida** se $\mathbf{x}'\mathbf{A}\mathbf{x} \geq 0$ para todos os vetores não-nulos $\mathbf{x}$.

Propriedades de uma matriz positiva-definida: - Todos os seus autovalores são estritamente positivos ($\lambda_i > 0$). - A matriz é invertível (não-singular). - Seu determinante é positivo.

Matrizes de covariância ($\mathbf{\Sigma}$) e correlação ($\mathbf{R}$) são, por natureza, positiva-semidefinidas. Para que a função de densidade da normal multivariada seja bem definida e a matriz $\mathbf{\Sigma}$ seja invertível, exigimos que ela seja **positiva-definida**. Isso implica que nenhuma variável no vetor aleatório é uma combinação linear perfeita de outras (ou seja, não há redundância linear total nos dados).

5.3 Decomposição Espectral

A decomposição espectral (ou de autovalores) é uma fatoração de uma matriz simétrica em seus autovalores e autovetores. Ela revela a estrutura fundamental da transformação linear representada pela matriz.

Toda matriz simétrica $\mathbf{A}$ de dimensão $p \times p$ pode ser reescrita como:

$$\mathbf{A} = \mathbf{E}\Lambda\mathbf{E}'$$

Onde:

- $\lambda_1, \ldots, \lambda_p$ são os **autovalores** de $\mathbf{A}$.
- $\mathbf{e}_1, \ldots, \mathbf{e}_p$ são os **autovetores** ortonormais correspondentes.
- Λ é a matriz diagonal com os autovalores λ_i na diagonal.
- $\mathbf{E}$ é a matriz ortogonal cujas colunas são os autovetores $\mathbf{e}_i$.

Exemplo 5.1. Vamos decompor a seguinte matriz de covariâncias $\mathbf{S}$:

$$\mathbf{S} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

1. **Autovalores**: Resolvendo a equação característica $\det(\mathbf{S} - \lambda\mathbf{I}) = 0$, encontramos $\lambda_1 = 3$ e $\lambda_2 = 1$.
2. **Autovetores**:
   - Para $\lambda_1 = 3$: O autovetor correspondente é $\mathbf{e}_1 = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}$.
   - Para $\lambda_2 = 1$: O autovetor correspondente é $\mathbf{e}_2 = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix}$.

A decomposição é $\mathbf{S} = \mathbf{E}\Lambda\mathbf{E}'$, com:

$$\mathbf{E} = \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix}, \quad \Lambda = \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}$$

Isso nos diz que a maior variância dos dados (igual a 3) está na direção do vetor $(1, 1)$, enquanto a variância na direção ortogonal $(1, -1)$ é menor (igual a 1).

5.4 Decomposição em Valores Singulares (SVD)

Enquanto a decomposição espectral é uma ferramenta poderosa para matrizes simétricas, a **Decomposição em Valores Singulares (SVD)** a generaliza para **qualquer** matriz $\mathbf{A}$ de dimensão $I \times J$. A SVD é uma das fatorações de matrizes mais importantes da álgebra linear, com aplicações vastas em estatística e aprendizado de máquina, incluindo a Análise de Componentes Principais e a Análise de Correspondência.

A SVD decompõe qualquer matriz $\mathbf{A}$ na forma:

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}'$$

Onde:

- $\mathbf{U}$ é uma matriz ortogonal $I \times I$ cujas colunas são chamadas de **vetores singulares à esquerda**.
- $\mathbf{V}$ é uma matriz ortogonal $J \times J$ cujas colunas são chamadas de **vetores singulares à direita**.
- $\mathbf{\Lambda}$ é uma matriz retangular $I \times J$ contendo os **valores singulares** σ_k em sua diagonal principal, em ordem decrescente ($\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$). Todos os outros elementos de $\mathbf{\Lambda}$ são zero.

Os valores singulares são as raízes quadradas dos autovalores não-nulos das matrizes $\mathbf{A}'\mathbf{A}$ e $\mathbf{A}\mathbf{A}'$.

5.4.1 Relação com a Decomposição Espectral

Para uma matriz simétrica e positiva-semidefinida $\mathbf{A}$ (como uma matriz de covariância), a SVD e a decomposição espectral são essencialmente a mesma coisa. Seus valores singulares são seus autovalores, e seus vetores singulares à esquerda e à direita são seus autovetores ($\mathbf{U} = \mathbf{V} = \mathbf{E}$).

5.4.2 Importância para Redução de Dimensionalidade

A grande utilidade da SVD vem do fato de que ela fornece a **melhor aproximação de baixo posto** de uma matriz. O Teorema de Eckart-Young afirma que, se truncarmos a decomposição para usar apenas os M maiores valores singulares, a matriz resultante $\mathbf{A}_M$ é a melhor aproximação de posto M da matriz original $\mathbf{A}$.

$$\mathbf{A} \approx \mathbf{A}_M = \mathbf{U}_M \mathbf{\Lambda}_M \mathbf{V}'_M = \sum_{k=1}^{M} \sigma_k \mathbf{u}_k \mathbf{v}'_k$$

Isso significa que podemos capturar a estrutura mais importante de uma matriz usando um número menor de dimensões.

6 Medidas de Distância e Similaridade

Um conceito fundamental que permeia quase todas as técnicas de análise multivariada é a medição da "proximidade" ou "distância" entre observações. Seja para agrupar dados semelhantes, classificar uma nova observação ou entender a estrutura de um conjunto de dados, essas medidas determinam uma forma quantitativa para expressar o quão perto ou longe duas observações estão uma da outra no espaço p-dimensional.

6.1 Distâncias vs. Dissimilaridades

Formalmente, uma função $d(\cdot, \cdot)$ é considerada uma **métrica de distância** se satisfaz as seguintes propriedades para quaisquer pontos $\mathbf{x}, \mathbf{y}, \mathbf{z}$:

1. **Não-negatividade:** $d(\mathbf{x}, \mathbf{y}) \geq 0$
2. **Identidade:** $d(\mathbf{x}, \mathbf{y}) = 0 \iff \mathbf{x} = \mathbf{y}$
3. **Simetria:** $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$
4. **Desigualdade Triangular:** $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})$

No entanto, em muitos contextos práticos, utilizamos medidas que não satisfazem todas essas propriedades, mas que ainda são extremamente úteis para quantificar o quão diferentes dois objetos são. Usamos o termo mais geral **medida de dissimilaridade** para nos referirmos a qualquer função que indique o grau de diferença entre dois pontos, onde valores pequenos indicam semelhança e valores grandes indicam diferença.

Um exemplo clássico de uma medida de dissimilaridade que não é uma métrica de distância estrita é a **distância Euclidiana quadrática**, $d^2(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})'(\mathbf{x} - \mathbf{y})$. Ela viola a propriedade da desigualdade triangular, mas pode ser usada em algoritmos como o K-médias e o método de Ward por suas convenientes propriedades computacionais (evitar o cálculo da raiz quadrada economiza tempo).

Nas seções a seguir, apresentamos algumas das medidas de dissimilaridade e distância mais populares. A escolha da medida ideal é um campo vasto e depende fundamentalmente da natureza dos dados e do objetivo da análise.

6.2 Medidas para Dados Contínuos

Definição 6.1. A **Distância Euclidiana** é a métrica de distância mais comum e corresponde à noção intuitiva de distância em linha reta entre dois pontos.

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2} = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j)}$$

Definição 6.2. A **Distância de Manhattan** (ou *City-Block*) calcula a distância como a soma das diferenças absolutas entre as coordenadas dos pontos. É como se deslocar entre dois pontos em uma cidade, movendo-se apenas ao longo das ruas (horizontais e verticais).

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{p} |x_{ik} - x_{jk}|$$

Esta medida é, em geral, mais robusta a *outliers* do que a distância Euclidiana.

Definição 6.3. A **Distância de Minkowski** é uma generalização tanto da Euclidiana quanto da de Manhattan.

$$d(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{k=1}^{p} |x_{ik} - x_{jk}|^m \right)^{1/m}$$

- Se $m = 2$, temos a distância Euclidiana.
- Se $m = 1$, temos a distância de Manhattan.

Quanto maior o valor de m, mais peso é dado às maiores diferenças entre as coordenadas.

> ⚠ Limitação das Distâncias Comuns
>
> As distâncias Euclidiana, de Manhattan e de Minkowski são sensíveis às escalas das variáveis. Se uma variável tiver uma magnitude muito maior que as outras, ela dominará o cálculo da distância. Por isso, é prática comum **padronizar** as variáveis (subtrair a média e dividir pelo desvio padrão) antes de calcular a matriz de distâncias.

Definição 6.4. A **Distância de Mahalanobis** é uma medida de distância estatística que leva em conta a correlação entre as variáveis e é invariante à escala.

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)' \mathbf{S}^{-1} (\mathbf{x}_i - \mathbf{x}_j)}$$

Onde $\mathbf{S}^{-1}$ é a inversa da matriz de covariâncias amostral. Ela mede a distância entre os pontos em unidades de desvio padrão, ajustando a contribuição de cada variável pela estrutura de covariância dos dados. Já encontramos essa forma quadrática no expoente da distribuição Normal Multivariada (Capítulo 4).

6.3 Medidas para Variáveis Binárias

Quando os dados são binários (0 ou 1), a interpretação da distância muda. A distância Euclidiana quadrática, por exemplo, simplesmente conta o número de posições em que os dois vetores discordam.

$$(x_{ij} - x_{kj})^2 = \begin{cases} 0, & \text{se } x_{ij} = x_{kj} \\ 1, & \text{se } x_{ij} \neq x_{kj} \end{cases}$$

O problema é que essa abordagem dá o mesmo peso para uma concordância de `1-1` e uma concordância de `0-0`. Em muitos contextos, a ausência conjunta de uma característica (concordância `0-0`) é menos informativa do que a presença conjunta (concordância `1-1`).

Para lidar com isso, podemos construir uma tabela de contingência para duas observações $\mathbf{x}_i$ e $\mathbf{x}_j$:

	Observação j = 1	Observação j = 0	Total
Obs i = 1	a	b	a+b
Obs i = 0	c	d	c+d
Total	a+c	b+d	p

Onde: - a: número de variáveis onde $x_{ik} = 1$ e $x_{jk} = 1$. - d: número de variáveis onde $x_{ik} = 0$ e $x_{jk} = 0$. - b e c: número de variáveis onde há discordância.

A distância Euclidiana quadrática corresponde a $b + c$.

Definição 6.5. O **Coeficiente de Jaccard** é uma medida de *similaridade* para dados binários que ignora as concordâncias `0-0`.

$$J(\mathbf{x}_i, \mathbf{x}_j) = \frac{a}{a + b + c}$$

A **Distância de Jaccard** é a sua contraparte de dissimilaridade, definida como $1 - J(\mathbf{x}_i, \mathbf{x}_j)$.

Definição 6.6. O **Coeficiente de Correspondência Simples** (*Simple Matching Coefficient*, SMC) considera tanto as presenças (`1-1`) quanto as ausências (`0-0`) como concordâncias. É útil quando a ausência de uma característica é tão informativa quanto a sua presença.

$$SMC = \frac{a+d}{a+b+c+d}$$

Definição 6.7. O **Coeficiente de Dice** (ou Sørensen-Dice) é outra medida de similaridade que, assim como Jaccard, ignora as concordâncias `0-0`. No entanto, ele dá um peso maior às concordâncias `1-1`.

$$Dice = \frac{2a}{2a+b+c}$$

Definição 6.8. O **Coeficiente de Russell-Rao** é uma medida mais simples que calcula a proporção de presenças conjuntas em relação ao total de variáveis.

$$RR = \frac{a}{a+b+c+d}$$

6.4 Matriz de Distâncias

Uma vez escolhida a medida de dissimilaridade, é comum pré-calcular todas as distâncias entre os pares de observações e organizá-las em uma **matriz de distâncias D**, de dimensão $n \times n$.

$$\mathbf{D} = \begin{pmatrix} 0 & d(\mathbf{x}_1, \mathbf{x}_2) & \cdots & d(\mathbf{x}_1, \mathbf{x}_n) \\ d(\mathbf{x}_2, \mathbf{x}_1) & 0 & \cdots & d(\mathbf{x}_2, \mathbf{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ d(\mathbf{x}_n, \mathbf{x}_1) & d(\mathbf{x}_n, \mathbf{x}_2) & \cdots & 0 \end{pmatrix}$$

Esta matriz é simétrica, ou seja $d(\mathbf{x}_i, \mathbf{x}_j) = d(\mathbf{x}_j, \mathbf{x}_i)$, e possui zeros na diagonal principal. Ela serve como a entrada para muitos algoritmos de agrupamento, especialmente os hierárquicos.

Parte II

Parte II: Métodos multivariados

7 Análise de Componentes Principais

A Análise de Componentes Principais (ACP ou *PCA* do acrônimo em inglês) é uma técnica estatística multivariada que transforma um conjunto de variáveis possivelmente correlacionadas em um novo conjunto de variáveis não correlacionadas, chamadas de **componentes principais**. O objetivo primário da ACP é a **redução de dimensionalidade**: representar a variabilidade presente nos dados originais com um número menor de variáveis, minimizando a perda de informação.

Cada componente principal é uma combinação linear das variáveis originais. O primeiro componente principal é construído para capturar a maior variabilidade possível nos dados. O segundo componente principal, ortogonal ao primeiro, captura a maior parte da variabilidade restante, e assim por diante. Ao final, o número de componentes principais é igual ao número de variáveis originais, mas a expectativa é que os primeiros componentes concentrem a maior parte da informação relevante.

Geometricamente, a ACP é uma projeção do espaço original de variáveis para um outro espaço com características mais interessantes: A variância dos dados é concentrada em direções especificas (os componentes principais) e não existem correlações entre os novos eixos.

Para construir a intuição geométrica, vamos começar com um exemplo simples, em duas dimensões.

Exemplo 7.1. Suponha que coletamos dados de duas variáveis, **Peso** (em kg) e **Altura** (em cm), de um grupo de 10 pessoas.

Tabela 7.1: Tabela de dados com Peso (kg) e Altura (cm).

	Peso	Altura
	65	170
	72	182
	58	165
	81	190
	75	178
	60	168
	68	175
	70	172
	78	185
	62	169
Média	68.90	175.40
Variância	59.88	66.71

Ao plotarmos esses dados (já centralizados), obtemos a nuvem de pontos abaixo. O sistema de eixos em preto (Peso, Altura) é a nossa perspectiva padrão.

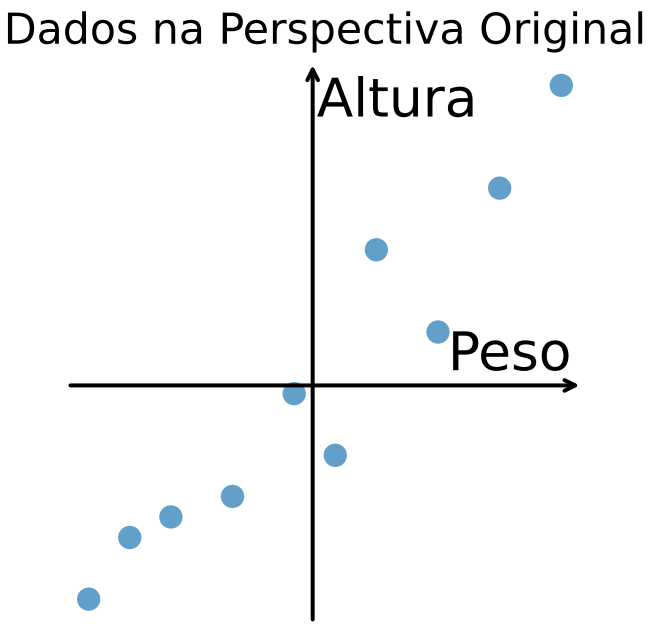

Figura 7.1: Diagrama de dispersão para os dados de Peso e Altura.

Observando o gráfico, notamos que a nuvem de pontos forma uma elipse inclinada, o que indica uma correlação entre Peso e Altura. Descrever os dados usando os eixos originais é perfeitamente válido, mas talvez não seja a forma mais eficiente. A maior parte da variabilidade ocorre ao longo de uma diagonal.

A ACP propõe uma rotação dos eixos para que eles se alinhem melhor com a estrutura dos dados. O resultado é um novo sistema de eixos, os **Componentes Principais** (CP_1 e CP_2), como mostrado abaixo.

O primeiro componente, CP_1, agora aponta na direção de maior "alongamento" da nuvem de pontos. O segundo, CP_2, é perpendicular ao primeiro e aponta na direção de maior variabilidade restante. Encontramos uma nova perspectiva que descreve a estrutura dos dados de forma mais natural e eficiente.

Em casos com mais de duas variáveis (p > 2), a lógica se estende: O **i-ésimo componente principal (**CP_i) aponta para a direção de maior variabilidade, sob a restrição de ser ortogonal (não correlacionado) a todos os componentes anteriores, $\text{Cov}[CP_j, CP_i] = 0 \, \forall \, j < i$.

Uma intuição geométrica para o problema é buscar o ângulo θ de rotação do eixo das variáveis tal que a variância dos componentes seja máxima. Essa rotação é simples de ser observada nesse exemplo bidimensional (veja Figura Figura 7.2).

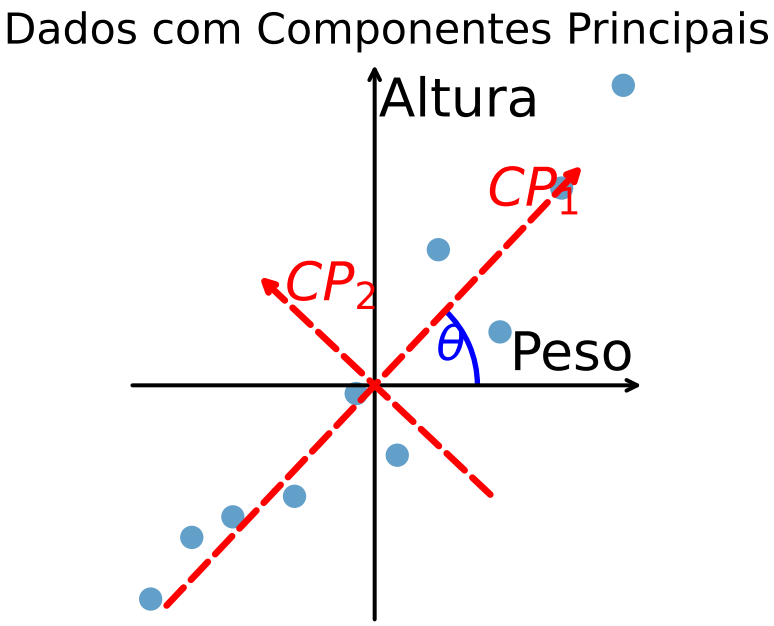

Figura 7.2: O sistema de eixos rotacionado (vermelho) se alinha com a máxima dispersão (variância) dos dados.

7.1 Variância como Medida de Informação

A essa altura, você deve estar se perguntando: por que a direção do "maior alongamento" é a mais importante? Em estatística, a **variância** é frequentemente usada como uma medida de **informação**. Uma variável com alta variância indica que seus valores são bem espalhados, o que nos ajuda a diferenciar as observações. Se a variância fosse zero, todos os pontos seriam idênticos, não nos fornecendo nenhuma informação sobre suas diferenças.

A ACP utiliza essa ideia para encontrar os eixos mais informativos. Ao rotacionar o sistema de coordenadas, ela não altera a variabilidade total dos dados, mas a redistribui de forma inteligente.

Definição 7.1. A **variância total** de um conjunto de dados com p variáveis é a soma das variâncias de cada variável individual. Matematicamente, se $\mathbf{x} = (X_1, \ldots, X_p)'$ é o vetor de variáveis aleatórias com matriz de covariâncias $\mathbf{\Sigma}$, a variância total é definida como:

$$\text{Variância Total} = \sum_{j=1}^{p} \text{Var}(X_j) = \sum_{j=1}^{p} \sigma_{jj} = tr(\mathbf{\Sigma})$$

Onde σ_{jj} é a variância da j-ésima variável e $tr(\mathbf{\Sigma})$ é o traço da matriz de covariâncias (a soma dos elementos da diagonal principal). Essa medida representa a dispersão total na nuvem de pontos,

somando a variabilidade em cada uma das direções dos eixos originais.

Exemplo 7.2. Voltando ao exemplo Exemplo 7.1, as variâncias das variáveis originais são:

- Variância do Peso: 59.88
- Variância da Altura: 66.71
- **Variância Total Original: 126.59**

Após a rotação, as variâncias ao longo dos novos eixos (os componentes principais) são:

- Variância de CP_1: **123.55**
- Variância de CP_2: **3.04**
- **Variância Total dos Componentes: 126.59**

Dois fatos cruciais se destacam:

1. **A variância total é conservada.** A soma das variâncias é a mesma nos dois sistemas de eixos. Nenhuma informação foi perdida; o ponto de vista foi apenas alterado.
2. **A variância foi eficientemente redistribuída.** O primeiro componente, CP_1, agora concentra **97.60%** da variância total. Isso significa que, se quiséssemos reduzir nossos dados de 2D para 1D, poderíamos manter apenas o CP_1 e ainda reter a maior parte da informação original. Essa é a essência da redução de dimensionalidade com ACP.

7.2 A Formalização Matemática

Com a intuição geométrica estabelecida, podemos formalizar a Análise de Componentes Principais. O objetivo é transformar um conjunto de variáveis correlacionadas $\mathbf{x} = (X_1, \ldots, X_p)'$ em um novo conjunto de variáveis não correlacionadas, os **componentes principais** $\mathbf{y} = (Y_1, \ldots, Y_p)'$. Cada componente é uma combinação linear das variáveis originais:

$$\begin{aligned} Y_1 &= e_{11}X_1 + e_{12}X_2 + \cdots + e_{1p}X_p = \mathbf{e}_1'\mathbf{x} \\ Y_2 &= e_{21}X_1 + e_{22}X_2 + \cdots + e_{2p}X_p = \mathbf{e}_2'\mathbf{x} \\ &\vdots \\ Y_p &= e_{p1}X_1 + e_{p2}X_2 + \cdots + e_{pp}X_p = \mathbf{e}_p'\mathbf{x} \end{aligned}$$

Em notação matricial, a transformação pode ser escrita de forma compacta:

$$\mathbf{Y} = \mathbf{E}'\mathbf{X} \tag{7.1}$$

Onde $\mathbf{Y}$ é o vetor $p \times 1$ de componentes principais e $\mathbf{E}$ é a matriz $p \times p$ cujas colunas são os vetores de coeficientes $\mathbf{e}_k$.

Esses componentes são construídos para satisfazer duas condições fundamentais:

1. **Variâncias Ordenadas**: A variância do primeiro componente é a maior possível, a do segundo é a maior possível entre as direções não correlacionadas com o primeiro, e assim por diante. Ou seja, $\text{Var}(Y_1) \geq \text{Var}(Y_2) \geq \cdots \geq \text{Var}(Y_p)$.
2. **Não Correlacionados**: Os componentes são ortogonais entre si, o que significa que $\text{Cov}[Y_i, Y_k] = 0$ para todo $i \neq k$.

7.2.1 O Problema de Maximização

O primeiro componente principal, $Y_1 = \mathbf{e}_1'\mathbf{x}$, é a combinação linear com variância máxima. A variância de Y_1 é dada por:

$$\text{Var}(Y_1) = \text{Var}(\mathbf{e}_1'\mathbf{x}) = \mathbf{e}_1'\text{Var}(\mathbf{x})\mathbf{e}_1 = \mathbf{e}_1'\boldsymbol{\Sigma}\mathbf{e}_1$$

Onde $\boldsymbol{\Sigma}$ é a matriz de covariâncias de $\mathbf{x}$. Para evitar que a variância seja aumentada simplesmente inflando os coeficientes em $\mathbf{e}_1$, impomos a restrição de que seu comprimento seja unitário, $\mathbf{e}_1'\mathbf{e}_1 = 1$. Formalmente, o problema de maximização para o primeiro componente principal se torna:

$$\begin{aligned} \max_{\mathbf{e}_1} \quad & \mathbf{e}_1'\boldsymbol{\Sigma}\mathbf{e}_1 \\ \text{sujeito a} \quad & \mathbf{e}_1'\mathbf{e}_1 = 1 \end{aligned}$$

Para maximizar a variância sujeita à restrição, utilizamos o método dos multiplicadores de Lagrange. A função a ser maximizada é:

$$L(\mathbf{e}_1, \lambda_1) = \mathbf{e}_1'\boldsymbol{\Sigma}\mathbf{e}_1 - \lambda_1(\mathbf{e}_1'\mathbf{e}_1 - 1)$$

Derivando em relação a $\mathbf{e}_1$ e igualando a zero, obtemos:

$$\frac{\partial L}{\partial \mathbf{e}_1} = 2\boldsymbol{\Sigma}\mathbf{e}_1 - 2\lambda_1\mathbf{e}_1 = 0$$

O que nos leva à equação fundamental de autovalores e autovetores:

$$\boldsymbol{\Sigma}\mathbf{e}_1 = \lambda_1\mathbf{e}_1$$

Esta equação mostra que o vetor de coeficientes $\mathbf{e}_1$ deve ser um autovetor da matriz de covariâncias $\boldsymbol{\Sigma}$. Para encontrar a variância, pré-multiplicamos a equação por $\mathbf{e}_1'$:

$$\mathbf{e}_1'\boldsymbol{\Sigma}\mathbf{e}_1 = \lambda_1\mathbf{e}_1'\mathbf{e}_1$$

Como $\text{Var}(Y_1) = \mathbf{e}_1' \mathbf{\Sigma} \mathbf{e}_1$ e a restrição é $\mathbf{e}_1' \mathbf{e}_1 = 1$, temos:

$$\text{Var}(Y_1) = \lambda_1$$

Para maximizar a variância de Y_1, devemos escolher o maior autovalor possível. Portanto, λ_1 é o **maior autovalor** de $\mathbf{\Sigma}$, e $\mathbf{e}_1$ é o **autovetor** correspondente.

> **Nota**
>
> A matriz de covariâncias $\mathbf{\Sigma}$ é, por construção, uma matriz simétrica e positiva-semidefinida. Conforme discutido em Seção 5.3, o **Teorema Espectral** garante que os autovalores de tal matriz são reais e não-negativos, e que seus autovetores correspondentes a autovalores distintos são ortogonais. Esta propriedade é fundamental para a existência e unicidade dos componentes principais.

> **Nota**
>
> A demonstração acima, utilizando multiplicadores de Lagrange, é uma maneira moderna e elegante de conduzir a derivação do problema de maximização. Uma abordagem clássica restringe a norma de e através do quociente,
>
> $$\text{Var}(Y_1) = \max_{e_1} \frac{\mathbf{e}_1' \Sigma \mathbf{e}_1}{\mathbf{e}_1' \mathbf{e_1}}$$
>
> Este é um problema clássico na álgebra linear (ver Seção 5.3). Um teorema fundamental afirma que para qualquer matriz simétrica A, o máximo da forma quadrática $\mathbf{x}' A \mathbf{x}$, sujeito à restrição $\mathbf{x}'\mathbf{x} = 1$, é o **maior autovalor** de A. O vetor $\mathbf{x}$ que atinge esse máximo é o autovetor correspondente. Como a matriz de covariâncias $\mathbf{\Sigma}$ é simétrica, este teorema se aplica diretamente ao nosso problema.

7.2.2 Componentes Subsequentes

Uma vez encontrada a primeira direção de máxima variância, o segundo componente principal, $Y_2 = \mathbf{e}_2' \mathbf{x}$, busca capturar o máximo da variabilidade *restante*, sob a condição de ser não correlacionado com Y_1. A condição de componentes não correlacionados garante que a informação presente no segundo componente principal não é redundante com relação aquela já presente no primeiro. Formalmente, temos:

$$Cov(Y_1, Y_2) = Cov(\mathbf{e}_1' \mathbf{x}, \mathbf{e}_2' \mathbf{x}) = \mathbf{e}_1' \mathbf{\Sigma} \mathbf{e}_2$$

Como $\mathbf{e}_1$ é o primeiro autovetor, temos $\mathbf{\Sigma} \mathbf{e}_1 = \lambda_1 \mathbf{e}_1$. Pela simetria de $\mathbf{\Sigma}$, podemos transpor essa equação para obter $\mathbf{e}_1' \mathbf{\Sigma} = \lambda_1 \mathbf{e}_1'$. Assim:

$$\mathbf{e}_1'\mathbf{\Sigma}\mathbf{e}_2 = (\lambda_1\mathbf{e}_1')\mathbf{e}_2 = \lambda_1\mathbf{e}_1'\mathbf{e}_2$$

Logo:

$$Cov(Y_1, Y_2) = 0 \iff \mathbf{e}_1'\mathbf{e}_2 = 0$$

Com essa condição bem definida, o problema para o segundo componente se torna:

$$\begin{aligned} &\max_{\mathbf{e}_2} \quad \mathbf{e}_2'\mathbf{\Sigma}\mathbf{e}_2 \\ &\text{sujeito a} \quad \begin{cases} \mathbf{e}_2'\mathbf{e}_2 = 1 \\ \mathbf{e}_1'\mathbf{e}_2 = 0 \end{cases} \end{aligned}$$

A função Lagrangiana agora inclui dois multiplicadores, λ_2 e ϕ:

$$L(\mathbf{e}_2, \lambda_2, \phi) = \mathbf{e}_2'\mathbf{\Sigma}\mathbf{e}_2 - \lambda_2(\mathbf{e}_2'\mathbf{e}_2 - 1) - \phi(\mathbf{e}_1'\mathbf{e}_2 - 0)$$

Derivando em relação a $\mathbf{e}_2$ e igualando a zero, temos:

$$\frac{\partial L}{\partial \mathbf{e}_2} = 2\mathbf{\Sigma}\mathbf{e}_2 - 2\lambda_2\mathbf{e}_2 - \phi\mathbf{e}_1 = \mathbf{0}$$

Pré-multiplicando por $\mathbf{e}_1'$:

$$2\mathbf{e}_1'\mathbf{\Sigma}\mathbf{e}_2 - 2\lambda_2\mathbf{e}_1'\mathbf{e}_2 - \phi\mathbf{e}_1'\mathbf{e}_1 = 0$$

Sabendo que:

- $\mathbf{e}_1'\mathbf{\Sigma} = \lambda_1\mathbf{e}_1'$
- $\mathbf{e}_1'\mathbf{e}_2 = 0$
- $\mathbf{e}_1'\mathbf{e}_1 = 1$

A equação se simplifica a $\phi = 0$. Substituindo $\phi = 0$ de volta na derivada, a equação se torna:

$$\mathbf{\Sigma}\mathbf{e}_2 = \lambda_2\mathbf{e}_2$$

Assim, $\mathbf{e}_2$ é o autovetor de $\mathbf{\Sigma}$ correspondente ao autovalor λ_2. Como λ_1 foi o maior autovalor, para maximizar a variância de Y_2, λ_2 deve ser o **segundo maior autovalor**. Este processo se generaliza para os componentes subsequentes.

Este processo continua: o k-ésimo componente principal (Y_k) é definido pelo autovetor $\mathbf{e}_k$ associado ao k-ésimo maior autovalor λ_k, garantindo que $\text{Var}(Y_k) = \lambda_k$ e que todos os componentes sejam mutuamente não correlacionados.

É neste ponto que a conexão com a Seção 5.3 se torna explícita. A matriz $\mathbf{P}$ (Equação 7.1), cujas colunas são os autovetores da matriz de covariâncias $\mathbf{\Sigma}$, é exatamente a mesma matriz $\mathbf{P}$ da decomposição espectral $\mathbf{\Sigma} = \mathbf{P}\Lambda\mathbf{P}'$. Além disso, Λ é uma matriz diagonal contendo a variância de cada componente principal. Logo, podemos obter todos os componentes principais de maneira prática e simultânea através da decomposição espectral.

7.3 A Importância do Pré-processamento dos Dados

A Análise de Componentes Principais é, em sua essência, uma análise de variabilidade. A forma como medimos essa variabilidade impacta diretamente o resultado. Dois pré-processamentos são cruciais: a **centralização** e o **escalonamento**.

7.3.1 Centralização

Na Análise de Componentes Principais (ACP), a centralização dos dados — ou seja, a subtração da média de cada variável — é uma etapa fundamental não apenas para o cálculo da matriz de covariâncias, mas também para a projeção dos dados nos componentes principais.

Ao projetar os dados em um componente $\mathbf{e}_i$, é imprescindível que a projeção seja feita a partir dos dados centralizados, ou seja:

$$Y_i = \mathbf{e}_i^T(\mathbf{x} - \bar{\mathbf{x}})$$

Esse detalhe é essencial porque os autovetores da ACP são obtidos com base na matriz de covariâncias, a qual descreve a dispersão dos dados em torno da média, e não em torno da origem. Se aplicarmos a projeção diretamente sobre $\mathbf{x}$, sem subtrair a média, os componentes resultantes não representarão adequadamente as direções de maior variabilidade — e sim uma combinação da dispersão com a posição média dos dados.

Portanto, para que os componentes principais preservem a interpretação correta como combinações lineares que explicam a variância dos dados em torno do centro da nuvem de pontos, é indispensável que tanto o cálculo da matriz de covariâncias quanto a projeção dos dados utilizem os dados centralizados.

> **!** Importante
>
> No contexto de ACP, é comum e prático denotar por $\mathbf{x}$ o vetor de variáveis já centralizado. Utilizamos esse abuso de notação durante esse capítulo para simplificação do texto sem perda de

generalidade.

7.3.2 Por que Escalonar? O Dilema da Covariância vs. Correlação

A Análise de Componentes Principais (ACP) é sensível à **escala** das variáveis. Se uma variável tiver uma variância numericamente muito maior que as outras — mesmo que apenas por causa da sua unidade de medida — ela poderá dominar os primeiros componentes principais.

Imagine incluir uma terceira variável no conjunto Altura/Peso: a **renda mensal**, medida em Reais. As variâncias poderiam ser aproximadamente:

- **Altura:** 80 cmˆ2
- **Peso:** 60 kgˆ2
- **Renda:** 4.000.000 (RS)ˆ2

Nesse cenário, a variância da Renda é **milhares de vezes maior** que a das outras variáveis. Se aplicarmos a ACP diretamente na **matriz de covariâncias**, o primeiro componente principal será fortemente direcionado pela Renda, mesmo que sua correlação com as demais variáveis seja baixa. Isso ocorre porque a ACP estará apenas "seguindo" a direção da variável com maior variância — não necessariamente a mais informativa.

Para evitar esse viés, escalonamos as variáveis: cada uma é dividida por seu desvio padrão. Isso padroniza todas para variância igual a 1. Ao fazer isso, estamos na prática realizando a ACP sobre a **matriz de correlação** ($\mathbf{R}$) em vez da matriz de covariâncias ($\mathbf{\Sigma}$).

Vantagem do escalonamento:
Usar a matriz de correlação "democratiza" a análise. Todas as variáveis começam com a **mesma importância inicial** (variância 1), e a ACP passa a capturar a **estrutura de correlações**, ao invés de ser enviesada pelas **diferenças de escala**.

Quando usar a matriz de covariâncias?
Somente quando todas as variáveis estão na **mesma unidade de medida** e possuem uma interpretação comparável. Por exemplo, comparar a temperatura em Celsius em diferentes regiões pode fazer sentido sem escalonamento. Fora isso, a matriz de **correlação** é geralmente a escolha mais robusta e segura.

7.4 Componentes Principais Populacionais vs. Amostrais

Até este ponto, discutimos os componentes principais em um contexto populacional, onde a matriz de covariâncias $\mathbf{\Sigma}$ (ou correlação $\mathbf{R}$) e seus autovalores λ_k e autovetores $\mathbf{e}_k$ são conhecidos. Na prática,

quase sempre trabalhamos com uma amostra de dados. Nesse caso, não conhecemos os verdadeiros parâmetros populacionais e devemos estimá-los.

Os **componentes principais amostrais** são obtidos da mesma maneira, mas usando a matriz de covariâncias amostral $\mathbf{S}$ (ou a matriz de correlação amostral $\mathbf{R}$). As quantidades resultantes são estimativas dos seus análogos populacionais:

- O k-ésimo autovalor amostral, $\hat{\lambda}_k$, é uma estimativa de λ_k.
- O k-ésimo autovetor amostral, $\hat{\mathbf{e}}_k$, é uma estimativa de $\mathbf{e}_k$.
- O k-ésimo componente principal amostral, $\hat{Y}_k = \hat{\mathbf{e}}_k'\mathbf{x}$, é uma estimativa de Y_k.

A teoria e a interpretação permanecem as mesmas. Para simplificar a notação, ao longo deste capítulo, omitimos o acento circunflexo (ˆ), mas é importante lembrar que, na aplicação prática, estamos sempre lidando com estimativas amostrais.

7.5 Escolhendo o Número de Componentes

A principal vantagem da ACP é a redução de dimensionalidade. Mas como decidimos quantos componentes ($q < p$) reter? A escolha de q envolve um trade-off entre a simplicidade (poucos componentes) e a fidelidade aos dados originais (muitos componentes). Não existe uma regra única, mas sim um conjunto de critérios que devem ser avaliados em conjunto.

7.5.1 Critério da Variância Explicada Acumulada

Este é o critério mais comum. Calculamos a proporção da variância total explicada por cada componente e acumulamos essa proporção.

$$\text{Proporção da Variância por } CP_k = \frac{\lambda_k}{\sum_{j=1}^{p} \lambda_j}$$

Em seguida, escolhemos o menor número de componentes q cuja variância explicada acumulada atinja um limiar satisfatório, geralmente entre 70% e 90%. A escolha do limiar depende do contexto da análise.

7.5.2 Critério do Autovalor (Critério de Kaiser)

Proposto por Henry Kaiser, este critério sugere reter apenas os componentes cujos autovalores (λ_k) são maiores que 1. A intuição por trás dessa regra é mais clara quando a ACP é aplicada sobre a matriz de correlação. Nesse caso, as variáveis originais são padronizadas para ter variância 1. Um componente com autovalor (variância) menor que 1 está, portanto, explicando menos variabilidade do que uma

única variável original. Reter tal componente não traria uma "economia" de informação", tornando-o um candidato à exclusão.

7.5.3 Scree Plot (Gráfico de Cotovelo)

O Scree Plot, proposto por Raymond Cattell, é uma ferramenta visual que nos ajuda a identificar o número ideal de componentes. Ele é um gráfico de linha dos autovalores (variâncias dos componentes) em ordem decrescente.

Tipicamente, o gráfico mostra uma queda acentuada nos primeiros autovalores, seguida por um nivelamento gradual para os autovalores restantes. O ponto onde a curva "dobra" ou forma um "cotovelo" (*elbow*) é considerado o ponto de corte. A ideia é reter os componentes que aparecem antes do cotovelo, pois eles são os que contribuem mais significativamente para a variância total. Os componentes após o cotovelo formam o "cascalho" (*scree*) na base de uma montanha e são considerados "ruído".

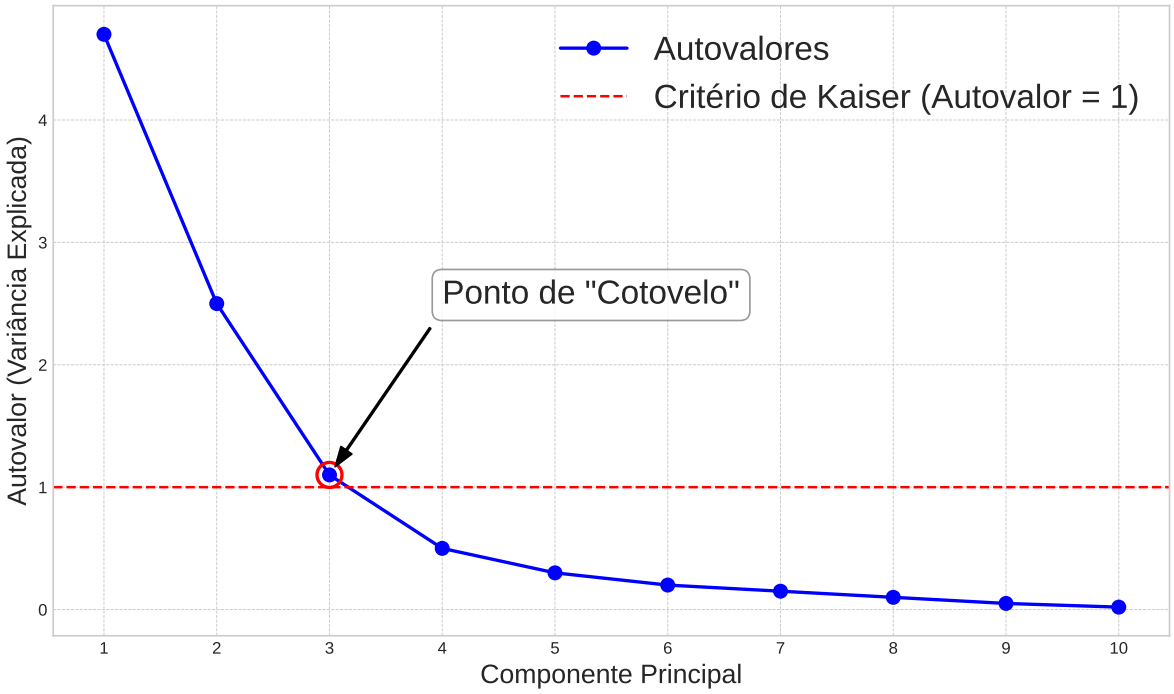

Figura 7.3: Exemplo de um Scree Plot. O 'cotovelo' em k=3 sugere a retenção de 3 componentes.

7.6 Interpretando os Componentes Principais

Uma vez que selecionamos o número de componentes a reter, o passo final é a **interpretação**. O que esses novos eixos, que são combinações de nossas variáveis originais, realmente significam?

Os coeficientes e_{kj} do autovetor $\mathbf{e}_k$ são chamados de **cargas** (*loadings*) e representam o peso da variável original X_j na formação do componente Y_k. Embora as cargas sejam importantes, a sua interpretação pode ser complicada, pois sua magnitude depende das unidades das variáveis originais.

Uma medida mais interpretável é a **correlação entre os componentes principais e as variáveis originais**, $\text{Cor}(Y_k, X_j)$. Ela nos diz o quão "alinhado" um componente está com cada variável original, numa escala padronizada de -1 a 1. A fórmula para essa correlação é:

$$\text{Cor}(Y_k, X_j) = \frac{e_{kj}\sqrt{\lambda_k}}{\sqrt{s_{jj}}}$$

Onde:

- e_{kj} é o j-ésimo elemento do autovetor $\mathbf{e}_k$ (a carga da variável j no componente k).
- λ_k é o autovalor (variância) do componente k.
- s_{jj} é a variância da variável original X_j.

Derivação da Fórmula de Correlação

A correlação entre o componente $Y_k = \mathbf{e}_k'\mathbf{x}$ e a variável X_j é:

$$\text{Cor}(Y_k, X_j) = \frac{\text{Cov}(Y_k, X_j)}{\sqrt{\text{Var}(Y_k)}\sqrt{\text{Var}(X_j)}}$$

Sabemos que $\text{Var}(Y_k) = \lambda_k$ e $\text{Var}(X_j) = s_{jj}$. Para a covariância:

$$\text{Cov}(Y_k, X_j) = \text{Cov}(\mathbf{e}_k'\mathbf{x}, X_j) = \mathbf{e}_k'\text{Cov}(\mathbf{x}, X_j) = \mathbf{e}_k'\mathbf{\Sigma}_{(\cdot j)}$$

onde $\mathbf{\Sigma}_{(\cdot j)}$ é a j-ésima coluna de $\mathbf{\Sigma}$. Utilizando $\mathbf{\Sigma}\mathbf{e}_k = \lambda_k\mathbf{e}_k$, podemos mostrar que $\text{Cov}(Y_k, X_j) = e_{kj}\lambda_k$. Portanto:

$$\text{Cor}(Y_k, X_j) = \frac{e_{kj}\lambda_k}{\sqrt{\lambda_k}\sqrt{s_{jj}}} = \frac{e_{kj}\sqrt{\lambda_k}}{\sqrt{s_{jj}}}$$

Quando a ACP é realizada sobre a **matriz de correlação** (ou seja, com dados padronizados), as variâncias s_{jj} são todas iguais a 1. Nesse caso, a fórmula simplifica para $\text{Cor}(Y_k, X_j) = e_{kj}\sqrt{\lambda_k}$. As correlações se tornam proporcionais às cargas, facilitando a interpretação.

Exemplo 7.3. Suponha que uma ACP sobre a matriz de correlação de 4 variáveis resultou nos seguintes autovetores e autovalores para os dois primeiros componentes:

	CP_1 ($\lambda_1 = 2.5$)	CP_2 ($\lambda_2 = 1.1$)
X_1 (Altura)	0.52	-0.45
X_2 (Peso)	0.55	-0.35
X_3 (Idade)	0.42	0.60
X_4 (Renda)	0.50	0.55

As correlações $\text{Cor}(Y_k, X_j) = e_{kj}\sqrt{\lambda_k}$ são:

	CP_1	CP_2
X_1	$0.52 \times \sqrt{2.5} = 0.82$	$-0.45 \times \sqrt{1.1} = -0.47$
X_2	$0.55 \times \sqrt{2.5} = 0.87$	$-0.35 \times \sqrt{1.1} = -0.37$
X_3	$0.42 \times \sqrt{2.5} = 0.66$	$0.60 \times \sqrt{1.1} = 0.63$
X_4	$0.50 \times \sqrt{2.5} = 0.79$	$0.55 \times \sqrt{1.1} = 0.58$

Interpretação: CP_1 está fortemente correlacionado com todas as variáveis (especialmente Peso e Altura), podendo ser interpretado como um componente de "tamanho/status geral". CP_2 contrasta Altura/Peso (correlações negativas) contra Idade/Renda (correlações positivas), sugerindo um eixo de "maturidade".

A ferramenta visual mais adequada para essa a interpretação da ACP é o **biplot**. O termo "biplot" significa "dois plots" (plot duplo), pois ele sobrepõe duas informações em um único gráfico:

1. **Os scores:** As coordenadas das observações no novo espaço dos componentes principais.
2. **Os loadings:** As contribuições das variáveis originais para a criação desses componentes.

O resultado é um mapa rico que mostra não apenas como as observações se agrupam, mas *por que* elas se agrupam daquela maneira. A interpretação de um biplot segue uma lógica visual. Vamos quebrar em partes:

1. **Eixos (Componentes Principais)**: O eixo horizontal é o CP1 e o vertical é o CP2. Eles são as "réguas" do nosso novo mapa e representam as direções de maior variabilidade nos dados. A porcentagem de variância que cada um explica é mostrada nos seus rótulos.
2. **Pontos (Observações)**: Cada ponto no gráfico é uma observação.
    - **Proximidade**: Pontos próximos uns dos outros representam observações com perfis semelhantes (conforme capturado pelos dois primeiros CPs).
    - **Agrupamentos**: Grupos de pontos (clusters) indicam subpopulações nos dados.

3. **Vetores (Variáveis Originais)**: Cada seta (vetor) representa uma das variáveis originais.

    - **Direção**: A direção da seta indica como a variável contribui para os dois componentes. Uma seta que aponta para a direita indica uma forte contribuição positiva para o CP1. Uma que aponta para cima, uma forte contribuição positiva para o CP2.
    - **Comprimento**: O comprimento da seta é proporcional a quão bem a variável é representada no espaço 2D do biplot. Setas mais longas significam que a variável tem uma forte influência nos componentes mostrados e é bem representada no gráfico. Setas curtas são menos importantes para os dois primeiros CPs ou sua variabilidade está melhor explicada em outros componentes (CP3, CP4, etc.).
    - **Relações entre Variáveis**: O ângulo entre os vetores nos informa sobre a correlação entre as variáveis originais.
        - **Ângulo pequeno (< 90°)**: As variáveis são positivamente correlacionadas.
        - **Ângulo de ~90°**: As variáveis não são correlacionadas.
        - **Ângulo obtuso (> 90°)**: As variáveis são negativamente correlacionadas.

4. **Relação entre Pontos e Vetores**: Para entender o perfil de um ponto (ou grupo de pontos), projete-o ortogonalmente sobre os vetores das variáveis. Se a projeção de um ponto cai na direção de um vetor, aquela observação tem um valor alto para aquela variável. Se cai na direção oposta, tem um valor baixo.

Com essas regras em mente, vamos analisar um biplot genérico.

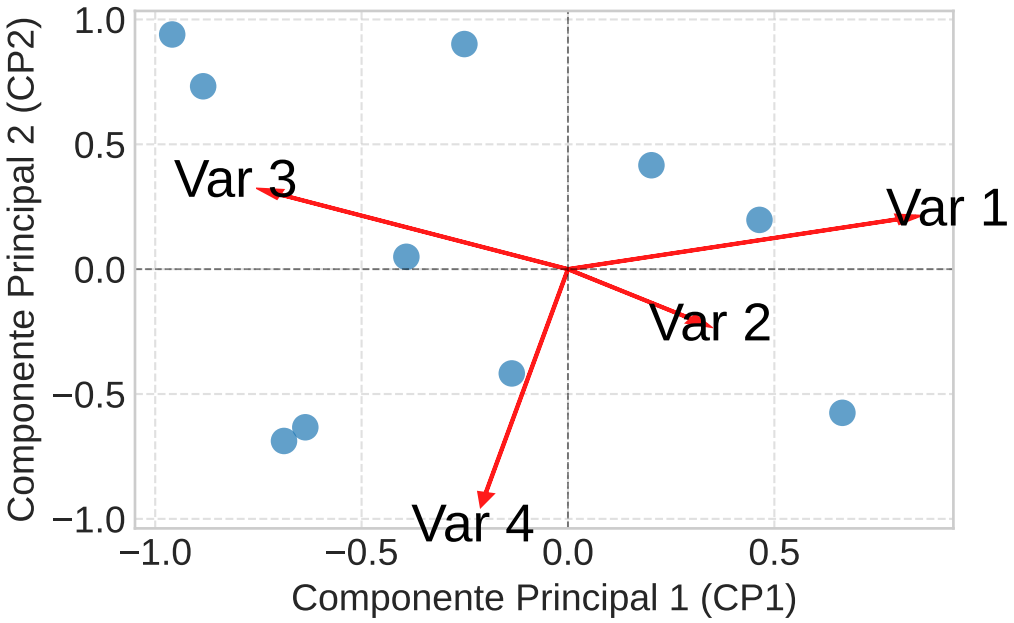

Figura 7.4: Exemplo de um biplot genérico para ilustrar a interpretação dos seus elementos. Os pontos representam as observações e as setas, as variáveis originais.

8 Análise Fatorial

A Análise Fatorial é uma técnica estatística utilizada para descrever a estrutura de covariância entre um conjunto de variáveis observadas. A hipótese central é que essa estrutura é gerada por um número menor de variáveis latentes não observáveis, denominadas **fatores comuns**.

Começamos com uma intuição. Suponha que temos as seguintes variáveis de gastos para diferentes famílias:

- X_1: Gasto em educação
- X_2: Gasto em cultura
- X_3: Gasto em alimentação

É razoável supor que essas variáveis sejam correlacionadas. Mais do que isso, pode existir um fator latente, como a **renda familiar** (F_1), que influencia todos esses gastos. A Análise Fatorial busca formalizar e quantificar essa relação.

8.1 O Modelo Fatorial Ortogonal

O modelo supõe que cada variável observada é linearmente dependente de um conjunto de fatores comuns, somado a um termo de variância individual, ou específico.

Definição 8.1. Seja $\mathbf{x}$ um vetor aleatório de p variáveis observadas com vetor de médias $\boldsymbol{\mu}$ e matriz de covariâncias $\boldsymbol{\Sigma}$. O modelo fatorial com m fatores comuns ($m < p$) postula que $\mathbf{x}$ é linearmente dependente de m fatores comuns $F_1, F_2, \ldots, F_m$ e p termos de erro $\epsilon_1, \epsilon_2, \ldots, \epsilon_p$. Em notação matricial, o modelo é:

$$\mathbf{x}_{(p\times 1)} - \boldsymbol{\mu}_{(p\times 1)} = \mathbf{L}_{(p\times m)}\mathbf{F}_{(m\times 1)} + \boldsymbol{\epsilon}_{(p\times 1)} \tag{8.1}$$

Onde:

- $\mathbf{L}$ é a matriz de **cargas fatoriais**: Uma matriz de pesos responsável por quantificar as relações entre as p variáveis e os m fatores.
- $\mathbf{F}$ é o vetor de **fatores comuns**, ou seja $\mathbf{F} = [F_1, F_2, \ldots, F_m]'$.
- $\boldsymbol{\epsilon}$ é o vetor de **erros**, ou **variâncias específicas**, ou seja, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \ldots, \epsilon_p]'$

Para que o ajuste desse modelo seja factível, as seguintes suposições são feitas para o modelo ortogonal:

1. $E[\mathbf{F}] = \mathbf{0}$ e $Cov(\mathbf{F}) = E[\mathbf{F}\mathbf{F}'] = \mathbf{I}$.
2. $E[\boldsymbol{\epsilon}] = \mathbf{0}$ e $Cov(\boldsymbol{\epsilon}) = E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}'] = \boldsymbol{\Psi}$, onde $\boldsymbol{\Psi}$ é uma matriz diagonal.
3. $Cov(\mathbf{F}, \boldsymbol{\epsilon}) = E[\mathbf{F}\boldsymbol{\epsilon}'] = \mathbf{0}$.

8.2 A Estrutura de Covariância Implícita

As suposições do modelo implicam uma estrutura específica para a matriz de covariâncias $\boldsymbol{\Sigma}$.

Teorema 8.1. *Sob as premissas do modelo fatorial ortogonal (Definição 8.1), a matriz de covariâncias $\boldsymbol{\Sigma}$ do vetor $\mathbf{x}$ é dada por:*

$$\boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}' + \boldsymbol{\Psi} \tag{8.2}$$

Comprovação. A partir do modelo fatorial em Equação 8.1, temos que $\mathbf{x} - \boldsymbol{\mu} = \mathbf{L}\mathbf{F} + \boldsymbol{\epsilon}$. A matriz de covariâncias de $\mathbf{x}$ é, por definição, $\boldsymbol{\Sigma} = E[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})']$. Substituindo a expressão do modelo, obtemos:

$$\begin{aligned}
\boldsymbol{\Sigma} &= E[(\mathbf{L}\mathbf{F} + \boldsymbol{\epsilon})(\mathbf{L}\mathbf{F} + \boldsymbol{\epsilon})'] \\
&= E[(\mathbf{L}\mathbf{F} + \boldsymbol{\epsilon})(\mathbf{F}'\mathbf{L}' + \boldsymbol{\epsilon}')] \\
&= E[\mathbf{L}\mathbf{F}\mathbf{F}'\mathbf{L}' + \mathbf{L}\mathbf{F}\boldsymbol{\epsilon}' + \boldsymbol{\epsilon}\mathbf{F}'\mathbf{L}' + \boldsymbol{\epsilon}\boldsymbol{\epsilon}'] \\
&= \mathbf{L}E[\mathbf{F}\mathbf{F}']\mathbf{L}' + \mathbf{L}E[\mathbf{F}\boldsymbol{\epsilon}'] + E[\boldsymbol{\epsilon}\mathbf{F}']\mathbf{L}' + E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}']
\end{aligned}$$

Pelas suposições do modelo ortogonal (Definição 8.1):

1. $E[\mathbf{F}\mathbf{F}'] = \text{Cov}(\mathbf{F}) = \mathbf{I}$ (os fatores são não correlacionados e têm variância unitária).
2. $E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}'] = \text{Cov}(\boldsymbol{\epsilon}) = \boldsymbol{\Psi}$ (os erros são não correlacionados entre si).
3. $E[\mathbf{F}\boldsymbol{\epsilon}'] = \text{Cov}(\mathbf{F}, \boldsymbol{\epsilon}) = \mathbf{0}$ (os fatores e os erros são não correlacionados).

Substituindo essas esperanças na equação de $\boldsymbol{\Sigma}$, temos:

$$\boldsymbol{\Sigma} = \mathbf{L}\mathbf{I}\mathbf{L}' + \mathbf{L}\mathbf{0} + \mathbf{0}\mathbf{L}' + \boldsymbol{\Psi} = \mathbf{L}\mathbf{L}' + \boldsymbol{\Psi}$$

Isso completa a prova. □

Esta equação decompõe a variância de cada variável X_i em:

- **Comunalidade (h_i^2)**: A porção da variância de X_i explicada pelos m fatores comuns.
- **Variância Específica (ψ_i)**: A porção da variância de X_i não explicada pelos fatores comuns.

Derivação da Comunalidade

A variância de X_i é o elemento diagonal σ_{ii} da matriz $\boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}' + \boldsymbol{\Psi}$. O elemento (i,i) de $\mathbf{L}\mathbf{L}'$ é:

$$[\mathbf{L}\mathbf{L}']_{ii} = \sum_{k=1}^{m} l_{ik} l_{ik} = \sum_{k=1}^{m} l_{ik}^2 = h_i^2$$

onde l_{ik} é a carga da variável i no fator k. Como $\boldsymbol{\Psi}$ é diagonal com elemento ψ_i na posição (i,i):

$$\sigma_{ii} = h_i^2 + \psi_i \quad \Rightarrow \quad \text{Var}(X_i) = \underbrace{\sum_{k=1}^{m} l_{ik}^2}_{\text{comunalidade}} + \underbrace{\psi_i}_{\text{variância específica}}$$

Exemplo 8.1. Suponha que a matriz de covariâncias de um vetor aleatório $\mathbf{x}$ com $p = 4$ variáveis seja:

$$\boldsymbol{\Sigma} = \begin{pmatrix} 19 & 30 & 2 & 12 \\ 30 & 57 & 5 & 23 \\ 2 & 5 & 37 & 47 \\ 12 & 23 & 47 & 68 \end{pmatrix}$$

É possível mostrar que um modelo fatorial com $m = 2$ fatores comuns pode gerar essa estrutura de covariância. Uma solução possível para $\mathbf{L}$ e $\boldsymbol{\Psi}$ é dada por:

$$\mathbf{L} = \begin{pmatrix} 4 & 1 \\ 7 & 2 \\ -1 & 6 \\ 1 & 8 \end{pmatrix}, \quad \boldsymbol{\Psi} = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}$$

O leitor pode verificar que $\boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}' + \boldsymbol{\Psi}$. O modelo decompõe a variância de cada variável.

- Para X_1, a comunalidade é $h_1^2 = 4^2 + 1^2 = 17$, e sua variância total é $Var(X_1) = \sigma_{11} = h_1^2 + \psi_1 = 17 + 2 = 19$.
- Para X_2, a comunalidade é $h_2^2 = 7^2 + 2^2 = 53$, e sua variância total é $Var(X_2) = \sigma_{22} = h_2^2 + \psi_2 = 53 + 4 = 57$.

8.3 Problemas no Modelo Fatorial

- **Existência da Solução**: Nem sempre existe uma solução factível para o modelo fatorial com m fatores. A estimação dos parâmetros, especialmente com um número inadequado de fatores,

pode levar a soluções impróprias, como uma variância específica negativa ($\hat{\psi}_i < 0$), conhecida como **caso de Heywood**. Isso viola a premissa de que ψ_i é uma variância e, portanto, deve ser não-negativa. Geralmente, uma solução imprópria indica que o modelo é inadequado para os dados.

Exemplo 8.2. Considere um modelo de um fator ($m = 1$) para $p = 3$ variáveis, com a seguinte matriz de correlação populacional:

$$\mathbf{R} = \begin{pmatrix} 1.0 & 0.4 & 0.9 \\ 0.4 & 1.0 & 0.7 \\ 0.9 & 0.7 & 1.0 \end{pmatrix}$$

O modelo fatorial para a matriz de correlação é $\mathbf{P} = \mathbf{L}\mathbf{L}' + \mathbf{\Psi}$. Para $m = 1$, as cargas são um vetor $\mathbf{L} = [l_{11}, l_{21}, l_{31}]'$. As covariâncias (correlações) são dadas por $\rho_{ij} = l_{i1}l_{j1}$. Temos o sistema:

- $\rho_{12} = l_{11}l_{21} = 0.4$
- $\rho_{13} = l_{11}l_{31} = 0.9$
- $\rho_{23} = l_{21}l_{31} = 0.7$

Multiplicando as três equações, obtemos $(l_{11}l_{21}l_{31})^2 = 0.4 \times 0.9 \times 0.7 = 0.252$. Isso nos permite resolver para as cargas:

- $l_{11}^2 = (l_{11}l_{21})(l_{11}l_{31})/(l_{21}l_{31}) = (0.4 \times 0.9)/0.7 \approx 0.514$
- $l_{21}^2 = (l_{11}l_{21})(l_{21}l_{31})/(l_{11}l_{31}) = (0.4 \times 0.7)/0.9 \approx 0.311$
- $l_{31}^2 = (l_{11}l_{31})(l_{21}l_{31})/(l_{11}l_{21}) = (0.9 \times 0.7)/0.4 = 1.575$

A comunalidade da terceira variável é $h_3^2 = l_{31}^2 = 1.575$. Como estamos modelando uma matriz de correlação, a variância total de cada variável é 1. A variância específica seria $\psi_3 = 1 - h_3^2 = 1 - 1.575 = -0.575$. Uma variância negativa é impossível, indicando que o modelo de um fator não é apropriado para descrever a estrutura de correlação dada.

- **Indeterminação da Solução (Rotação Fatorial)**: A solução para a matriz de cargas $\mathbf{L}$ não é única. Para qualquer matriz ortogonal $\mathbf{T}$ de dimensão $m \times m$ (ou seja, uma matriz tal que $\mathbf{T}\mathbf{T}' = \mathbf{T}'\mathbf{T} = \mathbf{I}$), podemos definir uma nova matriz de cargas $\mathbf{L}^* = \mathbf{L}\mathbf{T}$ que resulta na mesma matriz de covariâncias.

Isso ocorre porque a parte da covariância explicada pelos fatores, $\mathbf{L}\mathbf{L}'$, permanece inalterada:

$$\mathbf{L}^*(\mathbf{L}^*)' = (\mathbf{L}\mathbf{T})(\mathbf{L}\mathbf{T})' = \mathbf{L}\mathbf{T}\mathbf{T}'\mathbf{L}' = \mathbf{L}(\mathbf{T}\mathbf{T}')\mathbf{L}' = \mathbf{L}\mathbf{I}\mathbf{L}' = \mathbf{L}\mathbf{L}'$$

Portanto, o modelo $\mathbf{\Sigma} = \mathbf{L}^*(\mathbf{L}^*)' + \mathbf{\Psi}$ é equivalente ao modelo original. Essa propriedade é a base para a **rotação fatorial**, um procedimento que busca a solução $\mathbf{L}^*$ mais simples e interpretável, sem alterar o ajuste do modelo.

8.4 Adequabilidade do Modelo Fatorial

Antes de aplicar os métodos de estimação, pode-se avaliar se os dados são adequados para a Análise Fatorial. A premissa fundamental da AF é que as variáveis observadas são correlacionadas e que essa correlação pode ser explicada por fatores latentes. Se as variáveis são ortogonais ou se a correlação entre elas é espúria, o modelo fatorial não é apropriado.

Dois dos principais diagnósticos para verificar a adequabilidade dos dados são o Teste de Esfericidade de Bartlett e a medida de adequação da amostra de Kaiser-Meyer-Olkin (KMO).

8.4.1 Teste de Esfericidade de Bartlett

O Teste de Esfericidade de Bartlett avalia a hipótese nula (H_0) de que a matriz de correlação populacional $\mathbf{P}$ é uma matriz identidade ($H_0 : \mathbf{P} = \mathbf{I}$). Se essa hipótese for verdadeira, as variáveis são não correlacionadas, e não há estrutura latente para ser extraída.

A estatística de teste é baseada no determinante da matriz de correlação amostral $\mathbf{R}$ e, sob H_0, segue aproximadamente uma distribuição Qui-quadrado. Para uma amostra de tamanho *n* e *p* variáveis, a estatística é:

$$\chi^2 = -\left[(n-1) - \frac{2p+5}{6}\right] \ln(|\mathbf{R}|)$$

Esta estatística tem, aproximadamente, uma distribuição χ^2 com $p(p-1)/2$ graus de liberdade. Um p-valor baixo (e.g., < 0.05) leva à rejeição de H_0, indicando que existe correlação suficiente entre as variáveis para justificar a aplicação da Análise Fatorial.

8.4.2 Medida de Adequação da Amostra (KMO)

Enquanto o teste de Bartlett avalia se a matriz de correlação como um todo se desvia significativamente da identidade, a medida de Kaiser-Meyer-Olkin (KMO) quantifica o quão adequados os dados são para a fatorização. O KMO compara a magnitude dos coeficientes de correlação observados com a magnitude dos coeficientes de correlação parcial.

A lógica é que, se as variáveis compartilham fatores comuns, as correlações parciais entre pares de variáveis (controlando pelas outras variáveis) devem ser pequenas. A estatística KMO é calculada como:

$$\text{KMO} = \frac{\sum_{i \neq j} r_{ij}^2}{\sum_{i \neq j} r_{ij}^2 + \sum_{i \neq j} a_{ij}^2}$$

Onde r_{ij} é o coeficiente de correlação simples entre as variáveis X_i e X_j, e a_{ij} é o coeficiente de correlação parcial.

O valor do KMO varia de 0 a 1. Valores mais altos indicam que a Análise Fatorial é mais apropriada. Uma regra prática para a interpretação do KMO é:

- > **0.9**: Maravilhoso
- **0.8 - 0.9**: Meritório
- **0.7 - 0.8**: Razoável
- **0.6 - 0.7**: Medíocre
- **0.5 - 0.6**: Ruim
- < **0.5**: Inaceitável

Valores abaixo de 0.5 sugerem que a Análise Fatorial pode não ser uma boa ideia.

8.5 Métodos de Estimação

Assumindo uma amostra aleatória $\mathbf{x}_1, \ldots, \mathbf{x}_n$ de uma população com matriz de covariâncias $\mathbf{\Sigma}$, o desafio é estimar $\mathbf{L}$ e $\mathbf{\Psi}$ usando a matriz de covariâncias amostral $\mathbf{S}$ ou a matriz de correlação amostral $\mathbf{R}$.

Existem diversos métodos para estimar os parâmetros do modelo fatorial, cada um com suas próprias premissas e propriedades. Alguns dos mais conhecidos incluem:

- Método de Componentes Principais (MCP)
- Método da Máxima Verossimilhança (MMV)
- Método dos Fatores Principais (*Principal Axis Factoring*)
- Mínimos Quadrados Ponderados
- Mínimos Quadrados Generalizados

Neste capítulo, focaremos nos dois métodos mais amplamente utilizados na prática: o Método de Componentes Principais, por sua simplicidade computacional, e o Método da Máxima Verossimilhança, por sua fundamentação estatística robusta.

8.5.1 A Solução por Componentes Principais

O método de componentes principais (MCP) provê uma solução para $\mathbf{L}$ e $\mathbf{\Psi}$ a partir da decomposição espectral da matriz de covariâncias amostral $\mathbf{S}$ (ou da matriz de correlações $\mathbf{R}$).

! ACP vs. AF via Componentes Principais

É importante distinguir entre a **Análise de Componentes Principais** como técnica autônoma (Capítulo 7) e o **Método de Componentes Principais** para estimação em Análise Fatorial.

- Na **ACP**, os componentes são simplesmente combinações lineares das variáveis que capturam a máxima variância; não há modelo estatístico subjacente.
- Na **AF via MCP**, utilizamos a maquinaria da decomposição espectral como *ferramenta computacional* para estimar os parâmetros de um modelo fatorial. A AF assume que existe uma estrutura latente gerando as correlações observadas.

Em resumo: a ACP é uma técnica descritiva; a AF é uma técnica de modelagem que pode usar a ACP como método de estimação.

A ideia é que a matriz $\mathbf{S}$ pode ser decomposta em termos de seus pares de autovalor-autovetor $(\hat{\lambda}_i, \hat{\mathbf{e}}_i)$:

$$\mathbf{S} = \hat{\lambda}_1 \hat{\mathbf{e}}_1 \hat{\mathbf{e}}_1' + \hat{\lambda}_2 \hat{\mathbf{e}}_2 \hat{\mathbf{e}}_2' + \cdots + \hat{\lambda}_p \hat{\mathbf{e}}_p \hat{\mathbf{e}}_p'$$

A estrutura do modelo fatorial é $\mathbf{S} = \hat{\mathbf{L}}\hat{\mathbf{L}}' + \hat{\mathbf{\Psi}}$. O MCP busca uma aproximação para $\mathbf{S}$ retendo apenas os m primeiros componentes, que explicam a maior parte da variabilidade total. A matriz $\hat{\mathbf{L}}\hat{\mathbf{L}}'$ é construída para igualar a contribuição desses componentes:

$$\hat{\mathbf{L}}\hat{\mathbf{L}}' = \hat{\lambda}_1 \hat{\mathbf{e}}_1 \hat{\mathbf{e}}_1' + \hat{\lambda}_2 \hat{\mathbf{e}}_2 \hat{\mathbf{e}}_2' + \cdots + \hat{\lambda}_m \hat{\mathbf{e}}_m \hat{\mathbf{e}}_m'$$

Uma solução explícita para $\hat{\mathbf{L}}$ que satisfaz essa equação é uma matriz $p \times m$ cujas colunas são os autovetores reescalados pelos respectivos autovalores. A matriz $\hat{\mathbf{\Psi}}$ é então definida para garantir que as variâncias do modelo ($\text{diag}(\hat{\mathbf{L}}\hat{\mathbf{L}}' + \hat{\mathbf{\Psi}})$) sejam iguais às variâncias amostrais ($\text{diag}(\mathbf{S})$).

Isso nos leva à seguinte definição formal.

Definição 8.2. Seja $\mathbf{S}$ a matriz de covariância amostral com pares de autovalor-autovetor $(\hat{\lambda}_i, \hat{\mathbf{e}}_i)$. A solução de componentes principais com m fatores é definida por:

- **Matriz de Cargas Estimada ($\hat{\mathbf{L}}$)**:

$$\hat{\mathbf{L}} = [\sqrt{\hat{\lambda}_1}\hat{\mathbf{e}}_1 | \sqrt{\hat{\lambda}_2}\hat{\mathbf{e}}_2 | \ldots | \sqrt{\hat{\lambda}_m}\hat{\mathbf{e}}_m]$$

- **Matriz de Variâncias Específicas Estimada ($\hat{\mathbf{\Psi}}$)**:

$$\hat{\mathbf{\Psi}} = \text{diag}(\mathbf{S} - \hat{\mathbf{L}}\hat{\mathbf{L}}')$$

onde $\hat{\psi}_i = s_{ii} - \sum_{j=1}^{m} \hat{l}_{ij}^2$.

Se a matriz de correlações $\mathbf{R}$ for utilizada, as cargas $\hat{\mathbf{L}}$ são calculadas a partir dos autovalores e autovetores de $\mathbf{R}$, e as variâncias específicas são $\hat{\psi}_i = 1 - \sum_{j=1}^{m} \hat{l}_{ij}^2$.

Por construção, este método força a diagonal da matriz de covariâncias do modelo, $\hat{\mathbf{\Sigma}} = \hat{\mathbf{L}}\hat{\mathbf{L}}' + \hat{\mathbf{\Psi}}$, a ser idêntica à diagonal de $\mathbf{S}$. O ajuste do modelo é então avaliado pela magnitude dos resíduos fora da diagonal. A matriz de resíduos é:

$$\mathbf{S} - \hat{\mathbf{\Sigma}} = \mathbf{S} - (\hat{\mathbf{L}}\hat{\mathbf{L}}' + \hat{\mathbf{\Psi}})$$

Como $\hat{\mathbf{\Psi}} = \text{diag}(\mathbf{S} - \hat{\mathbf{L}}\hat{\mathbf{L}}')$, os elementos da diagonal da matriz de resíduos são zero. Os resíduos fora da diagonal são os elementos de $\mathbf{S} - \hat{\mathbf{L}}\hat{\mathbf{L}}'$. Pode-se demonstrar que a soma dos quadrados de todos os elementos da matriz $\mathbf{S} - \hat{\mathbf{L}}\hat{\mathbf{L}}'$ (incluindo a diagonal) é:

$$\sum_{i=1}^{p}\sum_{j=1}^{p}(s_{ij} - \sum_{k=1}^{m}\hat{l}_{ik}\hat{l}_{jk})^2 = \sum_{k=m+1}^{p}\hat{\lambda}_k^2$$

Isso mostra que, para que o ajuste seja bom, a soma dos autovalores descartados ($\hat{\lambda}_{m+1}, \ldots, \hat{\lambda}_p$) deve ser pequena.

8.5.2 Método da Máxima Verossimilhança (MMV)

O método da máxima verossimilhança (MMV) é uma abordagem mais rigorosa para a estimação, baseada em suposições sobre a distribuição dos dados.

Suposições Adicionais:

1. O vetor de fatores comuns $\mathbf{F}$ e o vetor de erros $\boldsymbol{\epsilon}$ seguem uma distribuição normal multivariada:
   - $\mathbf{F} \sim N_m(\mathbf{0}, \mathbf{I})$
   - $\boldsymbol{\epsilon} \sim N_p(\mathbf{0}, \mathbf{\Psi})$
2. $\mathbf{F}$ e $\boldsymbol{\epsilon}$ são independentes.

Sob essas condições, o vetor de variáveis observáveis $\mathbf{x}$ segue uma distribuição normal multivariada $N_p(\boldsymbol{\mu}, \mathbf{\Sigma})$, onde $\mathbf{\Sigma} = \mathbf{L}\mathbf{L}' + \mathbf{\Psi}$.

Dada uma amostra aleatória $\mathbf{x}_1, \ldots, \mathbf{x}_n$, a função de log-verossimilhança (ignorando constantes) para os parâmetros $\mathbf{L}$ e $\mathbf{\Psi}$ é:

$$\log L(\mathbf{L}, \mathbf{\Psi}) = -\frac{n}{2}\ln|\mathbf{\Sigma}| - \frac{n}{2}\text{tr}(\mathbf{\Sigma}^{-1}\mathbf{S})$$

onde $\mathbf{S}$ é a matriz de covariâncias amostral (versão ML, com divisor n). O objetivo é encontrar as estimativas $\hat{\mathbf{L}}$ e $\hat{\mathbf{\Psi}}$ que maximizam essa função, sujeito à restrição de que $\hat{\mathbf{L}}'\hat{\mathbf{\Psi}}^{-1}\hat{\mathbf{L}}$ seja uma matriz diagonal para garantir a unicidade da solução.

A maximização é realizada por meio de algoritmos numéricos (como o de Newton-Raphson), pois não há uma solução analítica fechada. As estimativas resultantes, $\hat{\mathbf{L}}$ e $\hat{\mathbf{\Psi}}$, satisfazem um conjunto complexo de equações.

A principal vantagem do MMV é que ele permite um teste de hipóteses para a adequação do número de fatores m, comparando a matriz de covariâncias do modelo, $\hat{\mathbf{\Sigma}} = \hat{\mathbf{L}}\hat{\mathbf{L}}' + \hat{\mathbf{\Psi}}$, com a matriz amostral $\mathbf{S}$. Isso é fundamental na **Análise Fatorial Confirmatória (AFC)**

8.6 A Escolha do Número de Fatores (m)

A determinação do número de fatores, m, é uma das decisões mais importantes na Análise Fatorial. Um número muito baixo de fatores pode não capturar a estrutura de covariância subjacente, enquanto um número muito alto pode levar a um modelo superajustado e de difícil interpretação, violando o princípio da parcimônia.

A escolha de m geralmente envolve uma combinação de critérios estatísticos e julgamento prático. Vários dos métodos utilizados são análogos aos empregados na Análise de Componentes Principais (Capítulo 7). Os mais comuns são:

1. **Proporção da Variância Total Explicada**: Um critério comum é reter fatores suficientes para explicar uma proporção substancial (e.g., 70-90%) da variância total. No contexto do método de componentes principais para AF, a proporção da variância explicada pelo fator j é $\hat{\lambda}_j / \text{tr}(\mathbf{S})$.
2. **Critério de Kaiser (Autovalores > 1)**: Ao trabalhar com a matriz de correlação $\mathbf{R}$, o critério de Kaiser sugere reter apenas os fatores correspondentes a autovalores maiores que 1. A lógica é que um fator deve explicar pelo menos a variância de uma variável original.
3. **Gráfico de cotovelo (*Scree Plot*)**: Este é um gráfico dos autovalores ordenados ($\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \ldots$). Procura-se por um "cotovelo" no gráfico, um ponto onde a magnitude dos autovalores começa a diminuir drasticamente. O número de fatores a reter seria o número de pontos antes do início do platô.
4. **Teste de Hipóteses (para MMV)**: Quando o método da máxima verossimilhança é utilizado, é possível realizar um teste de razão de verossimilhanças para testar a hipótese nula de que m fatores são suficientes para descrever a estrutura de covariância.

Na prática, é recomendável utilizar uma combinação desses critérios. A interpretabilidade da solução fatorial resultante é, em última análise, o guia mais importante.

8.7 Rotação Fatorial

Como visto anteriormente, a solução para a matriz de cargas fatoriais $\hat{\mathbf{L}}$ não é única. Qualquer rotação ortogonal dos fatores resulta em uma nova matriz de cargas $\hat{\mathbf{L}}^* = \hat{\mathbf{L}}\mathbf{T}$ que explica a estrutura de covariâncias dos dados exatamente da mesma forma, pois $\hat{\mathbf{L}}\hat{\mathbf{L}}' = \hat{\mathbf{L}}^*(\hat{\mathbf{L}}^*)'$.

Essa indeterminação, que a princípio parece um problema, é na verdade uma das ferramentas mais poderosas da Análise Fatorial. Ela nos permite girar a estrutura fatorial para uma posição que seja mais **simples e interpretável**, sem sacrificar o ajuste do modelo. O objetivo é alcançar o que o psicólogo Louis Thurstone chamou de **estrutura simples**.

A estrutura simples ideal teria as seguintes propriedades:

1. Cada variável deve ter pelo menos uma carga fatorial próxima de zero.
2. Cada fator deve ter várias cargas próximas de zero e algumas cargas altas.
3. Para cada par de fatores, deve haver variáveis com cargas altas em um fator, mas não no outro.

Em suma, busca-se uma matriz de cargas onde cada variável esteja fortemente associada a apenas um ou poucos fatores, e cada fator represente claramente um subconjunto de variáveis. Os métodos de rotação são algoritmos que buscam, de forma objetiva, uma matriz $\mathbf{T}$ que aproxime a matriz de cargas rotacionada $\hat{\mathbf{L}}^*$ a essa estrutura ideal.

As rotações dividem-se em duas categorias principais.

8.7.1 Rotações Ortogonais

Neste tipo de rotação, a matriz de transformação $\mathbf{T}$ é ortogonal, o que significa que os eixos dos fatores são girados, mas mantidos em um ângulo de 90 graus entre si. A consequência fundamental é que os fatores rotacionados **permanecem não correlacionados**.

Os métodos mais comuns de rotação ortogonal são:

- **Varimax**: É o método de rotação ortogonal mais popular. O objetivo do Varimax é simplificar as **colunas** da matriz de cargas fatoriais. Para cada fator, ele busca maximizar a variância das cargas ao quadrado, efetivamente empurrando as cargas para perto de 0 ou ± 1. Isso facilita a identificação de quais variáveis estão associadas a cada fator. O critério Varimax maximiza a seguinte função:

$$V = \sum_{j=1}^{m} \left[\frac{1}{p} \sum_{i=1}^{p} (\hat{l}_{ij}^*/h_i)^4 - \left(\frac{1}{p} \sum_{i=1}^{p} (\hat{l}_{ij}^*/h_i)^2 \right)^2 \right]$$

  Onde $\hat{l}_{ij}^*$ são as cargas rotacionadas e h_i^2 são as comunalidades (que permanecem invariantes sob rotação).

- **Quartimax**: Este método foca em simplificar as **linhas** da matriz de cargas. Ele tenta fazer com que cada variável tenha carga alta em apenas um fator. O Quartimax foi o primeiro método analítico proposto, mas tende a criar um fator geral com cargas altas para muitas variáveis, o que pode dificultar a interpretação.
- **Equimax**: É um meio termo entre o Varimax e o Quartimax. Ele tenta simplificar tanto as linhas quanto as colunas da matriz de cargas simultaneamente.

8.7.2 Rotações Oblíquas

Em muitos campos, especialmente nas ciências sociais, é teoricamente razoável esperar que os fatores latentes sejam correlacionados. Por exemplo, os fatores "habilidade verbal" e "habilidade matemática" são distintos, mas é provável que sejam positivamente correlacionados.

As rotações oblíquas permitem que os fatores se tornem correlacionados. A matriz de transformação $\mathbf{T}$ não é mais ortogonal, e os eixos dos fatores podem ter ângulos diferentes de 90 graus. A vantagem é a capacidade de encontrar uma estrutura mais simples e teoricamente mais realista, ao custo de uma complexidade maior na interpretação, pois é preciso analisar tanto a matriz de cargas quanto a matriz de correlação entre os fatores.

Os métodos mais comuns incluem:

- **Promax**: É um método muito utilizado que funciona em duas etapas. Primeiro, ele realiza uma rotação ortogonal (geralmente Varimax). Em seguida, ele relaxa a restrição de ortogonalidade, permitindo que os fatores se correlacionem para buscar uma estrutura ainda mais simples (com mais cargas próximas de zero).
- **Oblimin Direto**: É um método mais geral que busca minimizar a covariância das cargas ao quadrado para pares de fatores. Ele possui um parâmetro (delta) que controla o grau de correlação permitido entre os fatores.

A escolha entre uma rotação ortogonal e oblíqua depende de considerações teóricas. Se não há uma razão forte para acreditar que os fatores são correlacionados, a rotação ortogonal (como a Varimax) é geralmente preferida por sua simplicidade. Se a correlação entre os fatores é esperada, uma rotação oblíqua pode fornecer uma representação mais fiel da realidade.

9 Análise de Agrupamentos

A Análise de Agrupamentos, ou Análise de *Clusters*, é uma técnica exploratória multivariada cujo objetivo é particionar um conjunto de observações em subgrupos (os *clusters*). A partição é feita de tal forma que as observações dentro de um mesmo grupo sejam semelhantes entre si, enquanto observações em grupos diferentes sejam o mais distintas possível.

Diferentemente de outras técnicas como a análise de regressão ou a análise discriminante, a análise de agrupamentos é um método de **aprendizagem não supervisionada**. Isso significa que não temos uma variável resposta ou rótulos pré-definidos para os grupos; o objetivo é descobrir a estrutura de agrupamentos inerente aos próprios dados.

O princípio fundamental é a maximização da homogeneidade intra-grupo e, ao mesmo tempo, a maximização da heterogeneidade entre grupos.

Diversas técnicas apresentadas nesse capítulo dependem da definição de uma medida para quantificar o quão semelhantes ou diferentes as observações são. Essa medida é formalizada como uma **medida de dissimilaridade** ou **distância**. Uma discussão detalhada sobre as diferentes métricas de distância pode ser encontrada na Capítulo 6.

Definição 9.1. Dado um conjunto de dados com n observações $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, um **agrupamento** (ou *clustering*) é uma partição de $\mathbf{X}$ em K subconjuntos disjuntos $C_1, C_2, \ldots, C_K$, tal que:

1. $C_k \neq \emptyset$ para $k = 1, \ldots, K$
2. $C_k \cap C_j = \emptyset$ para $k \neq j$
3. $\bigcup_{k=1}^{K} C_k = \mathbf{X}$

Para cada grupo C_k, definimos:

- **Tamanho:** $n_k = \#C_k$, o número de observações no grupo.
- **Centroide:** $\bar{\mathbf{x}}_k = \frac{1}{n_k} \sum_{\mathbf{x}_i \in C_k} \mathbf{x}_i$, o centroide, ou vetor de médias, do grupo.

9.1 Decomposição da Variabilidade

Podemos formalizar o critério de "boa separação" dos grupos através de uma decomposição da variabilidade total dos dados, análoga à Análise de Variância (ANOVA).

Definição 9.2. Dado um conjunto de n observações e uma partição em K grupos $C_1, \ldots, C_K$:

- **Soma de Quadrados Total (SQT):** Mede a dispersão total dos dados em torno da média geral $\bar{\mathbf{x}}$.

$$SQT = \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}})'(\mathbf{x}_i - \bar{\mathbf{x}})$$

- **Soma de Quadrados Intra-grupos (SQI):** Mede a dispersão dentro dos grupos. É a soma das dispersões de cada observação em relação ao centroide do seu próprio grupo, $\bar{\mathbf{x}}_k$. Também é conhecida como *Within-Cluster Sum of Squares* (WCSS).

$$SQI = \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} (\mathbf{x}_i - \bar{\mathbf{x}}_k)'(\mathbf{x}_i - \bar{\mathbf{x}}_k)$$

- **Soma de Quadrados Entre-grupos (SQE):** Mede a dispersão entre os centroides dos grupos em relação à média geral.

$$SQE = \sum_{k=1}^{K} n_k (\bar{\mathbf{x}}_k - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_k - \bar{\mathbf{x}})$$

  Onde n_k é o número de observações no grupo C_k.

O objetivo da análise de agrupamentos pode ser visto como encontrar a partição que **minimiza a SQI** (grupos coesos) e **maximiza a SQE** (grupos separados).

Teorema 9.1. *A soma de quadrados total pode ser decomposta como a soma da variabilidade dentro dos grupos e entre os grupos.*

$$SQT = SQI + SQE$$

Comprovação. A prova parte da decomposição do desvio de uma observação $\mathbf{x}_i \in C_k$ em relação à média geral $\bar{\mathbf{x}}$:

$$(\mathbf{x}_i - \bar{\mathbf{x}}) = (\mathbf{x}_i - \bar{\mathbf{x}}_k) + (\bar{\mathbf{x}}_k - \bar{\mathbf{x}})$$

Elevando ao quadrado (no sentido de produto vetorial), temos:

$$(\mathbf{x}_i - \bar{\mathbf{x}})'(\mathbf{x}_i - \bar{\mathbf{x}}) = [(\mathbf{x}_i - \bar{\mathbf{x}}_k) + (\bar{\mathbf{x}}_k - \bar{\mathbf{x}})]'[(\mathbf{x}_i - \bar{\mathbf{x}}_k) + (\bar{\mathbf{x}}_k - \bar{\mathbf{x}})]$$

$$= (\mathbf{x}_i - \bar{\mathbf{x}}_k)'(\mathbf{x}_i - \bar{\mathbf{x}}_k) + (\bar{\mathbf{x}}_k - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_k - \bar{\mathbf{x}}) + 2(\mathbf{x}_i - \bar{\mathbf{x}}_k)'(\bar{\mathbf{x}}_k - \bar{\mathbf{x}})$$

Agora, somamos sobre todas as observações $i = 1, \ldots, n$. Para fazer isso, somamos primeiro dentro de cada grupo k e depois somamos os resultados sobre todos os grupos:

$$SQT = \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} (\mathbf{x}_i - \bar{\mathbf{x}})'(\mathbf{x}_i - \bar{\mathbf{x}})$$

$$= \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} (\mathbf{x}_i - \bar{\mathbf{x}}_k)'(\mathbf{x}_i - \bar{\mathbf{x}}_k) + \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} (\bar{\mathbf{x}}_k - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_k - \bar{\mathbf{x}}) + \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} 2(\mathbf{x}_i - \bar{\mathbf{x}}_k)'(\bar{\mathbf{x}}_k - \bar{\mathbf{x}})$$

Analisando cada termo:

1. O primeiro termo é, por definição, a Soma de Quadrados Intra-grupos (SQI).
2. No segundo termo, a expressão $(\bar{\mathbf{x}}_k - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_k - \bar{\mathbf{x}})$ é constante para todas as n_k observações no grupo C_k. Portanto, a soma interna resulta em $n_k(\bar{\mathbf{x}}_k - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_k - \bar{\mathbf{x}})$. Somar sobre k nos dá a Soma de Quadrados Entre-grupos (SQE).
3. Para o terceiro termo (o termo cruzado), podemos reescrevê-lo como:

   $$2 \sum_{k=1}^{K} \left[\left(\sum_{\mathbf{x}_i \in C_k} (\mathbf{x}_i - \bar{\mathbf{x}}_k) \right)' (\bar{\mathbf{x}}_k - \bar{\mathbf{x}}) \right]$$

   Pela definição do centroide $\bar{\mathbf{x}}_k$, a soma dos desvios em torno dele dentro de um grupo é zero: $\sum_{\mathbf{x}_i \in C_k} (\mathbf{x}_i - \bar{\mathbf{x}}_k) = \mathbf{0}$. Portanto, todo o terceiro termo é igual a zero.

Juntando os resultados, obtemos $SQT = SQI + SQE$. □

A prova deste teorema mostra que a variabilidade total é conservada e apenas particionada, de forma que minimizar a SQI é equivalente a maximizar a SQE.

9.2 Agrupamentos Hierárquicos

Os métodos de agrupamento hierárquico criam uma sequência de partições aninhadas, que pode ser representada visualmente por uma árvore chamada **dendrograma**. Existem duas abordagens principais:

1. **Aglomerativa (Bottom-Up):** Começa com cada observação em seu próprio grupo e, a cada passo, funde os dois grupos mais próximos até que reste apenas um único grupo contendo todas as observações.
2. **Divisiva (Top-Down):** Começa com todas as observações em um único grupo e, a cada passo, divide um grupo em dois até que cada observação esteja em seu próprio grupo.

> **Nota**
>
> Devido a dificuldades de implementação, agrupamentos divisivos raramente são utilizados na prática. Por esse motivo, os exemplos presentes nesta seção consideram apenas agrupamentos aglomerativos.

9.2.1 Métodos de Ligação (Linkage)

Enquanto as medidas de dissimilaridade retratam a distância entre observações, precisamos de uma regra que define a distância entre dois grupos. Para isso definimos alguns métodos de ligação populares a seguir.

- **Ligação Simples (*Single Linkage*):** A distância entre dois grupos é a distância mínima entre quaisquer dois pontos dos grupos. Tende a produzir grupos "alongados" e é sensível a ruído.

$$d(A, B) = \min_{\mathbf{a} \in A, \mathbf{b} \in B} d(\mathbf{a}, \mathbf{b})$$

- **Ligação Completa (*Complete Linkage*):** A distância é o máximo da distância entre quaisquer dois pontos. Produz grupos mais compactos e esféricos.

$$d(A, B) = \max_{\mathbf{a} \in A, \mathbf{b} \in B} d(\mathbf{a}, \mathbf{b})$$

- **Ligação Média (*Average Linkage*):** A distância é a média de todas as distâncias entre os pares de pontos dos dois grupos. É um meio-termo entre a simples e a completa.

$$d(A, B) = \frac{1}{n_A n_B} \sum_{\mathbf{a} \in A} \sum_{\mathbf{b} \in B} d(\mathbf{a}, \mathbf{b})$$

- **Método de Ward:** Este método se baseia em um critério de minimização da variância. A cada passo do algoritmo aglomerativo, ele funde o par de grupos que leva ao **menor aumento possível na Soma de Quadrados Intra-grupos (SQI)**. O objetivo é encontrar, a cada passo, a fusão mais "econômica" em termos de perda de coesão interna.

  Suponha que estejamos considerando fundir dois grupos, C_i e C_j. O aumento na SQI, que denotamos por $\Delta(C_i, C_j)$, é a diferença entre a SQI do novo grupo fundido (C_{ij}) e a soma das SQIs dos grupos individuais antes da fusão.

$$\Delta(C_i, C_j) = \text{SQI}(C_{ij}) - (\text{SQI}(C_i) + \text{SQI}(C_j))$$

A SQI para o novo grupo $C_{ij} = C_i \cup C_j$ é $\sum_{\mathbf{x} \in C_{ij}} (\mathbf{x} - \bar{\mathbf{x}}_{ij})'(\mathbf{x} - \bar{\mathbf{x}}_{ij})$, onde $\mathbf{x}_{ij}$ é o centroide do novo grupo. Podemos reescrever essa soma como:

$$\sum_{\mathbf{x} \in C_i} (\mathbf{x} - \bar{\mathbf{x}}_{ij})'(\mathbf{x} - \bar{\mathbf{x}}_{ij}) + \sum_{\mathbf{x} \in C_j} (\mathbf{x} - \bar{\mathbf{x}}_{ij})'(\mathbf{x} - \bar{\mathbf{x}}_{ij})$$

Usando a mesma lógica da decomposição da variância, podemos mostrar que $\sum_{\mathbf{x} \in C_i} (\mathbf{x} - \bar{\mathbf{x}}_{ij})'(\mathbf{x} - \bar{\mathbf{x}}_{ij}) = \text{SQI}(C_i) + n_i(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_{ij})'(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_{ij})$.

Aplicando o resultado para ambos os termos, a SQI do novo grupo é:

$$\text{SQI}(C_{ij}) = \text{SQI}(C_i) + \text{SQI}(C_j) + n_i(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_{ij})'(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_{ij}) + n_j(\bar{\mathbf{x}}_j - \bar{\mathbf{x}}_{ij})'(\bar{\mathbf{x}}_j - \bar{\mathbf{x}}_{ij})$$

Portanto, o aumento na SQI é:

$$\Delta(C_i, C_j) = n_i(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_{ij})'(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_{ij}) + n_j(\bar{\mathbf{x}}_j - \bar{\mathbf{x}}_{ij})'(\bar{\mathbf{x}}_j - \bar{\mathbf{x}}_{ij})$$

Substituindo $\bar{\mathbf{x}}_{ij} = \frac{n_i \bar{\mathbf{x}}_i + n_j \bar{\mathbf{x}}_j}{n_i + n_j}$ e simplificando a álgebra, chegamos a:

$$\Delta(C_i, C_j) = \frac{n_i n_j}{n_i + n_j} (\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j)'(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j)$$

A fórmula final nos dá uma maneira eficiente de calcular o critério de Ward. A cada passo, o algoritmo calcula $\Delta(C_i, C_j)$ para todos os pares de grupos e realiza a fusão para o par que tiver o menor valor. Note que a fórmula depende da distância euclidiana entre centroides, por esse motivo, o método de Ward tende a produzir grupos de tamanho semelhante e formato esférico.

9.2.2 O Dendrograma

O dendrograma é a principal ferramenta de visualização para o agrupamento hierárquico. Ele mostra como, passo a passo, as observações são fundidas em grupos. O eixo Y representa a distância ou dissimilaridade em que as fusões ocorrem. Quanto mais alta a "ponte" que une dois grupos, mais diferentes eles são.

O gráfico abaixo, gerado a partir de um exemplo simples de 4 observações, mostra a anatomia de um dendrograma.

O exemplo abaixo apresenta os dendrogramas de maneira prática, além de exemplificar também como a função de ligação escolhida impacta no agrupamento formado.

Exemplo 9.1. Vamos analisar o comportamento dos métodos de ligação com a matriz de 5x5 abaixo.

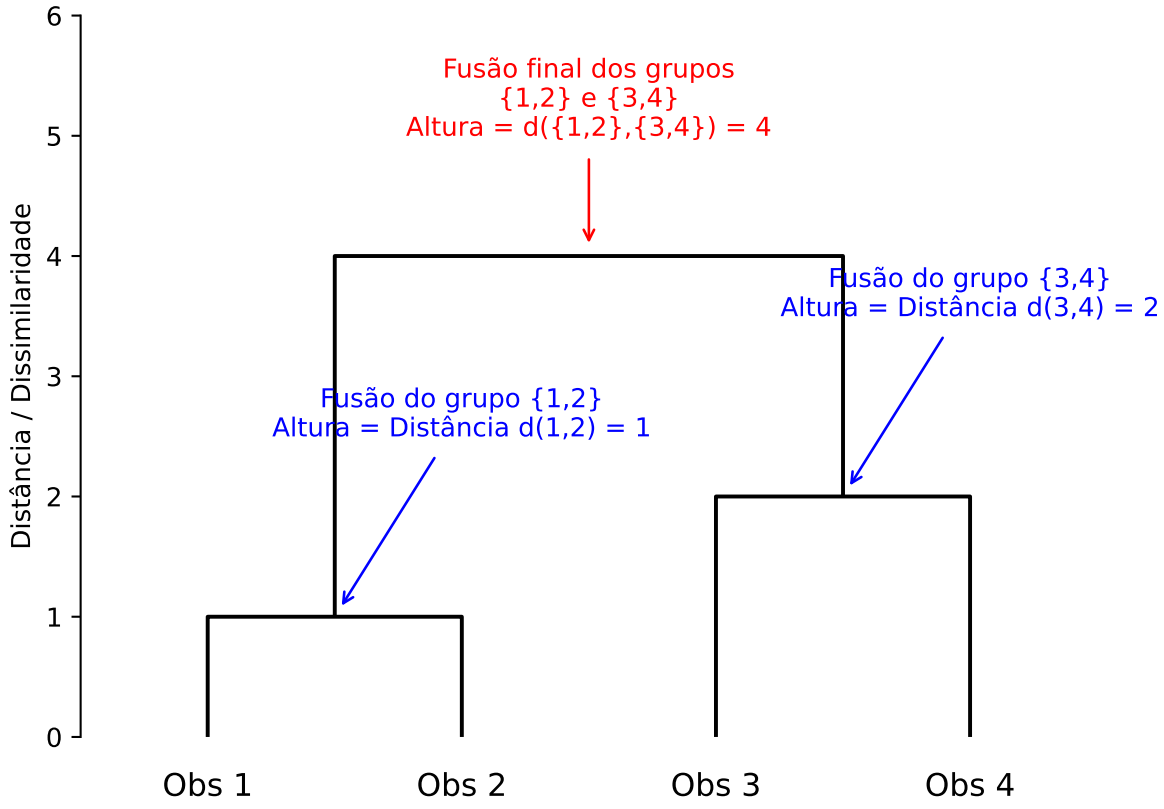

Figura 9.1: Anatomia de um dendrograma simples.

$$\mathbf{D} = \left(\begin{array}{c|ccccc} & \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} \\ \hline \mathbf{1} & 0 & & & & \\ \mathbf{2} & 1 & 0 & & & \\ \mathbf{3} & 8 & 2 & 0 & & \\ \mathbf{4} & 6 & 4 & 11 & 0 & \\ \mathbf{5} & 7 & 9 & 8 & 7 & 0 \end{array}\right)$$

1. Usando a Ligação Simples (*Single Linkage*)

- **Passo 1:** A menor distância na matriz é `d(1,2) = 1`. Fundimos `{1,2}` na altura 1.
- **Passo 2:** A distância do novo grupo `{1,2}` para `{3}` é `min(d(1,3), d(2,3)) = min(8, 2) = 2`. Todas as outras distâncias entre grupos ou pontos restantes são maiores que 2. Assim, fundimos `{1,2}` com `{3}` na altura 2.
- **Passo 3:** A distância de `{1,2,3}` para `{4}` é `min(d(1,4), d(2,4), d(3,4)) = min(6, 4, 11) = 4`. Esta é a próxima menor distância, então fundimos `{1,2,3}` com `{4}` na altura 4.
- **Passo 4:** A distância de `{1,2,3,4}` para `{5}` é `min(d(1,5), d(2,5), d(3,5), d(4,5)) = min(7, 9, 8, 7) = 7`. Fundimos o último ponto na altura 7.
- **Resultado:** O método cria uma longa cadeia: `(((({1,2},3),4),5)`.

2. Usando a Ligação Completa (*Complete Linkage*)

- **Passo 1:** A fusão inicial é a mesma: `{1,2}` (altura 1).
- **Passo 2:** A distância de `{1,2}` para `{4}` é `max(d(1,4), d(2,4)) = max(6, 4) = 6`. Já a distância para `{3}` é `max(d(1,3), d(2,3)) = max(8, 2) = 8`. A menor distância entre os grupos existentes é 6, então fundimos `{1,2}` com `{4}`.
- **Passo 3:** Temos os grupos `{1,2,4}`, `{3}` e `{5}`. A próxima menor distância entre os grupos restantes é `d(3,5) = 8`. Fundimos `{3,5}`.
- **Passo 4:** A distância entre `{1,2,4}` e `{3,5}` é `max(d(1,3), d(1,5), d(2,3), d(2,5), d(4,3), d(4,5)) = max(8, 7, 2, 9, 11, 7) = 11`. A fusão final ocorre na altura 11.
- **Resultado:** O método cria dois grupos distintos, `({1,2,4}, {3,5})`, antes da fusão final.

Os resultados são estruturalmente diferentes, como mostram os dendrogramas.

```
import numpy as np
from scipy.cluster.hierarchy import dendrogram, linkage
from scipy.spatial.distance import squareform
import matplotlib.pyplot as plt

# Matriz de distância 5x5 do exemplo
D = np.array([
    [0, 1, 8, 6, 7],
    [1, 0, 2, 4, 9],
    [8, 2, 0, 11, 8],
```

```
    [6, 4, 11, 0, 7],
    [7, 9, 8, 7, 0]
])

condensed_D = squareform(D)
labels = ['1', '2', '3', '4', '5']

fig, axes = plt.subplots(1, 2, figsize=(7, 5))
fig.suptitle('Dendrogramas Puros (Sem Definição de grupos)')

# Ligação Simples
linked_single = linkage(condensed_D, 'single')
dendrogram(linked_single, orientation='top', labels=labels, ax=axes[0], color_threshold=0
axes[0].set_title('Ligação Simples')
axes[0].set_xlabel('Observação')
axes[0].set_ylabel('Distância')

# Ligação Completa
linked_complete = linkage(condensed_D, 'complete')
dendrogram(linked_complete, orientation='top', labels=labels, ax=axes[1], color_threshold
axes[1].set_title('Ligação Completa')
axes[1].set_xlabel('Observação')

plt.tight_layout(rect=[0, 0.03, 1, 0.95])
plt.show()
```

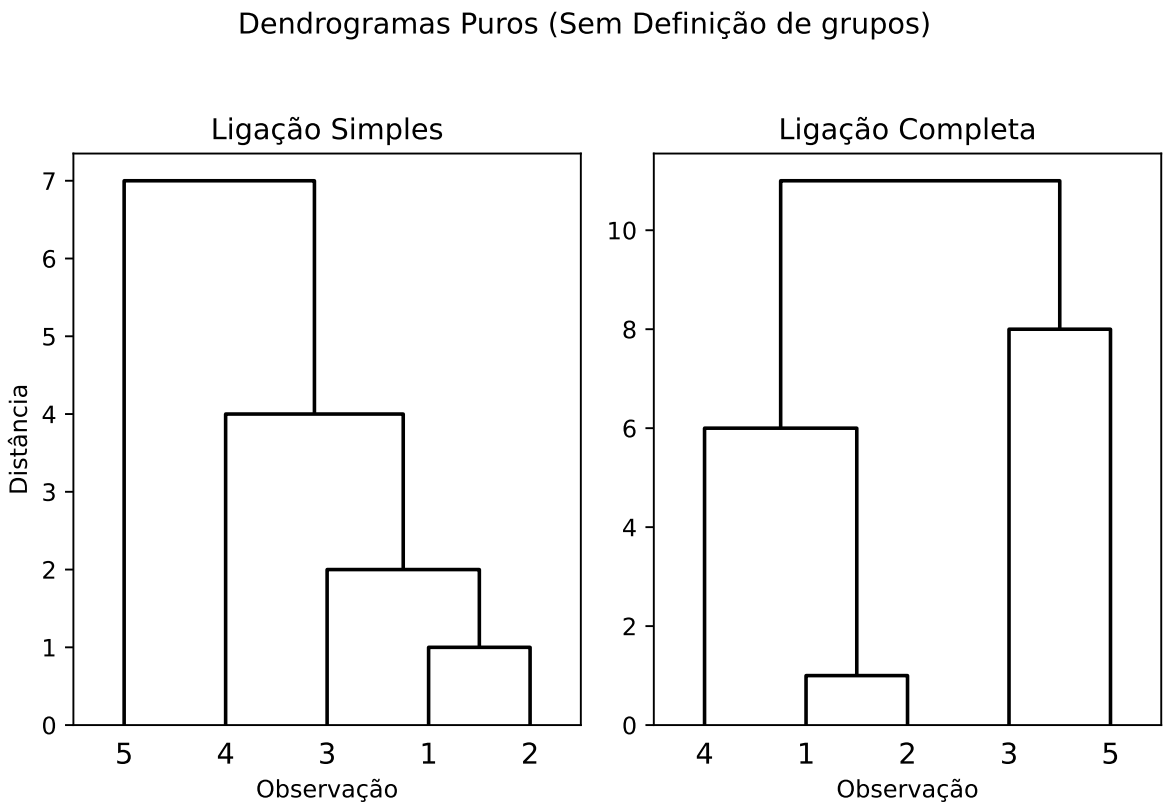

Figura 9.2: Comparação de dendrogramas para a matriz 5x5. As cores foram removidas para mostrar a estrutura pura.

O dendrograma não apenas mostra a hierarquia das fusões, mas também é a principal ferramenta para decidir o número final de grupos. A estratégia consiste em "cortar" a árvore em uma determinada altura. Todas as ramificações que estão abaixo da linha de corte constituem os grupos.

A regra geral é procurar por um corte que cruze as conexões mais longas. Uma conexão longa representa uma fusão que ocorreu a uma distância (ou aumento de SQI, no caso de Ward) muito maior do que as fusões anteriores. Isso sugere que estamos unindo grupos que são naturalmente muito diferentes entre si. Portanto, cortar a árvore logo acima dessa grande "distância de fusão" é uma escolha sensata.

A Figura 9.3 demonstra como a escolha de diferentes alturas de corte leva a diferentes números de grupos.

```
import numpy as np
from scipy.cluster.hierarchy import dendrogram, linkage
from scipy.spatial.distance import squareform
```

```
import matplotlib.pyplot as plt

# Matriz de distância 5x5 do exemplo
D = np.array([
    [0, 1, 8, 6, 7],
    [1, 0, 2, 4, 9],
    [8, 2, 0, 11, 8],
    [6, 4, 11, 0, 7],
    [7, 9, 8, 7, 0]
])

condensed_D = squareform(D)
labels = ['1', '2', '3', '4', '5']
linked_complete = linkage(condensed_D, 'complete')

fig, axes = plt.subplots(1, 3, figsize=(7, 5))
fig.suptitle('Visualização dos Cortes no Dendrograma (Ligação Completa)')

# --- Corte para K=2 ---
cut_height_k2 = 9
dendrogram(
    linked_complete,
    orientation='top',
    labels=labels,
    ax=axes[0],
    color_threshold=0,
    above_threshold_color='k'
)
axes[0].axhline(y=cut_height_k2, color='r', linestyle='--')
axes[0].set_title('Corte para K=2 grupos')
axes[0].set_xlabel('Observação')
axes[0].set_ylabel('Distância')

# --- Corte para K=3 ---
cut_height_k3 = 7
dendrogram(
    linked_complete,
    orientation='top',
    labels=labels,
    ax=axes[1],
    color_threshold=0,
    above_threshold_color='k'
```

```
)
axes[1].axhline(y=cut_height_k3, color='r', linestyle='--')
axes[1].set_title('Corte para K=3 grupos')
axes[1].set_xlabel('Observação')

# --- Corte para K=4 ---
cut_height_k4 = 3
dendrogram(
    linked_complete,
    orientation='top',
    labels=labels,
    ax=axes[2],
    color_threshold=0,
    above_threshold_color='k'
)
axes[2].axhline(y=cut_height_k4, color='r', linestyle='--')
axes[2].set_title('Corte para K=4 grupos')
axes[2].set_xlabel('Observação')

plt.tight_layout(rect=[0, 0.03, 1, 0.95])
plt.show()
```

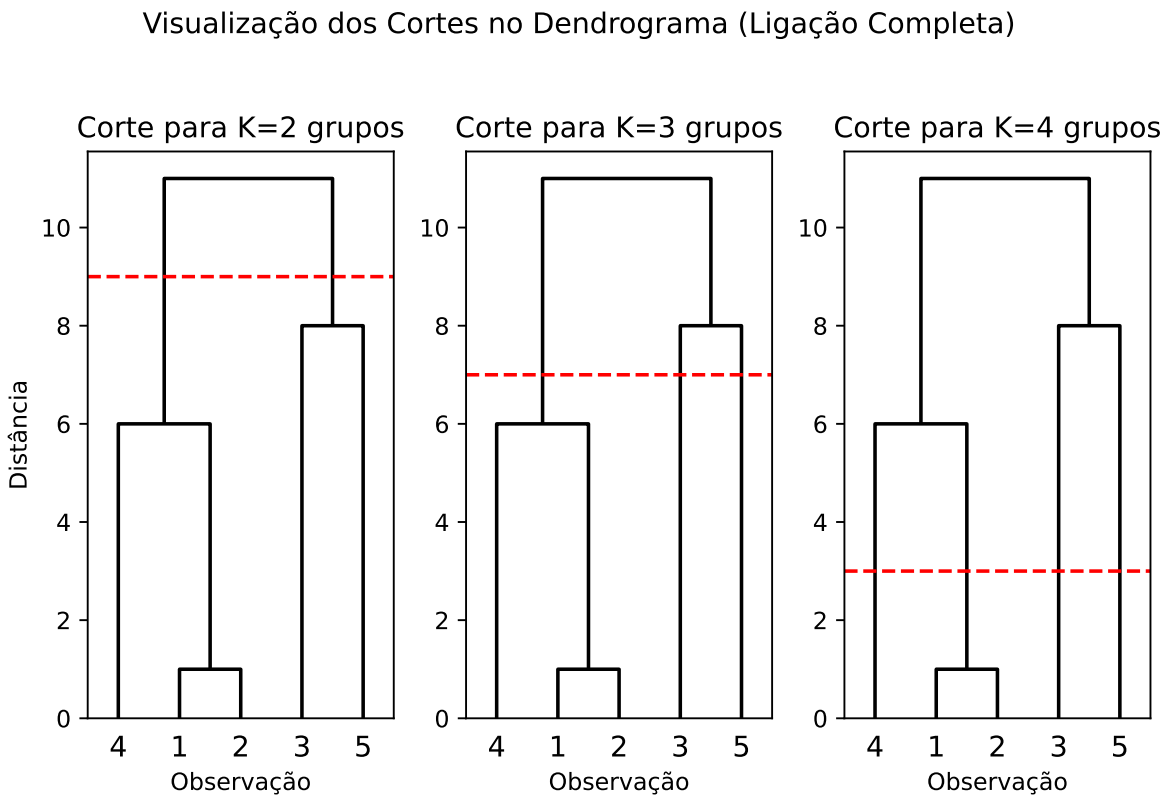

Figura 9.3: Ilustração do corte no dendrograma de Ligação Completa para obter K=2 (esquerda), K=3 (centro) e K=4 (direita) grupos.

A interpretação dos resultados para cada corte é a seguinte:

- **K=2:** Ao cortar o dendrograma na altura 9, obtemos dois grupos. O primeiro é `{1, 2, 4}` e o segundo é `{3, 5}`. Esta partição representa a estrutura de mais alto nível nos dados, separando os dois grupos mais distintos.
- **K=3:** Abaixando a linha de corte para a altura 7, o grupo `{3, 5}` (que era formado na altura 8) é quebrado. O resultado são três grupos: `{1, 2, 4}`, `{3}` e `{5}`.
- **K=4:** Com um corte ainda mais baixo, na altura 3, quebramos o grupo `{1, 2, 4}` (formado na altura 6). Os grupos resultantes são `{1, 2}`, `{4}`, `{3}` e `{5}`.

Observe que o primeiro grupo `{1, 2}` se forma em uma altura de uma unidade de distância. Já a segunda ligação, `{1, 2}` com `{4}` ocorre na altura `d({1, 2}, {4})=6`. Isso indica que já existe um salto na distância da ligação logo no segundo passo. Por isso, uma escolha sensível é $K = 4$.

> **O Eixo Vertical no Dendrograma**
>
> Para os métodos de ligação simples, completa e média o eixo vertical do dendrograma represente diretamente a distância da fusão. Já para o **Método de Ward**, a altura da fusão representa o **aumento na Soma de Quadrados Intra-grupos (SQI)** resultante da união dos dois grupos. Esse valor não é uma distância, mas sim uma medida de perda de homogeneidade.

9.2.3 Limitações do Agrupamento Hierárquico

Apesar de sua simplicidade e elegância, os métodos hierárquicos possuem limitações importantes. Uma delas é sua complexidade computacional, que cresce rapidamente para dados com muitas observações devido a grande quantidade de comparações.

Além disso, a decisão de fusão é final e não pode ser desfeita. Se um grupo inicial for inadequado, o erro se propagará por toda a hierarquia. Ou seja, o agrupamento é muito sensível à estrutura inicial. Finalmente, todos os métodos de ligação carregam suas vantagens e problemas.

Essas limitações motivam o uso de métodos não hierárquicos, como o K-médias, que abordaremos a seguir.

9.3 Agrupamento Não-Hierárquico

Diferente dos métodos hierárquicos, os métodos particionais, como o K-Médias, dividem os dados em um número K de grupos pré-especificado. O K-Médias é um dos algoritmos de agrupamento mais populares e eficientes.

O objetivo do K-Médias é particionar as n observações em K grupos de modo a minimizar a Soma de Quadrados Intra-grupos (SQI), também chamada de inércia.

$$\text{SQI} = \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} (\mathbf{x}_i - \bar{\mathbf{x}}_k)'(\mathbf{x}_i - \bar{\mathbf{x}}_k)$$

Onde $\bar{\mathbf{x}}_k$ é o centroide (média) do grupo C_k.

9.3.1 O Algoritmo K-Médias (K-Means)

O algoritmo K-médias é um processo iterativo que busca minimizar a SQI. Seus passos podem ser definidos matematicamente da seguinte forma:

1. **Inicialização:** Escolha K centroides iniciais $\boldsymbol{\mu}_1^{(0)}, \boldsymbol{\mu}_2^{(0)}, \ldots, \boldsymbol{\mu}_K^{(0)}$. Esta escolha pode ser feita selecionando K observações aleatórias do conjunto de dados.

2. **Atribuição:** Em cada iteração t, atribua cada observação $\mathbf{x}_i$ ao grupo cujo centroide é o mais próximo. Matematicamente, para cada observação i, encontramos o índice do grupo c_i que minimiza a distância Euclidiana quadrada:

$$c_i^{(t)} = \arg\min_k (\mathbf{x}_i - \boldsymbol{\mu}_k^{(t-1)})'(\mathbf{x}_i - \boldsymbol{\mu}_k^{(t-1)})$$

   Isso particiona os dados nos conjuntos $C_1^{(t)}, \ldots, C_K^{(t)}$.

3. **Atualização:** Recalcule o centroide de cada grupo como a média de todas as observações atribuídas a ele na iteração atual:

$$\boldsymbol{\mu}_k^{(t)} = \frac{1}{\#C_k^{(t)}} \sum_{\mathbf{x}_i \in C_k^{(t)}} \mathbf{x}_i$$

4. **Repetição:** Repita os passos 2 e 3 até que as atribuições dos grupos não mudem mais, ou seja, $c_i^{(t)} = c_i^{(t-1)}$ para todas as observações i.

O grande problema do método K-médias é a escolha dos centroides iniciais. O algoritmo tem a garantia de convergir, mas dependendo das condições iniciais pode chegar a um mínimo local, e não necessariamente o mínimo global da SQI. Uma opção comum é executar o algoritmo várias vezes com diferentes inicializações e escolher o resultado com a menor SQI.

A necessidade de pré-especificar o número de grupos, K, também pode ser uma desvantagem. Para contornar o problema, é comum executar o algoritmo para uma gama de valores de K e calcular a Soma de Quadrados Intra-grupos (SQI) em cada caso. A ideia é escolher K tal que a SQI seja baixa, mas sendo parcimonioso com o número de grupos. Vale lembrar que a SQI é inversamente proporcional a K – se $K = 1$, a SQI é máxima e se $K = n$ a SQI é zero. Na prática, aumentamos K progressivamente observando o decréscimo na SQI, seguimos aumentando K enquanto esse decréscimo for grande.

A seguir, definimos um procedimento mais robusto, que soluciona os problemas mencionados combinando diferentes métodos de agrupamento.

9.3.2 Procedimento sugerido para análise de agrupamentos

Levando em consideração as vantagens e desvantagens dos agrupamentos hierárquicos e de K-médias, podemos definir um procedimento que simplifica as escolhas durante um problema prático de análise de agrupamentos.

- Inicie com um agrupamento hierárquico. O método de Ward costuma ser uma boa primeira opção, no entanto, experimente também outros métodos de ligação conforme necessário.

- Observe o dendrograma para escolher o corte que seja mais plausível. Esse corte determina um número de grupos ótimo K^*.
- Calcule o centroide para cada um dos grupos obtidos via agrupamento hierárquico $\mu_1, \cdots, \mu_K$.
- Utilize os centroides obtidos como centroides iniciais para o método K-médias com K^* grupos.

Exemplificamos esse procedimento com dados reais no Capítulo 15.

9.4 Agrupamento Baseado em Modelos

Uma abordagem mais avançada e flexível é o agrupamento baseado em modelos. A ideia central é assumir que os dados são gerados a partir de uma **mistura de distribuições de probabilidade**, onde cada componente da mistura corresponde a um grupo.

O modelo mais comum é o **Modelo de Mistura Gaussiana (*Gaussian Mixture Model*, GMM)**. Ele assume que cada grupo segue uma distribuição normal multivariada. A densidade de probabilidade de todo o conjunto de dados é uma soma ponderada de K densidades Gaussianas:

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Onde, para cada grupo k:

- π_k: é o peso da mistura. Por ser uma probabilidade, deve satisfazer $\pi_k \geq 0$ e $\sum_{k=1}^{K} \pi_k = 1$.
- $\boldsymbol{\mu}_k$: é o vetor de médias (o centroide do grupo).
- $\boldsymbol{\Sigma}_k$: é a matriz de covariâncias (descreve a forma e orientação do grupo).

9.4.1 O Algoritmo de Expectation-Maximization (EM)

Os parâmetros do GMM $(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ são estimados usando o algoritmo de **Expectation-Maximization (EM)**, um processo iterativo que alterna entre dois passos.

1. **Inicialização:** Inicialize os parâmetros do modelo $\pi_k^{(0)}, \boldsymbol{\mu}_k^{(0)}, \boldsymbol{\Sigma}_k^{(0)}$ para cada grupo $k = 1, \ldots, K$. Isso pode ser feito de forma aleatória ou usando o resultado de um algoritmo mais simples, como o K-médias.
2. **Passo-E (Expectation):** Em cada iteração t, calculamos a "responsabilidade" de cada grupo k por cada observação $\mathbf{x}_i$. A responsabilidade, denotada por $p_{ik}^{(t)}$, é a probabilidade posterior de que a observação $\mathbf{x}_i$ tenha sido gerada pela componente k, dados os parâmetros da iteração anterior. Usando a regra de Bayes, a responsabilidade é calculada como:

$$p_{ik}^{(t)} = \frac{\pi_k^{(t-1)} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k^{(t-1)}, \boldsymbol{\Sigma}_k^{(t-1)})}{\sum_{j=1}^{K} \pi_j^{(t-1)} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_j^{(t-1)}, \boldsymbol{\Sigma}_j^{(t-1)})}$$

Essencialmente, para cada ponto, calculamos a probabilidade de ele pertencer a cada um dos K grupos.

3. **Passo-M (Maximization):** Nesta etapa, usamos as responsabilidades $p_{ik}^{(t)}$ para reestimar os parâmetros do modelo, maximizando a verossimilhança esperada. As atualizações para a iteração t são:

   - **Pesos da mistura:** O peso $\pi_k^{(t)}$ é a proporção média de responsabilidade do grupo k sobre todas as observações.

   $$\pi_k^{(t)} = \frac{N_k^{(t)}}{n}, \quad \text{onde } N_k^{(t)} = \sum_{i=1}^{n} p_{ik}^{(t)}$$

   - **Médias:** A média $\boldsymbol{\mu}_k^{(t)}$ é uma média ponderada de todas as observações, onde os pesos são as responsabilidades.

   $$\boldsymbol{\mu}_k^{(t)} = \frac{1}{N_k^{(t)}} \sum_{i=1}^{n} p_{ik}^{(t)} \mathbf{x}_i$$

   - **Covariâncias:** A matriz de covariâncias $\boldsymbol{\Sigma}_k^{(t)}$ é uma média ponderada das covariâncias, centrada na nova média.

   $$\boldsymbol{\Sigma}_k^{(t)} = \frac{1}{N_k^{(t)}} \sum_{i=1}^{n} p_{ik}^{(t)} (\mathbf{x}_i - \boldsymbol{\mu}_k^{(t)})(\mathbf{x}_i - \boldsymbol{\mu}_k^{(t)})'$$

4. **Repetição:** Os passos E e M são repetidos até que a log-verossimilhança do modelo, $\ln p(\mathbf{X}|\pi, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{n} \ln \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right)$, convirja para um valor estável.

Exemplificamos a aplicação do modelo de mistura Gaussiana para análise de agrupamentos na Capítulo 16.

9.4.2 Vantagens do Agrupamento Baseado em Modelos

- **Agrupamento "Suave":** O GMM fornece uma probabilidade de pertencimento a cada grupo para cada observação, em vez de uma atribuição "dura" (tudo ou nada) como o K-Médias.
- **Flexibilidade de Formato:** Ao permitir que cada grupo tenha sua própria matriz de covariâncias $\boldsymbol{\Sigma}_k$, os GMMs podem identificar grupos com diferentes formas (elípticas) e orientações, enquanto o K-Médias assume implicitamente que os grupos são esféricos.
- **Seleção de Modelo Criteriosa:** A natureza estatística do método permite o uso de critérios de informação, como o **Critério de Informação Bayesiano (BIC)** ou o **Critério de Informação de Akaike (AIC)**, para selecionar o número ideal de grupos K e a estrutura da matriz de covariâncias de forma mais objetiva. O modelo com o menor valor de BIC é geralmente preferido.

9.5 Índices de validação de agrupamentos

Uma das perguntas mais importantes na análise de grupo é "qual o número ideal de grupos?". Embora o procedimento que combina o método hierárquico com o K-médias ajude a guiar essa escolha, existem diversas métricas quantitativas que avaliam a qualidade de uma partição de grupos, independentemente do método utilizado para obtê-la. Esses índices nos permitem comparar os resultados para diferentes valores de K e escolher o que for estatisticamente mais robusto.

9.5.1 Método da Silhueta

A análise de silhueta é uma das técnicas mais populares. Ela mede o quão bem cada observação se encaixa em seu grupo em comparação com outros grupos. O coeficiente de silhueta para uma única observação i é:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

Onde:

- $a(i)$: A distância média de i para todas as outras observações no *mesmo* grupo (coesão).
- $b(i)$: A distância média de i para todas as observações no grupo *vizinho mais próximo* (separação).

O valor de $s(i)$ varia de -1 a 1. Para um dado K, calculamos o coeficiente de silhueta médio para todas as observações. O valor de K que **maximiza** a pontuação média da silhueta é considerado o ideal.

9.5.2 Índice de Calinski-Harabasz

Também conhecido como Critério da Razão de Variâncias, este índice avalia a qualidade do agrupamento pela razão entre a dispersão entre os grupos e a dispersão intra-grupo. A fórmula é:

$$CH(K) = \frac{SQE/(K-1)}{SQI/(n-K)}$$

Onde SQE é a Soma de Quadrados Entre-grupos, SQI é a Soma de Quadrados Intra-grupos, n é o número total de observações e K é o número de grupos.

Intuitivamente, um bom agrupamento tem uma alta dispersão entre os grupos (SQE alta) e uma baixa dispersão dentro dos grupos (SQI baixa). Portanto, procuramos o valor de K que **maximiza** o índice de Calinski-Harabasz.

9.5.3 Índice de Davies-Bouldin

Este índice mede a "similaridade" média de cada grupo com seu grupo mais semelhante. A similaridade entre dois grupos C_i e C_j é definida como:

$$R_{ij} = \frac{s_i + s_j}{d_{ij}}$$

Onde s_i é a dispersão média dentro do grupo i (por exemplo, a distância média de cada ponto ao centroide) e d_{ij} é a distância entre os centroides dos grupos i e j.

Para cada grupo i, encontramos o grupo j que maximiza R_{ij}. O índice de Davies-Bouldin é a média desses valores máximos sobre todos os grupos:

$$DB(K) = \frac{1}{K} \sum_{i=1}^{K} \max_{j \neq i}(R_{ij})$$

Um valor baixo de $DB(K)$ indica que os grupos estão bem separados e compactos. Portanto, procuramos o valor de K que **minimiza** o índice de Davies-Bouldin.

Em resumo, a escolha de K deve ser guiada por uma combinação dessas ferramentas, a análise do dendrograma e, mais importante, o contexto do problema. Se os grupos resultantes não forem interpretáveis ou úteis, o valor de K deve ser reconsiderado.

9.6 Interpretações em análises de agrupamentos

Uma vez que os grupos são formados, o passo final e mais importante é a **interpretação**. Um agrupamento matematicamente sólido é inútil se não pudermos extrair dele insights práticos. A interpretação envolve a caracterização e a validação dos grupos.

9.6.1 Perfil dos grupos

O objetivo aqui é responder à pergunta: "Quem são os membros de cada grupo?". Para isso, descrevemos cada grupo em termos das variáveis usadas na análise (e também de outras variáveis descritivas, se disponíveis). O método mais comum é calcular estatísticas descritivas para cada variável, segmentadas por grupo.

- **Variáveis Contínuas:** Calcule a média e o desvio padrão de cada variável para cada grupo. Em seguida, compare a média de um grupo com a média geral da amostra. Por exemplo: "O grupo 2 tem uma renda média 30% acima da média geral, enquanto o grupo 1 tem uma renda 20% abaixo".

- **Variáveis Categóricas:** Calcule a distribuição de frequência (ou porcentagens) de cada categoria para cada grupo. Por exemplo: "O grupo 3 é composto por 80% de clientes do sexo feminino, enquanto na amostra total essa proporção é de 55%".

A criação de uma tabela de perfis, com os grupos nas linhas e as estatísticas das variáveis nas colunas, é uma ferramenta extremamente eficaz para resumir e comunicar os resultados.

Depois de entender o perfil de cada grupo, é uma boa prática dar a eles nomes descritivos. Nomes como "Jovens Urbanos Conectados", "Famílias Rurais Conservadoras" ou "Clientes de Alto Risco" são muito mais fáceis de comunicar e lembrar do que "Grupo 1", "Grupo 2", etc. O nome deve capturar a essência do que torna aquele grupo único.

A interpretação é um processo iterativo. Às vezes, os perfis dos grupos podem sugerir que o número de grupos escolhido não foi o ideal, levando o analista a revisitar as etapas anteriores.

9.6.2 Visualização do agrupamento

A visualização é pode auxiliar na validação da separação dos grupos.

- **Dados de Baixa Dimensão (2D ou 3D):** Um simples gráfico de dispersão, com os pontos identificados de acordo com seu grupo designado, é suficiente.
- **Dados de Alta Dimensão:** Quando temos mais de três variáveis, precisamos primeiro reduzir a dimensionalidade para poder visualizar. A **Análise de Componentes Principais (ACP)** é a técnica mais comum para isso. Podemos plotar as observações em um gráfico de dispersão usando os dois primeiros componentes principais como eixos e identificar os pontos por grupo. Isso nos dá uma visão da separação dos grupos no espaço que captura a maior parte da variabilidade dos dados.

10 Análise Discriminante

A Análise Discriminante (AD) é uma técnica estatística multivariada utilizada para descrever e classificar observações em grupos pré-definidos. A partir de um conjunto de variáveis preditoras, a AD busca encontrar combinações lineares dessas variáveis — chamadas de **funções discriminantes** — que maximizem a separação entre os grupos.

Em contraste com a Análise de Agrupamentos (Capítulo 9) — uma técnica de **aprendizagem não supervisionada** que visa *descobrir* grupos em um conjunto de dados —, a Análise Discriminante é uma técnica de **aprendizagem supervisionada**. Isso significa que partimos de uma classificação de grupos já conhecida e o objetivo é encontrar uma regra que melhor separe esses grupos, permitindo-nos, assim, classificar novas observações.

A AD possui duas finalidades principais:

1. **Separação (Análise exploratória):** Encontrar as dimensões (funções discriminantes) que melhor revelam as diferenças entre as populações, e interpretar tais funções, a fim de entender quais características levam à distinção dos grupos.
2. **Classificação (Análise preditiva):** Desenvolver uma regra para alocar novas observações, cujas afiliações de grupo são desconhecidas, a um dos grupos existentes.

10.1 Regras de discriminação para dois grupos

Para alocar os indivíduos em seus grupos mais prováveis, devemos definir uma regra de discriminação ou classificação. Esse é motivado probabilisticamente e tem como objetivo minimizar o custo de alocação/classificação incorreta, i.e., o erro em afirmar que um objeto pertence um grupo, quando na verdade pertence a outro.

Formalmente, considere um vetor p-dimensional de variáveis aleatórias $\mathbf{X}$, e duas populações distintas π_1 e π_2 com funções densidade de probabilidade $f_1(\mathbf{x})$ e $f_2(\mathbf{x})$, respectivamente.

Seja Ω o espaço amostral de $\mathbf{X}$. A criação de uma regra de discriminação pode ser formulada como a tarefa de particionar $\Omega = R_1 \cup R_2$, onde R_1 e R_2 são as partições do espaço nas quais os objetos ali pertencentes são alocados em π_1 e π_2, respectivamente.

Denotamos como p_1 e p_2 as probabilidades *a priori* de π_1 e π_2, respectivamente. Esses valores podem ser por exemplo, a proporção populacional de cada um dos grupos. As probabilidades de alocação correta e incorreta das observações são,

- **Corretamente alocado em** π_1: $P(X \in R_1|\pi_1)P(\pi_1) = P(1|1)p_1$
- **Incorretamente alocado em** π_1: $P(X \in R_1|\pi_2)P(\pi_2) = P(1|2)p_2$
- **Corretamente alocado em** π_2: $P(X \in R_2|\pi_2)P(\pi_2) = P(2|2)p_2$
- **Incorretamente alocado em** π_2: $P(X \in R_2|\pi_1)P(\pi_1) = P(2|1)p_1$

Note que existem dois tipos distintos de erros de alocação, probabilisticamente definidos por $P(1|2)$ e $P(2|1)$, ilustrados esquematicamente na Figura 10.1.

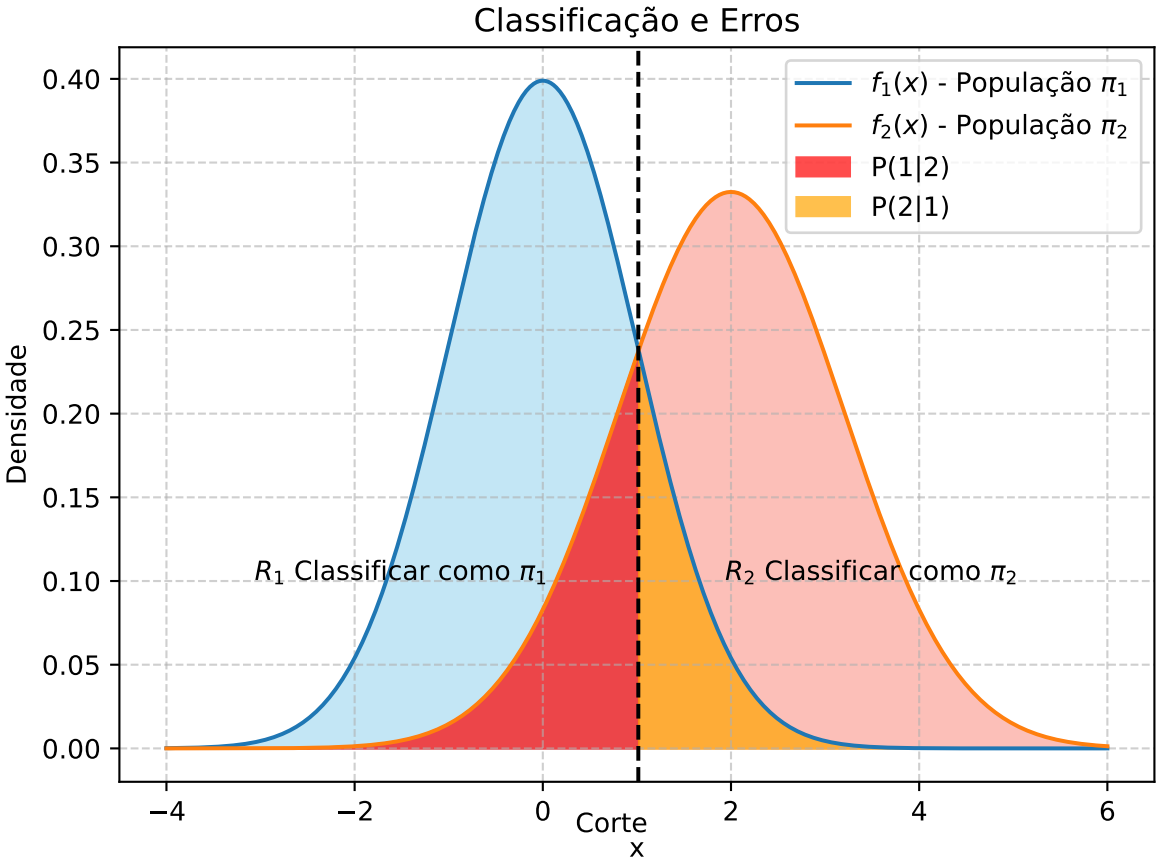

Figura 10.1: Regiões de classificação para duas populações normais univariadas.

No caso mais simples, pode-se assumir que os erros são igualmente importantes. No entanto, em diversos casos tal suposição não é razoável. Por exemplo, na análise de crédito, o custo de classificar um mau pagador como um bom pagador (concedendo um empréstimo que não será quitado) é drasticamente maior do que o custo de classificar um bom pagador como um mau pagador (negando um empréstimo e perdendo o lucro do juros). Nesses casos, definimos os custos, $C(1|2)$: Custo de alocação incorreta em π_1 e $C(2|1)$: Custo de alocação incorreta em π_2. Os custos de alocação incorretas podem ser representados em uma matriz de custos:

População Verdadeira	Classifica como π_1	Classifica como π_2
π_1	0	$C(2\|1)$
π_2	$C(1\|2)$	0

Definição 10.1. O **custo médio de classificação incorreta (ECM)** é o custo esperado dos erros de classificação, ponderado pelas probabilidades *a priori* de cada grupo. É definido como:

$$ECM = C(1|2)P(1|2)p_2 + C(2|1)P(2|1)p_1$$

O objetivo de uma regra de classificação ótima é, portanto, minimizar o Custo Médio de Classificação Incorreta (ECM). As regiões de classificação R_1 e R_2 que atingem esse mínimo são definidas pela seguinte regra de desigualdade:

- R_1**: Alocar em** π_1 **se**
$$\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \geq \frac{C(1|2)}{C(2|1)}\frac{p_2}{p_1}$$
- R_2**: Alocar em** π_2 **se**
$$\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} < \frac{C(1|2)}{C(2|1)}\frac{p_2}{p_1}$$

A regra de alocação ótima, portanto, equilibra três componentes essenciais:

1. A **razão de verossimilhança**, $\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}$, que compara quão provável é a observação $\mathbf{x}$ em cada população.
2. A **razão de custos**, $\frac{C(1|2)}{C(2|1)}$, que pondera a gravidade de cada tipo de erro.
3. A **razão das probabilidades a priori**, $\frac{p_2}{p_1}$, que considera a frequência relativa de cada população.

Quando os custos de classificação incorreta são considerados iguais ($C(1|2) = C(2|1) = C$), o ECM se torna proporcional à probabilidade total de classificação incorreta. Se os custos forem desconhecidos, uma prática comum é assumi-los como iguais, o que nos leva a minimizar a seguinte métrica.

Definição 10.2. A **probabilidade total de classificação incorreta (TPM)** é a probabilidade de classificar erroneamente uma observação, ponderada pelas probabilidades *a priori* de cada grupo. É definida como:

$$\begin{aligned} TPM &= p_1 P(2|1) + p_2 P(1|2) \\ &= p_1 \int_{R_2} f_1(\mathbf{x}) d\mathbf{x} + p_2 \int_{R_1} f_2(\mathbf{x}) d\mathbf{x} \end{aligned}$$

A regra que minimiza o TPM é a mesma que minimiza o ECM, apenas com $C(1|2)/C(2|1) = 1$.

> **Nota**
>
> Os acrônimos ECM e TPM vêm do inglês *Expected Cost of Misclassification* e *Total Probability of Misclassification*, respectivamente.

10.2 Análise Discriminante Linear (LDA) para Dois Grupos

A Análise Discriminante Linear é o método de discriminação mais fundamental. Seu objetivo é encontrar a **combinação linear** das variáveis preditoras que melhor separa os dois grupos. Essa ideia pode ser abordada de duas perspectivas principais: a de Fisher, que busca maximizar a separação, e a probabilística, que busca minimizar o erro de classificação sob certas condições. Ambas chegam à mesma solução.

10.2.1 A Perspectiva de Fisher: Maximizando a Separação

A intuição original de R.A. Fisher era geométrica e notavelmente simples: se quisermos separar dois grupos, devemos encontrar a projeção dos dados que torne essa separação o mais clara possível.

Imagine que você está olhando para duas nuvens de pontos no espaço. Se você as olhar de um ângulo ruim, elas podem parecer sobrepostas e indistinguíveis. Se você encontrar o ângulo certo, no entanto, as duas nuvens se separam claramente. Esse "ângulo de visão" em estatística é uma **combinação linear** das variáveis originais, que projeta os dados de um espaço p-dimensional para uma única linha.

O critério de Fisher, portanto, busca a projeção (a combinação linear `y = a'x`) que simultaneamente:

1. **Maximiza a distância entre as médias dos grupos projetados.**
2. **Minimiza a variância dentro de cada grupo projetado.**

Em outras palavras, queremos que os grupos fiquem o mais "espalhados" possível *entre si*, e o mais "compactos" possível *internamente*. O vetor de coeficientes a que realiza essa tarefa define a **função discriminante linear**. O método recebe esse nome porque o escore resultante é uma função linear das variáveis originais.

A Figura 10.2 ilustra a intuição de Fisher em duas dimensões. No espaço original das variáveis X_1 e X_2, os dois grupos possuem alguma sobreposição. Se projetarmos os dados em uma direção inadequada (como a linha a_2), os grupos projetados se misturam consideravelmente, tornando a classificação difícil. No entanto, ao projetar os dados na direção ótima encontrada pela LDA (a linha a_1), as distribuições dos escores resultantes ficam muito mais separadas, permitindo uma discriminação eficaz.

Formalmente, o **critério de Fisher** busca o vetor $\mathbf{a}$ que maximiza a razão entre a variância entre grupos e a variância dentro dos grupos:

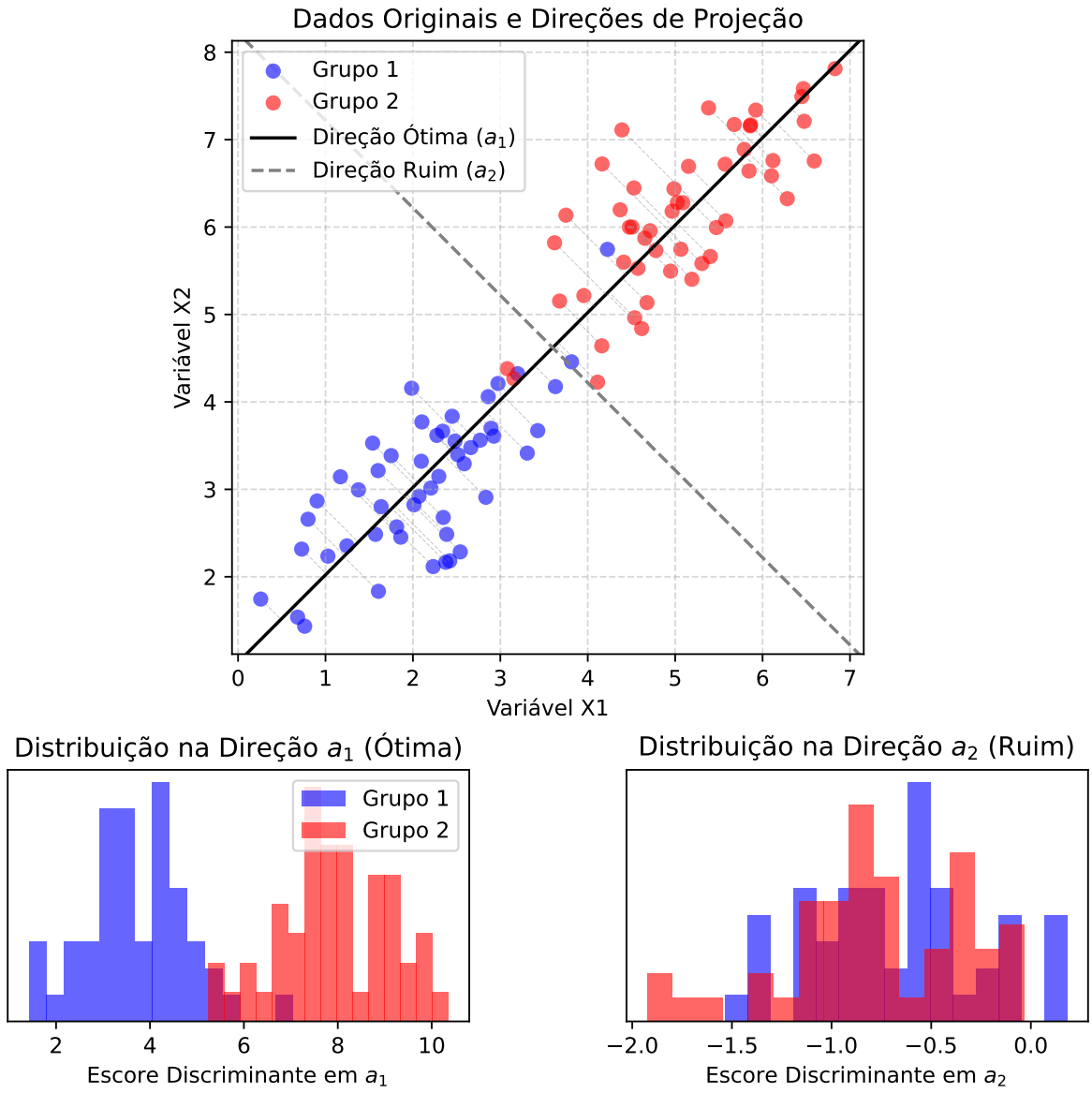

Figura 10.2: Intuição geométrica da Análise Discriminante. A projeção na direção inadequada a_2 resulta em grande sobreposição, enquanto a projeção na direção ótima a_1 maximiza a separação. As linhas tracejadas cinzas mostram a projeção perpendicular de cada ponto na reta ótima.

$$J(\mathbf{a}) = \frac{(\mathbf{a}'(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2))^2}{\mathbf{a}'\boldsymbol{\Sigma}_W\mathbf{a}}$$

onde $\boldsymbol{\Sigma}_W$ é a matriz de covariância intra-grupos (ponderada). O numerador mede a separação entre as médias projetadas, enquanto o denominador mede a dispersão total dentro dos grupos.

> **Derivação do Critério de Fisher**
>
> Para maximizar $J(\mathbf{a})$, derivamos em relação a $\mathbf{a}$ e igualamos a zero. Usando cálculo matricial, obtemos:
>
> $$\frac{\partial J}{\partial \mathbf{a}} = 0 \quad \Rightarrow \quad \boldsymbol{\Sigma}_W\mathbf{a} \propto (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$$
>
> Portanto, a solução é $\mathbf{a} \propto \boldsymbol{\Sigma}_W^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$. Na prática, estimamos com a matriz pooled:
>
> $$\hat{\mathbf{a}} = \mathbf{S}_{pooled}^{-1}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)$$

Em vez de detalhar completamente essa derivação aqui, vamos explorar a perspectiva probabilística, que, sob certas condições, chega a um resultado idêntico, fornecendo uma base estatística mais robusta para a regra de classificação.

10.2.2 A Perspectiva Probabilística: Minimizando o Erro

Uma abordagem alternativa parte da regra de classificação ótima que minimiza o erro total (Definição 10.1). Assumindo custos iguais, alocamos uma observação $\mathbf{x}$ ao grupo π_1 se for mais provável que ela venha de π_1 do que de π_2 (ponderado pelas probabilidades a priori), ou seja, se $p_1 f_1(\mathbf{x}) \geq p_2 f_2(\mathbf{x})$.

Para tornar essa regra explícita, a LDA introduz dois pressupostos-chave:

1. **Normalidade:** Os dados de cada grupo seguem uma distribuição Normal Multivariada, $\mathbf{X}_i \sim N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$.
2. **Homogeneidade de Covariâncias:** As matrizes de covariância dos grupos são iguais, $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}$.

A regra $p_1 f_1(\mathbf{x}) \geq p_2 f_2(\mathbf{x})$ é equivalente a $\ln(p_1 f_1(\mathbf{x})) \geq \ln(p_2 f_2(\mathbf{x}))$. Expandindo o logaritmo da densidade Normal Multivariada (e ignorando a constante $-(p/2)\ln(2\pi)$), temos:

$$\ln(p_i) - \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)'\boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)$$

A desigualdade se torna:

$$\ln(p_1) - \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)'\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_1) \geq \ln(p_2) - \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)$$

Os termos $\ln|\boldsymbol{\Sigma}|$ se cancelam. Expandindo os termos quadráticos e simplificando, a regra de alocação para π_1 se torna uma função linear da observação $\mathbf{x}$.

> **Detalhes da Simplificação**
>
> Cada termo quadrático pode ser expandido como:
>
> $$(\mathbf{x}-\boldsymbol{\mu}_i)'\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) = \mathbf{x}'\boldsymbol{\Sigma}^{-1}\mathbf{x} - 2\boldsymbol{\mu}_i'\boldsymbol{\Sigma}^{-1}\mathbf{x} + \boldsymbol{\mu}_i'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i$$
>
> Quando subtraímos $-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)'\ldots$ de $-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_2)'\ldots$, os termos $\mathbf{x}'\boldsymbol{\Sigma}^{-1}\mathbf{x}$ se cancelam (pois aparecem igualmente em ambos os lados), deixando apenas termos lineares em $\mathbf{x}$ e constantes.

A expressão completa, conhecida como **função discriminante linear populacional**, é:

$$\underbrace{(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}\mathbf{x}}_{\text{Escore Discriminante Linear } (y)} \geq \underbrace{\frac{1}{2}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_2) + \ln\left(\frac{p_2}{p_1}\right)}_{\text{Ponto de Corte } (c)} \tag{10.1}$$

Esta regra pode ser escrita de forma compacta como $y \geq c$. O **escore discriminante linear** (y) é obtido por $y = \mathbf{a}'\mathbf{x}$, onde o vetor de coeficientes $\mathbf{a} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ é idêntico ao encontrado pela abordagem de Fisher.

O ponto de corte c também pode ser expresso de forma mais intuitiva como a média dos escores médios de cada população, com um ajuste para as probabilidades a priori:

$$c = \frac{1}{2}(\bar{y}_1 + \bar{y}_2) + \ln\left(\frac{p_2}{p_1}\right)$$

onde $\bar{y}_1 = E[Y|\pi_1] = \mathbf{a}'\boldsymbol{\mu}_1$ e $\bar{y}_2 = E[Y|\pi_2] = \mathbf{a}'\boldsymbol{\mu}_2$ são os escores discriminantes médios (esperados) para as populações π_1 e π_2, respectivamente.

A abordagem probabilística, portanto, não apenas confirma o vetor de projeção de Fisher, mas também fornece um ponto de corte ótimo com base estatística. Se as probabilidades a priori forem iguais, o ponto de corte se torna $c = (\bar{y}_1 + \bar{y}_2)/2$, o ponto médio exato entre as médias dos grupos projetados.

10.2.3 A Função Discriminante Amostral

Na prática, os parâmetros populacionais ($\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}$) e as probabilidades a priori (p_1, p_2) são desconhecidos. Nós os substituímos por suas estimativas da amostra:

- $\boldsymbol{\mu}_i$ é estimado pela média amostral do grupo i, $\bar{\mathbf{x}}_i$.
- p_i é estimado pela proporção amostral do grupo i, n_i/n_{total}.
- A covariância comum $\boldsymbol{\Sigma}$ é estimada pela **matriz de covariância combinada (pooled)**:

$$\mathbf{S}_{pooled} = \frac{(n_1 - 1)\mathbf{S}_1 + (n_2 - 1)\mathbf{S}_2}{n_1 + n_2 - 2}$$

Substituindo esses valores, obtemos o **vetor de coeficientes da função discriminante linear amostral**:

$$\hat{\mathbf{a}} = \mathbf{S}_{pooled}^{-1}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)$$

O **escore discriminante amostral** para uma observação $\mathbf{x}$ é então $\hat{y} = \hat{\mathbf{a}}'\mathbf{x}$. A regra de classificação é alocar uma nova observação $\mathbf{x}_{new}$ a π_1 se seu escore $\hat{y}_{new}$ for maior que o ponto de corte ajustado $\hat{c}$. O ponto de corte pode ser expresso de forma mais intuitiva usando as médias dos escores discriminantes para cada grupo:

- Média do escore para o grupo 1: $\hat{\bar{y}}_1 = \hat{\mathbf{a}}'\bar{\mathbf{x}}_1$
- Média do escore para o grupo 2: $\hat{\bar{y}}_2 = \hat{\mathbf{a}}'\bar{\mathbf{x}}_2$

O ponto de corte $\hat{c}$ é então a média dos escores médios, ajustado pelas probabilidades a priori:

$$\hat{c} = \frac{1}{2}(\hat{\bar{y}}_1 + \hat{\bar{y}}_2) + \ln\left(\frac{p_2}{p_1}\right)$$

Se as probabilidades a priori forem assumidas como iguais ($p_1 = p_2 = 0.5$), o termo de log se anula, e o ponto de corte se torna simplesmente a média dos escores médios de cada grupo: $\hat{c} = (\hat{\bar{y}}_1 + \hat{\bar{y}}_2)/2$. Isso reforça a intuição de que a fronteira de decisão fica exatamente no meio do caminho entre os centroides dos grupos projetados.

10.3 Análise discriminante quadrática para dois grupos

Quando o pressuposto de homogeneidade das matrizes de covariância é violado ($\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_2$), a LDA pode não ser a melhor abordagem. A **Análise Discriminante Quadrática (QDA)** surge como uma alternativa que não assume que as matrizes de covariância populacionais são iguais.

Partindo da regra geral de alocar $\mathbf{x}$ em π_k que maximize $\ln(p_k f_k(\mathbf{x}))$, e usando a densidade normal multivariada sem assumir $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$, o termo quadrático $(\mathbf{x} - \boldsymbol{\mu}_k)' \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)$ não se cancela. A regra é então baseada no **escore de discriminação quadrática** para cada grupo:

$$d_k^Q(\mathbf{x}) = -\frac{1}{2} \ln |\boldsymbol{\Sigma}_k| - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)' \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) + \ln(p_k)$$

A observação $\mathbf{x}$ é alocada à população π_k para a qual o escore $d_k^Q(\mathbf{x})$ é maior. A fronteira de decisão, onde $d_1^Q(\mathbf{x}) = d_2^Q(\mathbf{x})$, é uma função quadrática de $\mathbf{x}$ (uma cônica), o que dá o nome ao método.

Na prática, com parâmetros desconhecidos, usamos as estimativas amostrais:

$$\hat{d}_k^Q(\mathbf{x}) = -\frac{1}{2} \ln |\mathbf{S}_k| - \frac{1}{2} (\mathbf{x} - \bar{\mathbf{x}}_k)' \mathbf{S}_k^{-1} (\mathbf{x} - \bar{\mathbf{x}}_k) + \ln(p_k)$$

A diferença fundamental entre as fronteiras de decisão da LDA e da QDA é melhor compreendida visualmente. A Figura 10.3 mostra um cenário onde os dois grupos têm matrizes de covariância distintas. A LDA, restrita a uma fronteira linear, não consegue separar os grupos de forma eficaz. A QDA, por outro lado, cria uma fronteira quadrática que se adapta muito melhor à forma e orientação de cada grupo, resultando em uma classificação superior.

10.4 Avaliando a Qualidade da Separação

Para que a análise discriminante seja útil, deve haver uma separação real entre os grupos. Avaliamos essa qualidade sob duas perspectivas: significância estatística e acurácia de classificação.

10.4.1 Teste de Significância da Separação

Primeiro, precisamos verificar se as médias dos grupos são estatisticamente diferentes. A hipótese nula é que os vetores de médias populacionais são iguais, $H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2$. Se não pudermos rejeitar H_0, a tentativa de separar os grupos não tem fundamento estatístico.

- **Teste T^2 de Hotelling:** Este é o teste multivariado análogo ao teste t para duas amostras. Ele testa diretamente a hipótese $H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2$ usando os dados multivariados originais.

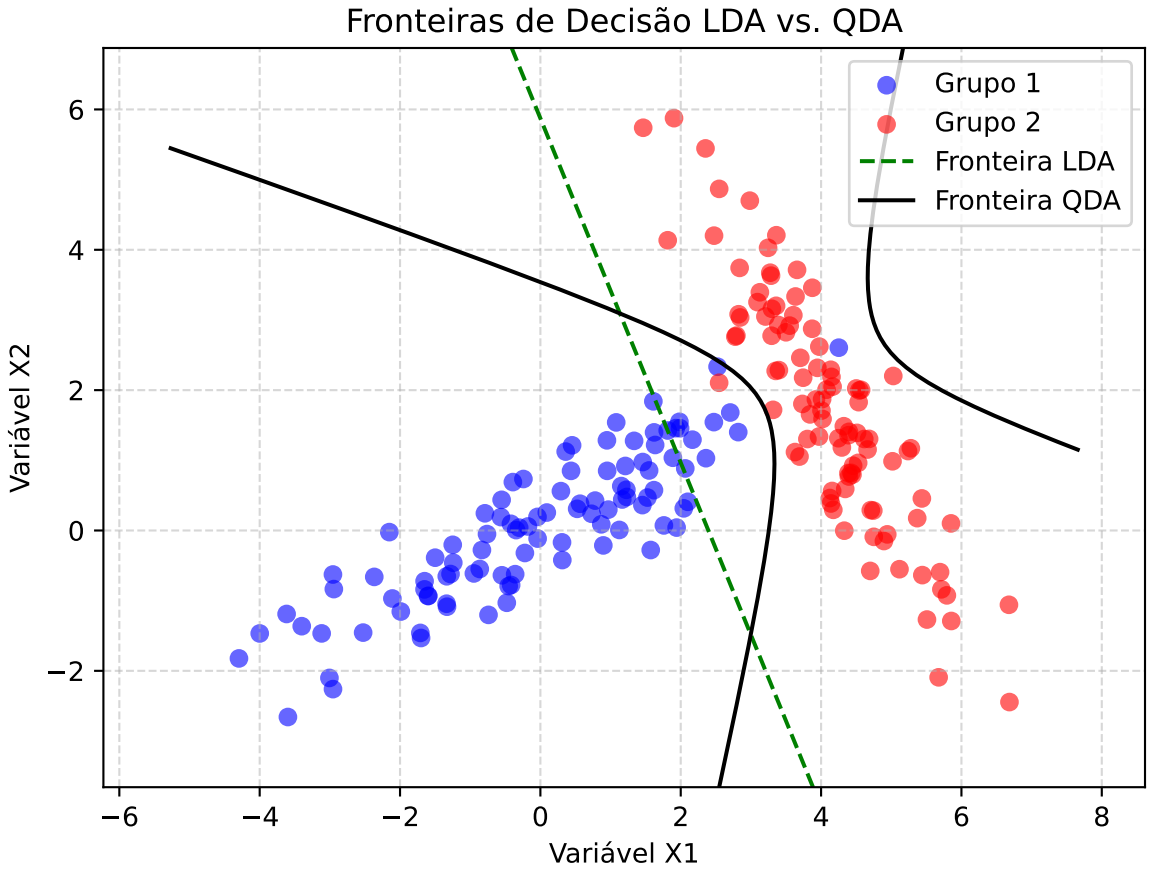

Figura 10.3: Comparação das fronteiras de decisão para LDA (linear) e QDA (quadrática). Quando as covariâncias dos grupos são diferentes, a fronteira quadrática da QDA se ajusta melhor à estrutura dos dados.

- **Teste t nos Escores Discriminantes:** Uma abordagem alternativa e equivalente é realizar um teste t de duas amostras nas médias dos escores discriminantes, $\bar{y}_1$ e $\bar{y}_2$. A hipótese nula é $H_0 : E[Y_1] = E[Y_2]$. Se a função discriminante de Fisher de fato separa os grupos, então as médias dos escores projetados devem ser significativamente diferentes.

10.4.2 Acurácia da Classificação

A qualidade de uma regra de classificação é medida pela sua taxa de erro. Existem várias maneiras de estimar essa taxa.

A estimativa mais direta é a **Taxa de Erro Aparente (APER)**, também conhecida como **taxa de erro por resubstituição**. Ela é calculada reclassificando as próprias observações da amostra de treinamento. Para isso, construímos uma **matriz de confusão**. Para dois grupos, esta matriz tem a seguinte forma:

População Real	Classifica como π_1	Classifica como π_2	Total
π_1	$n_{1\|1}$	$n_{2\|1}$	n_1
π_2	$n_{1\|2}$	$n_{2\|2}$	n_2

Onde: - n_1 e n_2 são os tamanhos das amostras dos grupos 1 e 2. - $n_{i|j}$ é o número de observações alocadas no grupo i dado que eram provenientes da população j.

Ou seja, os elementos na diagonal, $n_{1|1}$ e $n_{2|2}$, são as classificações corretas. Os elementos fora da diagonal, $n_{2|1}$ e $n_{1|2}$, são os erros de classificação.

Definição 10.3. A **Taxa de Erro Aparente (APER)** é a proporção do total de observações classificadas incorretamente na amostra de treinamento.

$$APER = \frac{n_{2|1} + n_{1|2}}{n_1 + n_2}$$

A APER tende a subestimar a taxa de erro real, pois o modelo foi ajustado para otimizar a separação justamente nesses dados. Por essa razão, ela é considerada uma medida "otimista" da performance.

10.4.3 Método Holdout (Divisão Amostral)

Uma abordagem mais robusta para estimar a taxa de erro é o método **holdout**, ou divisão amostral. A ideia é dividir aleatoriamente o conjunto de dados original em dois subconjuntos:

1. **Amostra de Treinamento:** Usada para construir o modelo discriminante (i.e., estimar os coeficientes das funções).

2. **Amostra de Validação (ou Teste):** Usada para testar o modelo. As observações nesta amostra são classificadas usando a regra criada com a amostra de treinamento.

A taxa de erro calculada na amostra de validação é uma estimativa mais honesta e imparcial do desempenho do modelo em dados futuros, pois o modelo não "viu" essas observações durante o seu treinamento. A desvantagem do método holdout é que ele reduz o tamanho da amostra usada para treinar o modelo, o que pode ser um problema em datasets pequenos. Variações como a validação cruzada (*k-fold cross-validation*) são ainda mais robustas.

10.5 Generalização para K > 2 Grupos

As ideias da análise discriminante podem ser estendidas para o caso de K populações ou grupos.

10.5.1 Funções Discriminantes de Fisher

A abordagem de Fisher de maximizar a separação entre grupos enquanto minimiza a variação dentro deles é generalizada para encontrar múltiplas funções discriminantes. Para K grupos e p variáveis, podemos encontrar até $s = \min(p, K-1)$ funções discriminantes.

Cada função é uma combinação linear das variáveis originais, $y_j = \mathbf{a}_j'\mathbf{x}$, e elas são estimadas de forma hierárquica:

1. A **primeira função discriminante**, y_1, é a combinação linear que maximiza a separação entre as médias dos K grupos.
2. A **segunda função discriminante**, y_2, é a combinação linear, não correlacionada com y_1, que melhor maximiza a separação restante.
3. O processo continua até que s funções discriminantes sejam extraídas.

Essas funções fornecem um novo espaço de dimensão reduzida (um subespaço) que é otimizado para visualização e interpretação da separação dos grupos, como vimos no exemplo do dataset Iris.

10.5.2 Regras de Classificação e Fronteiras de Decisão

A abordagem probabilística de classificação também se generaliza. A regra é alocar uma nova observação $\mathbf{x}$ ao grupo π_k para o qual a probabilidade a posteriori $P(\pi_k|\mathbf{x})$ é máxima. Isso é equivalente a alocar $\mathbf{x}$ ao grupo k que maximiza o escore $\ln(p_k f_k(\mathbf{x}))$.

Para a LDA (Covariâncias Homogêneas):

Com $\mathbf{\Sigma}_1 = \cdots = \mathbf{\Sigma}_K = \mathbf{\Sigma}_{pooled}$, o escore para o grupo k (ignorando termos constantes) é linear em $\mathbf{x}$:

$$d_k^L(\mathbf{x}) = \mathbf{x}'\mathbf{S}_{pooled}^{-1}\bar{\mathbf{x}}_k - \frac{1}{2}\bar{\mathbf{x}}_k'\mathbf{S}_{pooled}^{-1}\bar{\mathbf{x}}_k + \ln(p_k)$$

A fronteira de decisão entre quaisquer dois grupos π_i e π_j é o conjunto de pontos onde seus escores são iguais: $d_i^L(\mathbf{x}) = d_j^L(\mathbf{x})$. Isso define um **hiperplano**. O espaço p-dimensional é, portanto, particionado em K **regiões de decisão convexas**.

Para a QDA (Covariâncias Heterogêneas):

Quando as matrizes de covariância são diferentes, o escore para o grupo k permanece quadrático:

$$d_k^Q(\mathbf{x}) = -\frac{1}{2}\ln|\mathbf{S}_k| - \frac{1}{2}(\mathbf{x} - \bar{\mathbf{x}}_k)'\mathbf{S}_k^{-1}(\mathbf{x} - \bar{\mathbf{x}}_k) + \ln(p_k)$$

A fronteira de decisão entre dois grupos π_i e π_j é definida por $d_i^Q(\mathbf{x}) = d_j^Q(\mathbf{x})$, que é uma equação **quadrática**. As fronteiras são superfícies cônicas (elipses, parábolas, hipérboles), e as regiões de decisão não são necessariamente convexas.

11 Análise de Correspondência

A Análise de Correspondência (CA) é uma família de técnicas de análise exploratória de dados projetada para visualizar e analisar as associações entre variáveis categóricas. Neste capítulo, focaremos na sua forma mais fundamental: a **Análise de Correspondência Simples (ACS)**, que analisa a relação entre **duas** variáveis categóricas a partir de uma tabela de contingência. A análise de três ou mais variáveis é tratada pela **Análise de Correspondência Múltipla (ACM)**.

O principal objetivo da ACS é representar graficamente as categorias de ambas as variáveis (as linhas e colunas da tabela) como pontos em um espaço de baixa dimensão — geralmente um plano bidimensional — de forma que as relações entre elas possam ser facilmente interpretadas.

Assim como a Análise de Componentes Principais (ACP) busca resumir variáveis contínuas, a CA busca resumir variáveis categóricas. A ACP utiliza a variância como medida de dispersão e a distância Euclidiana para avaliar a proximidade entre as observações. A CA, por sua vez, utiliza uma medida análoga à variância, chamada **inércia**, e uma métrica de distância ponderada, a **distância Qui-quadrado**, para levar em conta a frequência relativa das categorias.

Geometricamente, a CA transforma os perfis de frequência das categorias em pontos em um mapa (chamado de *biplot*). Categorias com perfis de frequência semelhantes são posicionadas próximas umas das outras, enquanto categorias com perfis distintos são posicionadas distantes. Isso nos permite identificar padrões de associação, como quais categorias de uma variável tendem a ocorrer com mais frequência com quais categorias de outra variável.

11.1 A Tabela de Contingência

O ponto de partida da Análise de Correspondência é uma **tabela de contingência** (ou tabela de dupla entrada), que cruza duas variáveis categóricas.

Definição 11.1. Sejam X e Y duas variáveis categóricas com I e J categorias, respectivamente. Uma tabela de contingência $\mathbf{N}$ de dimensão $I \times J$ é uma matriz onde cada elemento n_{ij} representa a contagem (frequência absoluta) de observações que pertencem simultaneamente à i-ésima categoria de X e à j-ésima categoria de Y.

A partir de $\mathbf{N}$, definimos as seguintes quantidades:

- **Total Geral:** $n = \sum_{i=1}^{I} \sum_{j=1}^{J} n_{ij}$, o número total de observações.

- **Totais Marginais de Linha:** $n_{i.} = \sum_{j=1}^{J} n_{ij}$, o total de observações na i-ésima categoria de X.
- **Totais Marginais de Coluna:** $n_{.j} = \sum_{i=1}^{I} n_{ij}$, o total de observações na j-ésima categoria de Y.

Dividindo cada elemento da tabela pelo total geral n, obtemos a **matriz de correspondência $\mathbf{P}$**, onde cada elemento $p_{ij} = n_{ij}/n$ representa a frequência relativa conjunta.

$$\mathbf{P} = \frac{1}{n}\mathbf{N}$$

As marginais de $\mathbf{P}$ são os vetores de **probabilidades marginais** (termo que na Análise de Correspondência é frequentemente chamado de **massa**). O vetor com as probabilidades marginais das linhas, $\mathbf{r}$, e o das colunas, $\mathbf{c}$, são definidos como:

- **Probabilidade Marginal de Linha i:** $p_{i.} = \sum_{j=1}^{J} p_{ij} = n_{i.}/n$. O vetor é $\mathbf{r} = (p_{1.}, \ldots, p_{I.})'$.
- **Probabilidade Marginal de Coluna j:** $p_{.j} = \sum_{i=1}^{I} p_{ij} = n_{.j}/n$. O vetor é $\mathbf{c} = (p_{.1}, \ldots, p_{.J})'$.

As probabilidades marginais representam a importância relativa de cada categoria. Uma categoria com probabilidade marginal alta tem uma frequência maior na amostra e, portanto, terá mais peso na análise.

Exemplo 11.1. Suponha uma pesquisa sobre a preferência de plataforma de *streaming* (Netflix, Prime Video, HBO Max) por faixa etária (Jovem, Adulto). A tabela de contingência $\mathbf{N}$ com as contagens poderia ser:

Faixa Etária	Netflix	Prime Video	HBO Max	**Total (Marginal Linha)**
Jovem	50	20	30	**100** (0.50)
Adulto	40	40	20	**100** (0.50)
Total (Marginal Coluna)	**90** (0.45)	**60** (0.30)	**50** (0.25)	**200** (1.00)

O total geral é $n = 200$. A matriz de correspondência $\mathbf{P}$ seria:

$$\mathbf{P} = \begin{pmatrix} 50/200 & 20/200 & 30/200 \\ 40/200 & 40/200 & 20/200 \end{pmatrix} = \begin{pmatrix} 0.25 & 0.10 & 0.15 \\ 0.20 & 0.20 & 0.10 \end{pmatrix}$$

Os vetores de probabilidades marginais são $\mathbf{r} = (0.50, 0.50)'$ e $\mathbf{c} = (0.45, 0.30, 0.25)'$.

11.2 Perfis e a Hipótese de Independência

A CA analisa a estrutura de associação entre as variáveis examinando como os perfis de frequência de uma variável mudam conforme as categorias da outra.

Definição 11.2.

- O **perfil da linha** i é um vetor de J elementos que mostra a distribuição de frequência da variável de coluna *dentro* da i-ésima categoria da variável de linha. Cada elemento é calculado como $p_{ij}/p_{i.}$.
- O **perfil da coluna** j é um vetor de I elementos que mostra a distribuição de frequência da variável de linha *dentro* da j-ésima categoria da variável de coluna. Cada elemento é calculado como $p_{ij}/p_{.j}$.

O **perfil médio das linhas** é simplesmente o vetor de probabilidades marginais das colunas, $\mathbf{c}$. Similarmente, o **perfil médio das colunas** é o vetor de probabilidades marginais das linhas, $\mathbf{r}$.

Exemplo 11.2. Usando os dados do Exemplo 11.1:

Perfis de Linha:

Faixa Etária	Netflix	Prime Video	HBO Max	Total
Jovem	0.50	0.20	0.30	1.00
Adulto	0.40	0.40	0.20	1.00
Perfil Médio (c)	**0.45**	**0.30**	**0.25**	1.00

O perfil da linha "Jovem" (0.50, 0.20, 0.30) é diferente do perfil médio (0.45, 0.30, 0.25), indicando uma associação. Jovens têm uma preferência maior por Netflix (50% vs 45% na média) e menor por Prime Video (20% vs 30% na média).

Perfis de Coluna:

Faixa Etária	Netflix	Prime Video	HBO Max	Perfil Médio (r)
Jovem	0.56	0.33	0.60	**0.50**
Adulto	0.44	0.67	0.40	**0.50**
Total	1.00	1.00	1.00	1.00

O perfil da coluna "Netflix" (0.56, 0.44) mostra que, entre os assinantes de Netflix, 56% são jovens, um valor ligeiramente acima da média de 50%.

O conceito central para a CA é a **hipótese de independência**. Sob independência, não há associação entre as variáveis, e o conhecimento de uma não altera a expectativa sobre a outra.

Definição 11.3. Duas variáveis categóricas são independentes se a frequência conjunta esperada for o produto de suas frequências marginais. Em termos da matriz de correspondência, a independência ocorre se:

$$p_{ij} = p_{i\cdot} p_{\cdot j} \quad \forall i, j$$

A matriz contendo estas probabilidades esperadas, denotada por $\mathbf{E} = \mathbf{r}\mathbf{c}'$, representa a ausência total de associação. Sob independência, todos os perfis de linha são idênticos entre si e iguais ao perfil médio das linhas ($\mathbf{c}$). Da mesma forma, todos os perfis de coluna são idênticos e iguais ao perfil médio das colunas ($\mathbf{r}$).

A CA mede e visualiza o desvio da matriz observada $\mathbf{P}$ em relação à matriz esperada $\mathbf{E}$. Quanto mais o perfil de uma categoria específica se afasta do perfil médio, mais forte é a sua contribuição para a associação entre as variáveis.

Exemplo 11.3. Visualmente, podemos comparar os perfis de linha com o perfil médio. Se a hipótese de independência fosse verdadeira, os perfis de "Jovem" e "Adulto" seriam idênticos ao perfil "Médio". O gráfico de barras abaixo ilustra como os perfis observados se desviam do perfil médio, evidenciando a associação entre faixa etária e preferência de streaming.

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

# Dados do Exemplo 11.2
platforms = ['Netflix', 'Prime Video', 'HBO Max']
dados = {
    'Jovem': [0.50, 0.20, 0.30],
    'Adulto': [0.40, 0.40, 0.20],
    'Médio': [0.45, 0.30, 0.25]
}
df = pd.DataFrame(dados, index=platforms).reset_index().melt(id_vars='index', var_name='P
df.rename(columns={'index': 'Plataforma'}, inplace=True)

# Gráfico
fig, ax = plt.subplots(figsize=(7, 4))
sns.barplot(x='Plataforma', y='Proporção', hue='Perfil', data=df, palette='viridis', ax=a

ax.set_title('Comparação dos Perfis de Linha vs. Perfil Médio')
ax.set_ylabel('Proporção de Preferência')
ax.set_xlabel('Plataforma de Streaming')
```

```
ax.set_ylim(0, 0.6)
ax.grid(axis='y', linestyle='--', alpha=0.7)

# Adiciona os valores nas barras
for p in ax.patches:
    height = p.get_height()
    if height > 0:
        ax.annotate(f'{p.get_height():.2f}',
                    (p.get_x() + p.get_width() / 2., p.get_height()),
                    ha='center', va='center',
                    xytext=(0, 9),
                    textcoords='offset points')

plt.show()
```

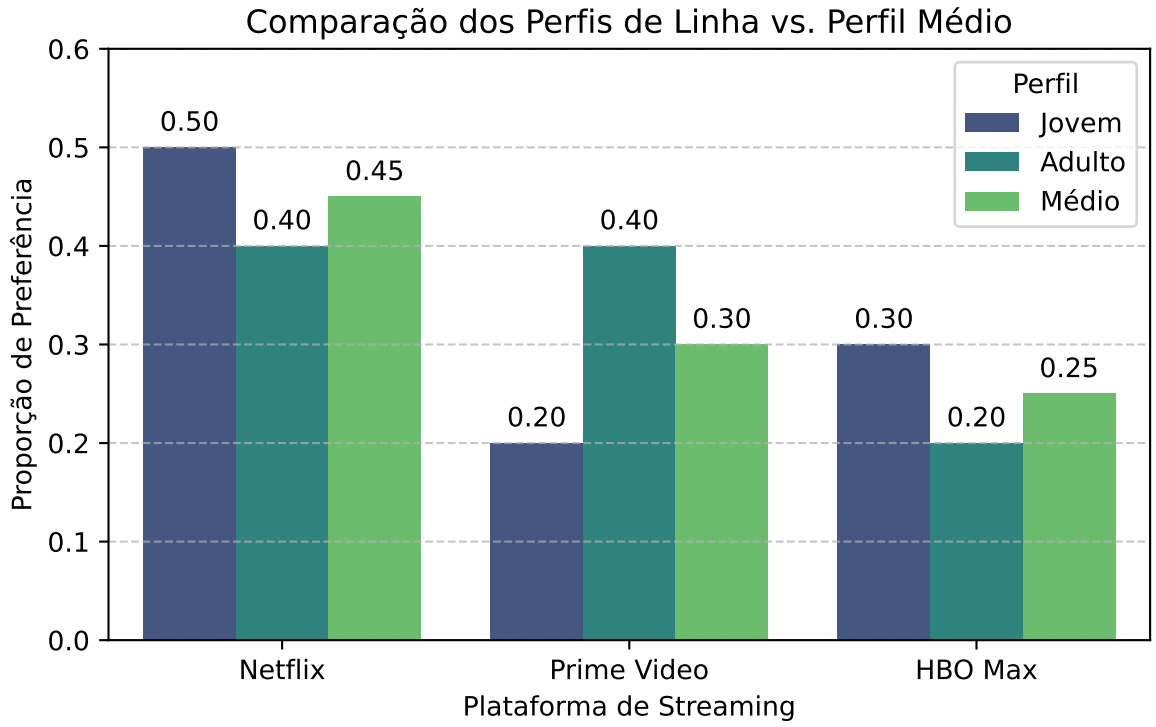

Figura 11.1: Comparação dos perfis de linha (Jovem, Adulto) com o perfil médio. A diferença na altura das barras para uma mesma plataforma indica uma associação entre as variáveis.

Da mesma forma, podemos visualizar a independência a partir dos perfis de coluna. O gráfico abaixo mostra que o perfil de preferência por faixa etária para os assinantes da "Prime Video" se destaca, com

uma proporção muito maior de adultos (67%) em comparação com o perfil médio (50%).

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

# Dados para os perfis de coluna
age_groups = ['Jovem', 'Adulto']
dados_col = {
    'Netflix': [0.56, 0.44],
    'Prime Video': [0.33, 0.67],
    'HBO Max': [0.60, 0.40],
    'Médio': [0.50, 0.50]
}
df_cols = pd.DataFrame(dados_col, index=age_groups).reset_index().melt(id_vars='index', v
df_cols.rename(columns={'index': 'Faixa Etária'}, inplace=True)

# Gráfico
fig, ax = plt.subplots(figsize=(7, 4))
sns.barplot(x='Faixa Etária', y='Proporção', hue='Perfil', data=df_cols, palette='plasma'

ax.set_title('Comparação dos Perfis de Coluna vs. Perfil Médio')
ax.set_ylabel('Proporção de Assinantes')
ax.set_xlabel('Faixa Etária')
ax.set_ylim(0, 0.8)
ax.grid(axis='y', linestyle='--', alpha=0.7)

# Adiciona os valores nas barras
for p in ax.patches:
    height = p.get_height()
    if height > 0:
        ax.annotate(f'{p.get_height():.2f}',
                    (p.get_x() + p.get_width() / 2., p.get_height()),
                    ha='center', va='center',
                    xytext=(0, 9),
                    textcoords='offset points')

# Move a legenda para fora do gráfico
ax.legend(bbox_to_anchor=(1.02, 1), loc='upper left', borderaxespad=0)

plt.show()
```

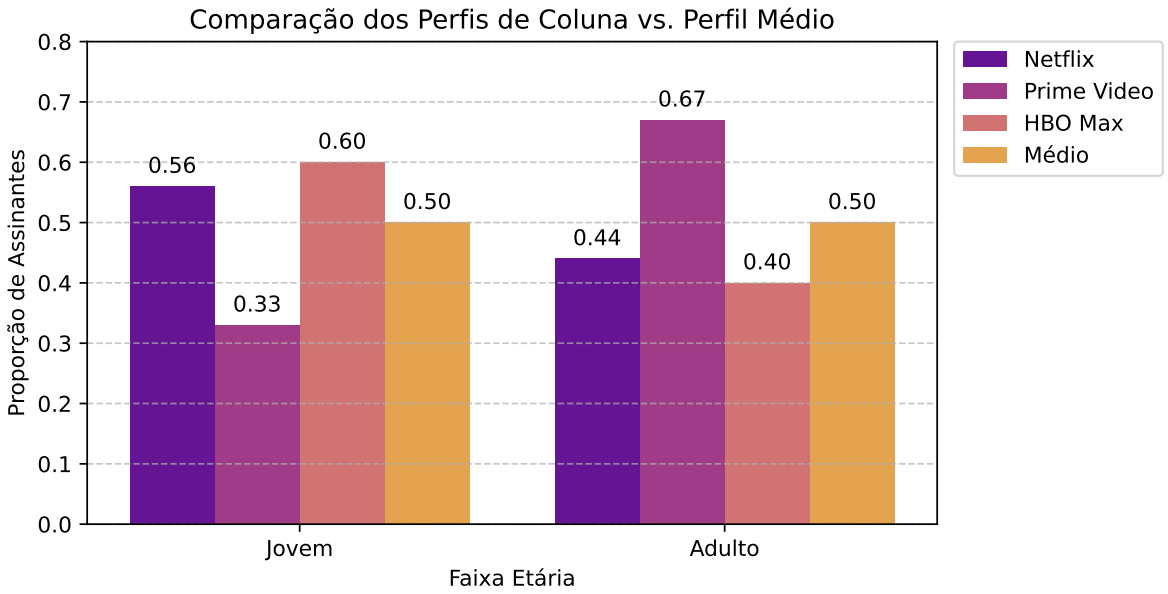

Figura 11.2: Comparação dos perfis de coluna (Netflix, Prime Video, HBO Max) com o perfil médio. As diferenças mostram como a distribuição de faixa etária varia entre os assinantes de cada plataforma.

11.3 A Distância Qui-Quadrado

Para medir a dissimilaridade entre os perfis, a distância Euclidiana padrão não é adequada. Ela sofre de dois problemas principais:

1. **Sensibilidade à Frequência Marginal:** A distância Euclidiana trata todas as categorias como igualmente importantes. No entanto, categorias com baixa frequência (e, portanto, baixa probabilidade marginal, ou massa) são mais suscetíveis a flutuações amostrais. Um pequeno desvio em uma categoria rara pode ser mais significativo do que um desvio semelhante em uma categoria muito comum.
2. **Não considera o Perfil Médio:** A distância não leva em conta a estrutura geral dos dados, representada pelos perfis médios.

Para resolver isso, a CA utiliza a **distância Qui-quadrado (χ^2)**.

Definição 11.4. A distância χ^2 é uma **distância Euclidiana ponderada**, projetada para resolver os problemas da distância Euclidiana padrão. A chave da sua formulação é a **ponderação pelo inverso da probabilidade marginal** do perfil médio (ou seja, $1/p_{\cdot j}$ para a comparação entre linhas e $1/p_{i\cdot}$ para a comparação entre colunas).

Essa ponderação tem um efeito importante:

- **Aumenta o peso** de categorias com baixa frequência (baixa probabilidade marginal). Desvios nessas categorias são considerados mais significativos, pois são mais raros.
- **Diminui o peso** de categorias com alta frequência (alta probabilidade marginal). Desvios nessas categorias são considerados menos informativos, pois pequenas flutuações são mais comuns.

Dessa forma, a distância foca na estrutura relativa dos perfis, e não apenas nas diferenças absolutas. As fórmulas são:

Distância entre dois perfis de linha (r e s):

$$d^2(r,s) = \sum_{j=1}^{J} \frac{1}{p_{.j}} \left(\frac{p_{rj}}{p_{r.}} - \frac{p_{sj}}{p_{s.}} \right)^2$$

Distância entre dois perfis de coluna (c e d):

$$d^2(c,d) = \sum_{i=1}^{I} \frac{1}{p_{i.}} \left(\frac{p_{ic}}{p_{.c}} - \frac{p_{id}}{p_{.d}} \right)^2$$

11.4 Inércia Total

A **inércia total** é a medida de dispersão central na Análise de Correspondência, análoga à variância total na ACP. Ela quantifica o desvio total dos perfis em relação à condição de independência.

Definição 11.5. A inércia total (ϕ^2) de uma tabela de contingência é a soma ponderada das distâncias χ^2 de cada perfil de linha ao perfil médio das linhas (o vetor $\mathbf{c}$, que representa as probabilidades marginais das colunas).

$$\text{Inércia Total} = \phi^2 = \sum_{i=1}^{I} p_{i.} d^2(i, \mathbf{c}) = \sum_{i=1}^{I} p_{i.} \sum_{j=1}^{J} \frac{1}{p_{.j}} \left(\frac{p_{ij}}{p_{i.}} - p_{.j} \right)^2$$

Devido à dualidade da análise, a inércia total também pode ser calculada de forma simétrica, como a soma ponderada das distâncias χ^2 de cada perfil de coluna ao perfil médio das colunas (o vetor $\mathbf{r}$):

$$\phi^2 = \sum_{j=1}^{J} p_{.j} d^2(j, \mathbf{r}) = \sum_{j=1}^{J} p_{.j} \sum_{i=1}^{I} \frac{1}{p_{i.}} \left(\frac{p_{ij}}{p_{.j}} - p_{i.} \right)^2$$

Ambas as fórmulas para a inércia total são equivalentes e se simplificam para a mesma expressão final, que revela a conexão direta da inércia com a estatística Qui-quadrado de Pearson (χ^2).

Demonstração da Inércia Total

Para mostrar como a fórmula da inércia total se simplifica, vamos partir da definição baseada nos perfis de linha:

$$\phi^2 = \sum_{i=1}^{I} p_{i.} \sum_{j=1}^{J} \frac{1}{p_{.j}} \left(\frac{p_{ij}}{p_{i.}} - p_{.j} \right)^2$$

Primeiro, trabalhamos o termo dentro do parênteses, colocando $p_{i.}$ como denominador comum:

$$\frac{p_{ij}}{p_{i.}} - p_{.j} = \frac{p_{ij} - p_{i.}p_{.j}}{p_{i.}}$$

Substituindo de volta na fórmula da inércia:

$$\phi^2 = \sum_{i=1}^{I} p_{i.} \sum_{j=1}^{J} \frac{1}{p_{.j}} \left(\frac{p_{ij} - p_{i.}p_{.j}}{p_{i.}} \right)^2 = \sum_{i=1}^{I} p_{i.} \sum_{j=1}^{J} \frac{1}{p_{.j}} \frac{(p_{ij} - p_{i.}p_{.j})^2}{p_{i.}^2}$$

Agora, podemos simplificar o termo $p_{i.}$ que está fora do somatório de j com o $p_{i.}^2$ no denominador:

$$\phi^2 = \sum_{i=1}^{I} \sum_{j=1}^{J} \frac{(p_{ij} - p_{i.}p_{.j})^2}{p_{i.}p_{.j}}$$

Esta é exatamente a fórmula da estatística Qui-quadrado de Pearson (χ^2) dividida pelo total de observações (n), onde p_{ij} é a frequência observada e $p_{i.}p_{.j}$ é a frequência esperada sob independência. O mesmo resultado é obtido ao partir da fórmula baseada nos perfis de coluna.

A expressão final, portanto, é:

$$\phi^2 = \sum_{i=1}^{I} \sum_{j=1}^{J} \frac{(p_{ij} - p_{i.}p_{.j})^2}{p_{i.}p_{.j}} = \frac{\chi^2}{n}$$

A inércia total é, portanto, a estatística χ^2 por observação, o que a torna uma medida de associação livre da influência do tamanho da amostra. Um valor de inércia igual a zero indica independência perfeita. Quanto maior a inércia, maior o grau de associação entre as variáveis. O objetivo da CA é decompor essa inércia total em dimensões ortogonais.

Relação com a Estatística χ^2 de Pearson

A estatística χ^2 de Pearson é tradicionalmente calculada com as contagens (frequências absolutas), não com as probabilidades (frequências relativas). A fórmula é:

$$\chi^2 = \sum_{i=1}^{I}\sum_{j=1}^{J} \frac{(\text{Observado}_{ij} - \text{Esperado}_{ij})^2}{\text{Esperado}_{ij}}$$

Onde $\text{Observado}_{ij} = n_{ij}$ e a contagem esperada sob independência é $\text{Esperado}_{ij} = \frac{n_{i\cdot} \times n_{\cdot j}}{n}$.
Para ver a relação entre ϕ^2 e χ^2, partimos da fórmula da inércia e substituímos as probabilidades pelas contagens. Lembrando que $p_{ij} = n_{ij}/n$ e a contagem esperada sob independência é $E_{ij} = n \cdot p_{i\cdot}p_{\cdot j}$:

$$\phi^2 = \sum_{i,j} \frac{(p_{ij} - p_{i\cdot}p_{\cdot j})^2}{p_{i\cdot}p_{\cdot j}} = \sum_{i,j} \frac{(n_{ij}/n - E_{ij}/n)^2}{E_{ij}/n}$$

Simplificando a fração, obtemos:

$$\phi^2 = \sum_{i,j} \frac{\frac{1}{n^2}(n_{ij} - E_{ij})^2}{\frac{1}{n}E_{ij}} = \frac{1}{n}\sum_{i,j} \frac{(n_{ij} - E_{ij})^2}{E_{ij}}$$

Como n_{ij} é a contagem observada, a expressão é exatamente a estatística Qui-quadrado dividida por n:

$$\phi^2 = \frac{1}{n}\sum_{i,j} \frac{(\text{Observado}_{ij} - \text{Esperado}_{ij})^2}{\text{Esperado}_{ij}} = \frac{\chi^2}{n}$$

Exemplo 11.4. Para a tabela do Exemplo 11.1, podemos calcular a inércia total usando a fórmula $\phi^2 = \sum_{i,j} \frac{(p_{ij} - p_{i\cdot}p_{\cdot j})^2}{p_{i\cdot}p_{\cdot j}}$.

A matriz de probabilidades observadas $\mathbf{P}$ é:

$$\mathbf{P} = \begin{pmatrix} 0.25 & 0.10 & 0.15 \\ 0.20 & 0.20 & 0.10 \end{pmatrix}$$

A matriz de probabilidades esperadas sob independência, $\mathbf{E}$, com elementos $e_{ij} = p_{i\cdot}p_{\cdot j}$, é:

$$\mathbf{E} = \begin{pmatrix} 0.225 & 0.150 & 0.125 \\ 0.225 & 0.150 & 0.125 \end{pmatrix}$$

A inércia é a soma dos quadrados das diferenças entre $\mathbf{P}$ e $\mathbf{E}$, ponderada pelo inverso de $\mathbf{E}$:

$$\begin{aligned} \phi^2 &= \frac{(0.25 - 0.225)^2}{0.225} + \frac{(0.10 - 0.150)^2}{0.150} + \frac{(0.15 - 0.125)^2}{0.125} + \\ &= \frac{(0.20 - 0.225)^2}{0.225} + \frac{(0.20 - 0.150)^2}{0.150} + \frac{(0.10 - 0.125)^2}{0.125} \end{aligned}$$

$$\phi^2 \approx 0.00278 + 0.01667 + 0.005 + 0.00278 + 0.01667 + 0.005 \approx 0.04889$$

Um valor de inércia total de $\phi^2 \approx 0.0489$ indica um desvio considerável da independência. A significância desta associação é avaliada pela estatística qui-quadrado de Pearson, dada por $\chi^2 = n \times \phi^2$. Para nossa amostra de $n = 200$, temos $\chi^2 \approx 200 \times 0.0489 = 9.78$.

Considerando $(2-1)(3-1) = 2$ graus de liberdade, o p-valor correspondente é de aproximadamente 0.0075. Como o p-valor é baixo, rejeita-se a hipótese de independência, o que confirma que a associação entre as variáveis é estatisticamente significativa e justifica a aplicação da Análise de Correspondência para explorar sua estrutura.

11.5 Análise de Correspondência Simples (ACS)

Sabemos que a inércia total (ϕ^2) mede o quão longe nossa tabela está da independência. O próximo passo é entender a *estrutura* dessa associação. Assim como a Análise de Componentes Principais (Capítulo 7) decompõe a variância total para encontrar as direções de maior variação, a Análise de Correspondência decompõe a **inércia total** para encontrar os eixos (ou dimensões) que melhor explicam a associação entre as categorias.

Matematicamente, isso é resolvido encontrando a melhor aproximação de baixa dimensão para a nossa tabela de associação. A ferramenta para isso é a Decomposição em Valores Singulares (SVD), que é aplicada a uma matriz que representa os desvios da independência.

Primeiro, definimos as matrizes diagonais das probabilidades marginais (massas):

- $\mathbf{D}_r$: uma matriz diagonal $I \times I$ com as probabilidades marginais das linhas ($p_{i.}$) na diagonal.
- $\mathbf{D}_c$: uma matriz diagonal $J \times J$ com as probabilidades marginais das colunas ($p_{.j}$) na diagonal.

A matriz a ser decomposta, $\mathbf{S}$, representa os resíduos da independência, padronizados pelas probabilidades marginais:

$$\mathbf{S} = \mathbf{D}_r^{-1/2}(\mathbf{P} - \mathbf{E})\mathbf{D}_c^{-1/2} \tag{11.1}$$

Onde $\mathbf{E} = \mathbf{r}\mathbf{c}'$ é a matriz esperada sob independência.

> **Relação entre S e a Inércia Total**
>
> Os elementos s_{ij} da matriz $\mathbf{S}$ são os resíduos padronizados. Uma propriedade fundamental é que a **soma dos quadrados de todos os elementos de S é igual à inércia total**:

$$\phi^2 = \sum_{i=1}^{I}\sum_{j=1}^{J} s_{ij}^2$$

Lembrando a fórmula da inércia, $\phi^2 = \sum_{i,j} \frac{(p_{ij} - p_{i.}p_{.j})^2}{p_{i.}p_{.j}}$, podemos ver que cada elemento de $\mathbf{S}$ é $s_{ij} = \frac{p_{ij} - p_{i.}p_{.j}}{\sqrt{p_{i.}p_{.j}}}$. Portanto, $s_{ij}^2 = \frac{(p_{ij} - p_{i.}p_{.j})^2}{p_{i.}p_{.j}}$, e a soma de todos os s_{ij}^2 resulta exatamente na inércia total.
Isso significa que a SVD de $\mathbf{S}$ está, de fato, decompondo a inércia total da tabela.

A SVD de $\mathbf{S}$ é:

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}'$$

Onde:

- $\mathbf{U}$ é a matriz $I \times K$ de vetores singulares à esquerda (ortogonais, $\mathbf{U}'\mathbf{U} = \mathbf{I}$).
- $\mathbf{V}$ é a matriz $J \times K$ de vetores singulares à direita (ortogonais, $\mathbf{V}'\mathbf{V} = \mathbf{I}$).
- $\mathbf{\Lambda}$ é a matriz diagonal $K \times K$ dos valores singulares λ_k, em ordem decrescente. $K = \min(I - 1, J - 1)$.

Os quadrados dos valores singulares, λ_k^2, são chamados de **inércias principais** ou autovalores. Eles representam a porção da inércia total explicada por cada dimensão. A inércia total é a soma de todas as inércias principais: $\phi^2 = \sum_k \lambda_k^2$.

Note que podemos rearranjar a Equação 11.1 para expressar a matriz de resíduos da independência, $\mathbf{P} - \mathbf{E}$, em termos dos componentes da SVD. A SVD nos dá uma forma de decompor a inércia total em componentes ortogonais:

$$\begin{aligned}
\mathbf{P} - \mathbf{E} &= \mathbf{D}_r^{1/2}\mathbf{S}\mathbf{D}_c^{1/2} \\
&= \mathbf{D}_r^{1/2}\left(\sum_{k=1}^{K} \lambda_k \mathbf{u}_k \mathbf{v}_k'\right)\mathbf{D}_c^{1/2} \\
&= \sum_{k=1}^{K} \mathbf{T}_k
\end{aligned} \tag{11.2}$$

Onde cada $\mathbf{T}_k$ é uma **matriz de fator** de posto 1 que representa a associação capturada pela k-ésima dimensão:

$$\mathbf{T}_k = \lambda_k (\mathbf{D}_r^{1/2}\mathbf{u}_k)(\mathbf{D}_c^{1/2}\mathbf{v}_k)'$$

Essa decomposição é central para a CA. Ela nos diz que a matriz de resíduos (a "tabela de associação") pode ser reconstruída como uma soma de tabelas de fatores. Como os valores singulares λ_k estão em ordem decrescente, os primeiros fatores ($\mathbf{T}_1, \mathbf{T}_2, \ldots$) são os mais importantes, pois capturam a maior parte da inércia.

Isso nos permite criar uma aproximação de baixa dimensão para a nossa tabela. Para uma aproximação com M dimensões (onde $M < K$), simplesmente somamos os primeiros M fatores:

$$\mathbf{P} - \mathbf{E} \approx \sum_{k=1}^{M} \mathbf{T}_k = \mathbf{T}_1 + \mathbf{T}_2 + \cdots + \mathbf{T}_M \tag{11.3}$$

Por exemplo, a melhor aproximação de posto 2, que será usada para criar o biplot, é $\mathbf{T}_1 + \mathbf{T}_2$. A qualidade da aproximação é medida pela proporção da inércia total capturada: $(\lambda_1^2+\lambda_2^2)/\phi^2$. Mostramos isso na prática no exemplo da Capítulo 18.

As coordenadas principais, que são muito úteis para representações visuais, são calculadas a partir dos componentes da SVD:

- **Coordenadas Principais das Linhas:** $\mathbf{F} = \mathbf{D}_r^{-1/2}\mathbf{U}\mathbf{\Lambda}$.
- **Coordenadas Principais das Colunas:** $\mathbf{G} = \mathbf{D}_c^{-1/2}\mathbf{V}\mathbf{\Lambda}$.

> **Relação entre as Coordenadas Principais e a Decomposição da Inércia**
>
> Podemos reescrever a matriz de fator $\mathbf{T}_k$ usando as coordenadas da k-ésima dimensão (os vetores coluna $\mathbf{f}_k$ e $\mathbf{g}_k$).
> A partir das definições de $\mathbf{F}$ e $\mathbf{G}$, podemos expressar os vetores singulares $\mathbf{u}_k$ e $\mathbf{v}_k$ em função das coordenadas:
>
> $$\mathbf{u}_k = \frac{1}{\lambda_k}\mathbf{D}_r^{1/2}\mathbf{f}_k \quad \text{e} \quad \mathbf{v}_k = \frac{1}{\lambda_k}\mathbf{D}_c^{1/2}\mathbf{g}_k$$
>
> Substituindo-os na definição de $\mathbf{T}_k = \lambda_k(\mathbf{D}_r^{1/2}\mathbf{u}_k)(\mathbf{D}_c^{1/2}\mathbf{v}_k)'$, a matriz de fator pode ser expressa como:
>
> $$\mathbf{T}_k = \frac{1}{\lambda_k}(\mathbf{D}_r\mathbf{f}_k)(\mathbf{g}_k'\mathbf{D}_c)$$

11.6 O Biplot e a Interpretação dos Resultados

A partir da decomposição da inércia, podemos criar uma representação visual aproximada, mas informativa, da associação, projetando as categorias em um espaço de poucas dimensões — geralmente duas. O resultado é o **biplot**, um gráfico que exibe as categorias de linha e de coluna simultaneamente, cujos eixos são as dimensões de maior inércia. Usando as coordenadas das duas primeiras dimensões, o biplot nos permite interpretar a estrutura subjacente dos dados.

A seguir, apresentamos notas práticas de como interpretar um biplot. Essa explicação é melhor entendida com um exemplo, por isso sugere-se a leitura do Capítulo 18 na sequência.

Como interpretar o biplot simétrico:

1. **Distância entre pontos do mesmo conjunto:** A distância Euclidiana entre dois pontos de linha (ou dois pontos de coluna) no biplot aproxima a distância χ^2 entre seus perfis.
    - **Pontos próximos:** Categorias com perfis semelhantes.
    - **Pontos distantes:** Categorias com perfis muito diferentes.
2. **Posição relativa à origem:** A distância de um ponto à origem (0,0) indica sua contribuição para a inércia total.
    - **Pontos longe da origem:** Categorias com perfis atípicos, que se afastam do perfil médio e contribuem significativamente para a associação.
    - **Pontos perto da origem:** Categorias com perfis próximos ao perfil médio, que desempenham um papel menor na estrutura de associação.
3. **Relação entre pontos de conjuntos diferentes:** A associação entre uma categoria de linha e uma categoria de coluna é interpretada pela sua **proximidade e direção** em relação à origem.
    - Categorias cujos pontos estão no mesmo lado (quadrante) do gráfico tendem a ter uma **associação positiva** (ocorrem juntas com mais frequência do que o esperado).
    - Categorias em lados opostos da origem tendem a ter uma **associação negativa**.

> ⚠ Cuidado na Interpretação de Distâncias
>
> Em um biplot simétrico (o tipo mais comum), a distância Euclidiana **entre um ponto de linha e um ponto de coluna** não tem uma interpretação direta e formal. Embora a proximidade visual possa sugerir uma forte associação — o que muitas vezes é verdade devido às relações de transição — a interpretação rigorosa deve se basear na posição relativa dos pontos em relação à origem e nas distâncias dentro de cada conjunto de categorias.

Além de inspecionar o biplot visualmente, é importante verificar a **qualidade de sua representação** no mapa. Para isso, observamos a proporção da inércia de cada ponto que é capturada pelas dimensões do biplot. A métrica mais adequada é o **Cosseno ao Quadrado** (cos^2). A fórmula para a linha i na dimensão k é:

$$cos^2_{ik} = \frac{F^2_{ik}}{\sum_{j=1}^{K} F^2_{ij}}$$

Na prática, para que o biplot seja altamente informativo, precisamos queremos que as primeiras duas dimensões capturem uma grande fração da inércia. Em outras palavras, queremos $cos^2_{i1} + cos^2_{i2}$ próximo de 1. É importante que isso seja verificado também para as colunas.

Uma vez confirmada a boa representação dos pontos, investigamos o significado dos eixos através da **contribuição** de cada ponto para a inércia de uma dimensão. A contribuição da linha i para a dimensão k é:

$$\text{Ctr}_{ik} = \frac{p_{i.} F_{ik}^2}{\lambda_k^2}$$

onde λ_k^2 é a inércia do eixo. Um eixo é interpretado com base nas categorias que mais contribuem para ele.

11.7 Análise de Correspondência Múltipla (ACM)

Enquanto a Análise de Correspondência Simples (ACS) explora a associação entre **duas** variáveis, a **Análise de Correspondência Múltipla (ACM)** generaliza essa técnica para **três ou mais** variáveis categóricas. Embora seja frequentemente apresentada como uma única técnica, é muito útil pensar na ACM como dois métodos distintos, dependendo da matriz que serve de base para a análise:

1. **ACM via Matriz Indicadora (Z):** Foca na relação entre indivíduos e variáveis.
2. **ACM via Matriz (ou tabela) de Burt (B):** Foca exclusivamente na relação entre as variáveis.

Ambas as abordagens revelam a mesma estrutura de associação entre as categorias, mas diferem no objeto de análise e nas propriedades matemáticas (especialmente na inércia).

11.7.1 ACM via Matriz Indicadora (Z)

A abordagem mais completa da ACM utiliza a **matriz indicadora** (**Z**), também conhecida como **tabela de lógica**.

Definição 11.6. Dado um conjunto de dados com N indivíduos e Q variáveis categóricas, a **matriz indicadora Z** é uma matriz binária de dimensão $N \times J$, onde J é o número total de categorias somadas de todas as variáveis.

- As **linhas** representam os indivíduos.
- As **colunas** representam as categorias de todas as variáveis.
- O elemento $z_{ij} = 1$ se o indivíduo i possui a categoria j, e 0 caso contrário.

Ao aplicar o algoritmo da ACS diretamente sobre a matriz **Z**, tratamos os dados como uma grande tabela de contingência.

Principais Características:

- **Mapa Conjunto:** Gera coordenadas tanto para as **categorias** quanto para os **indivíduos**.
- **Interpretação:** Indivíduos próximos possuem perfis de resposta semelhantes. Um indivíduo é posicionado no centróide das categorias que ele selecionou.

- **Aplicação:** Ideal para identificar perfis de indivíduos e para etapas subsequentes de análise, como agrupamento (clusterização).

11.7.2 ACM via Matriz de Burt ($\mathbf{B}$)

Se o interesse é estritamente analisar como as variáveis se relacionam entre si, ignorando os indivíduos, utilizamos a **Matriz de Burt** ($\mathbf{B}$).

Definição 11.7. A **Matriz de Burt** é uma matriz simétrica $J \times J$ que contém todas as tabelas de contingência cruzada entre os pares de variáveis. Ela é definida matematicamente como o produto da matriz indicadora por sua transposta:

$$\mathbf{B} = \mathbf{Z}'\mathbf{Z}$$

A aplicação da ACS sobre a Matriz de Burt foca puramente nas associações entre categorias.

Principais Características:

- **Mapa de Categorias:** Gera coordenadas apenas para as categorias. A configuração geométrica é idêntica à obtida via $\mathbf{Z}$.
- **Inércia Inflada:** As inércias (autovalores) obtidas nesta análise são os **quadrados** das inércias obtidas na análise via matriz $\mathbf{Z}$. Isso faz com que as dimensões pareçam explicar uma porcentagem muito maior da inércia total, o que pode ser enganoso.
- **Eficiência:** Computacionalmente mais eficiente para grandes bases de dados onde N é muito grande, pois a matriz $J \times J$ é geralmente menor que $N \times J$.

Tabela 11.4: Comparação entre as abordagens de ACM via Matriz Indicadora e Matriz de Burt.

Característica	ACM via Matriz Indicadora ($\mathbf{Z}$)	ACM via Matriz de Burt ($\mathbf{B}$)
Matriz Base	Indivíduos $\times$ Categorias ($N \times J$)	Categorias $\times$ Categorias ($J \times J$)
Foco	Indivíduos e Categorias	Apenas Categorias
Inércia (λ)	λ_k (menores)	λ_k^2 (maiores, infladas)
Uso Principal	Análise detalhada, Clusterização	Análise estrutural das variáveis

11.7.3 A Particularidade da Inércia na ACM

A interpretação da inércia (variância explicada) requer cuidados especiais na ACM e se manifesta de forma diferente dependendo da matriz utilizada:

1. **Na ACM via Matriz Indicadora (Z):** A inércia total é fixa e depende apenas do desenho da tabela (J categorias e Q variáveis):

$$\text{Inércia Total} = \frac{J - Q}{Q}$$

   Devido a essa "inflação" da inércia total (que inclui muita variância de ruído), as porcentagens de inércia explicada por cada dimensão tendem a ser **muito baixas** (pessimistas), frequentemente abaixo de 20%, mesmo quando existem associações fortes.

2. **Na ACM via Matriz de Burt (B):** Como os autovalores são os quadrados dos autovalores de $\mathbf{Z}$ (λ_k^2), as porcentagens de explicação tornam-se matematicamente maiores e mais **otimistas**.

Portanto, a "particularidade" é relevante para ambos, mas com sinais opostos: $\mathbf{Z}$ subestima a qualidade da representação visual, enquanto $\mathbf{B}$ tende a superestimá-la.

Dica Prática:

- Na ACM via matriz indicadora, não descarte dimensões apenas porque a % de inércia parece baixa.
- Utilize o **Critério de Benzécri**: considere significativas as dimensões cujos autovalores superam a inércia média ($1/Q$).
- Para reportar resultados, é comum calcular a **Inércia Ajustada** (frequentemente baseada nos autovalores da Matriz de Burt), que fornece uma estimativa mais realista da porcentagem de variação explicada.

12 Análise de Correlação Canônica

A Análise de Correlação Canônica (ACC), proposta por Hotelling (1935), é uma técnica estatística multivariada que busca identificar e quantificar a associação entre dois conjuntos de variáveis. O seu princípio básico é desenvolver uma combinação linear das variáveis em cada um dos grupos, tal que a correlação entre essas duas combinações seja maximizada.

Essas combinações lineares são chamadas de **variáveis canônicas** e suas associações são denominadas de **correlações canônicas**. Diferente da Análise de Componentes Principais (ACP), que trata de um único conjunto de variáveis, a ACC foca na relação *entre* dois conjuntos distintos.

O objetivo principal é simplificar a estrutura de correlação entre os grupos, reduzindo-a a um pequeno número de pares de variáveis canônicas independentes. A técnica é ideal quando se deseja explorar a interdependência entre dois conjuntos de métricas (por exemplo, desempenho de vendas vs. perfil psicológico) ou prever um conjunto a partir do outro.

Exemplo 12.1. Suponha que 50 vendedores foram avaliados por dois conjuntos de variáveis numa empresa:

1. **Desempenho de trabalho**: crescimento de vendas, rentabilidade das vendas, e novas contas.
2. **Desempenho psicológico**: criatividade, raciocínio, habilidade matemática.

Pode-se obter uma variável canônica para cada grupo e verificar a associação entre elas – a correlação canônica. Se esta associação for positivamente significativa, é interessante para a empresa aplicar os testes psicológicos aos candidatos de emprego, pois indivíduos com maiores escores nesses testes tenderiam a ser os melhores vendedores.

12.1 Fundamentação Teórica

Seja $\mathbf{X}$ um vetor aleatório de dimensão $(p+q)\times 1$, particionado em dois subvetores:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}^{(1)} \\ \mathbf{X}^{(2)} \end{bmatrix}$$

onde $\mathbf{X}^{(1)}$ tem dimensão $p \times 1$ e $\mathbf{X}^{(2)}$ tem dimensão $q \times 1$. Assumimos, sem perda de generalidade, que $p \leq q$.

O vetor de médias e a matriz de covariâncias populacional são particionados correspondentemente:

$$\boldsymbol{\mu} = E[\mathbf{X}] = \begin{bmatrix} \boldsymbol{\mu}^{(1)} \\ \boldsymbol{\mu}^{(2)} \end{bmatrix}, \quad \boldsymbol{\Sigma} = \text{Cov}(\mathbf{X}) = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix}$$

onde $\boldsymbol{\Sigma}_{11}$ ($p \times p$) e $\boldsymbol{\Sigma}_{22}$ ($q \times q$) são as matrizes de covariâncias de $\mathbf{X}^{(1)}$ e $\mathbf{X}^{(2)}$, respectivamente, e $\boldsymbol{\Sigma}_{12} = \boldsymbol{\Sigma}_{21}'$ ($p \times q$) é a matriz de covariâncias cruzadas. Assumimos que $\boldsymbol{\Sigma}$ é positiva definida, implicando que $\boldsymbol{\Sigma}_{11}$ e $\boldsymbol{\Sigma}_{22}$ são não singulares.

A ACC busca pares de combinações lineares

$$U = \mathbf{a}'\mathbf{X}^{(1)} \quad \text{e} \quad V = \mathbf{b}'\mathbf{X}^{(2)}$$

onde $\mathbf{a}$ ($p \times 1$) e $\mathbf{b}$ ($q \times 1$) são vetores de coeficientes. As propriedades estatísticas dessas combinações são:

$$\begin{aligned} \text{Var}(U) &= \mathbf{a}'\boldsymbol{\Sigma}_{11}\mathbf{a} \\ \text{Var}(V) &= \mathbf{b}'\boldsymbol{\Sigma}_{22}\mathbf{b} \\ \text{Cov}(U, V) &= \mathbf{a}'\boldsymbol{\Sigma}_{12}\mathbf{b} \end{aligned}$$

A correlação entre U e V é:

$$\text{Corr}(U, V) = \frac{\mathbf{a}'\boldsymbol{\Sigma}_{12}\mathbf{b}}{\sqrt{\mathbf{a}'\boldsymbol{\Sigma}_{11}\mathbf{a}}\sqrt{\mathbf{b}'\boldsymbol{\Sigma}_{22}\mathbf{b}}}$$

O objetivo é maximizar essa correlação, o que equivale (sob restrições de normalização $\text{Var}(U) = \text{Var}(V) = 1$) a maximizar a covariância $\mathbf{a}'\boldsymbol{\Sigma}_{12}\mathbf{b}$.

Solução via SVD. A abordagem moderna e computacionalmente estável utiliza a decomposição em valores singulares (SVD). A ideia central é "branquear" os dados de cada grupo separadamente, para eliminar as dependências internas de cada grupo.

Considere a transformação de variáveis:

$$\tilde{\mathbf{X}}^{(1)} = \boldsymbol{\Sigma}_{11}^{-1/2}\mathbf{X}^{(1)}, \quad \tilde{\mathbf{X}}^{(2)} = \boldsymbol{\Sigma}_{22}^{-1/2}\mathbf{X}^{(2)}$$

Após essa transformação, cada grupo tem matriz de covariância identidade: $\text{Cov}(\tilde{\mathbf{X}}^{(1)}) = \mathbf{I}_p$ e $\text{Cov}(\tilde{\mathbf{X}}^{(2)}) = \mathbf{I}_q$. A covariância cruzada entre os grupos transformados é precisamente a **matriz de coerência**:

$$\text{Cov}(\tilde{\mathbf{X}}^{(1)}, \tilde{\mathbf{X}}^{(2)}) = \mathbf{K} = \boldsymbol{\Sigma}_{11}^{-1/2}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1/2}$$

Agora, buscar as combinações lineares $U = \mathbf{a}'\mathbf{X}^{(1)}$ e $V = \mathbf{b}'\mathbf{X}^{(2)}$ com variância unitária que maximizem a covariância é equivalente a buscar vetores unitários $\tilde{\mathbf{a}}$ e $\tilde{\mathbf{b}}$ que maximizem $\tilde{\mathbf{a}}'\mathbf{K}\tilde{\mathbf{b}}$. Este é exatamente o problema que a SVD resolve: a SVD de $\mathbf{K}$,

$$\mathbf{K} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}'$$

fornece os vetores singulares à esquerda $\mathbf{u}_k$ (colunas de $\mathbf{U}$) e à direita $\mathbf{v}_k$ (colunas de $\mathbf{V}$) que maximizam sucessivamente $\mathbf{u}_k'\mathbf{K}\mathbf{v}_k$, sujeitos à ortogonalidade com os pares anteriores. Os valores singulares ρ_k^* (elementos diagonais de $\mathbf{\Lambda}$, ordenados $\rho_1^* \geq \rho_2^* \geq \cdots \geq \rho_p^*$) são as **correlações canônicas**.

Os vetores de coeficientes canônicos no espaço original são obtidos revertendo a transformação:

$$\mathbf{a}_k = \mathbf{\Sigma}_{11}^{-1/2}\mathbf{u}_k \quad \text{e} \quad \mathbf{b}_k = \mathbf{\Sigma}_{22}^{-1/2}\mathbf{v}_k$$

i Derivação Clássica via Multiplicadores de Lagrange

Historicamente, Hotelling (1935) derivou a ACC usando multiplicadores de Lagrange. O problema de otimização é:

$$\max_{\mathbf{a},\mathbf{b}} \mathbf{a}'\mathbf{\Sigma}_{12}\mathbf{b} \quad \text{sujeito a} \quad \mathbf{a}'\mathbf{\Sigma}_{11}\mathbf{a} = 1,\ \mathbf{b}'\mathbf{\Sigma}_{22}\mathbf{b} = 1$$

O Lagrangiano, com multiplicadores λ e μ, é:

$$L(\mathbf{a}, \mathbf{b}, \lambda, \mu) = \mathbf{a}'\mathbf{\Sigma}_{12}\mathbf{b} - \frac{\lambda}{2}(\mathbf{a}'\mathbf{\Sigma}_{11}\mathbf{a} - 1) - \frac{\mu}{2}(\mathbf{b}'\mathbf{\Sigma}_{22}\mathbf{b} - 1)$$

As condições de primeira ordem são obtidas derivando em relação a $\mathbf{a}$ e $\mathbf{b}$:

$$\frac{\partial L}{\partial \mathbf{a}} = \mathbf{\Sigma}_{12}\mathbf{b} - \lambda\mathbf{\Sigma}_{11}\mathbf{a} = \mathbf{0} \quad \Rightarrow \quad \mathbf{\Sigma}_{12}\mathbf{b} = \lambda\mathbf{\Sigma}_{11}\mathbf{a}$$

$$\frac{\partial L}{\partial \mathbf{b}} = \mathbf{\Sigma}_{21}\mathbf{a} - \mu\mathbf{\Sigma}_{22}\mathbf{b} = \mathbf{0} \quad \Rightarrow \quad \mathbf{\Sigma}_{21}\mathbf{a} = \mu\mathbf{\Sigma}_{22}\mathbf{b}$$

Pode-se mostrar que $\lambda = \mu = \rho$, onde ρ é a correlação canônica. Da primeira equação, isolamos $\mathbf{b} = \frac{1}{\rho}\mathbf{\Sigma}_{22}^{-1}\mathbf{\Sigma}_{21}\mathbf{a}$ e substituímos na segunda:

$$\mathbf{\Sigma}_{12}\left(\frac{1}{\rho}\mathbf{\Sigma}_{22}^{-1}\mathbf{\Sigma}_{21}\mathbf{a}\right) = \rho\mathbf{\Sigma}_{11}\mathbf{a}$$

Reorganizando, obtemos a equação de autovalores:

$$\mathbf{\Sigma}_{11}^{-1}\mathbf{\Sigma}_{12}\mathbf{\Sigma}_{22}^{-1}\mathbf{\Sigma}_{21}\mathbf{a} = \rho^2\mathbf{a}$$

onde ρ^2 é o autovalor associado. A maior raiz ρ_1^{*2} fornece a primeira correlação canônica ρ_1^*, e os vetores $\mathbf{a}_1$ e $\mathbf{b}_1$ definem o primeiro par de variáveis canônicas. Os pares subsequentes são obtidos dos autovalores seguintes.

Esta abordagem é matematicamente equivalente à SVD de $\mathbf{K}$: os autovalores ρ^2 são exatamente os quadrados dos valores singulares de $\mathbf{K}$. A formulação via SVD é preferida por sua estabilidade numérica.

Propriedades das Variáveis Canônicas. As variáveis canônicas $(U_1, V_1), \ldots, (U_p, V_p)$ satisfazem:

1. **Variância unitária**: $\text{Var}(U_k) = \text{Var}(V_k) = 1$ para todo k.
2. **Não correlação intra-grupo**: $\text{Cov}(U_k, U_l) = 0$ e $\text{Cov}(V_k, V_l) = 0$ para $k \neq l$.
3. **Não correlação inter-grupo**: $\text{Cov}(U_k, V_l) = \begin{cases} \rho_k^* & \text{se } k = l \\ 0 & \text{se } k \neq l \end{cases}$

Essas propriedades garantem que toda a estrutura de correlação complexa entre $\mathbf{X}^{(1)}$ e $\mathbf{X}^{(2)}$ foi decomposta em p pares de correlações simples e independentes, ordenadas por magnitude.

12.2 Interpretação e Visualização

O que são as correlações canônicas? As correlações canônicas $\rho_1^*, \rho_2^*, \ldots, \rho_p^*$ quantificam a força da associação linear entre os dois conjuntos de variáveis ao longo de diferentes dimensões ortogonais. A primeira correlação canônica ρ_1^* captura a associação linear máxima possível entre qualquer combinação linear dos dois grupos. A segunda ρ_2^* captura a máxima associação residual, ortogonal à primeira, e assim sucessivamente.

Por exemplo, se $\rho_1^* = 0.85$, isso significa que existe uma direção no espaço de $\mathbf{X}^{(1)}$ e outra em $\mathbf{X}^{(2)}$ ao longo das quais as variáveis canônicas têm correlação de 0.85 - a mais forte possível. Se $\rho_2^* = 0.12$ é muito pequena, isso indica que, após remover a estrutura explicada pela primeira dimensão, há pouca associação linear residual.

Quantas dimensões usar? Embora haja p pares de variáveis canônicas, frequentemente apenas os primeiros pares têm correlações canônicas substanciais. Na prática:

- Correlações canônicas pequenas ($\rho_k^* < 0.3$) geralmente não são substantivamente relevantes
- Testes de significância sequenciais (seção Inferência) ajudam a determinar quantas são estatisticamente diferentes de zero
- O índice de redundância (abaixo) indica se as dimensões retidas explicam uma fração razoável da variância

12.2.1 Cargas Canônicas

As variáveis canônicas são construções matemáticas sem significado físico direto. Os **coeficientes canônicos** $\mathbf{a}_k$ e $\mathbf{b}_k$ definem as combinações lineares, mas são difíceis de interpretar devido à multicolinearidade e ao branqueamento. A interpretação é feita através das **cargas canônicas** (*canonical loadings*), que são as correlações entre as variáveis originais e suas respectivas variáveis canônicas.

A carga canônica da j-ésima variável do primeiro grupo com a k-ésima variável canônica U_k é:

$$\ell_{j,k}^{(1)} = \text{Corr}(X_j^{(1)}, U_k) = \frac{\text{Cov}(X_j^{(1)}, U_k)}{\sqrt{\text{Var}(X_j^{(1)})}\sqrt{\text{Var}(U_k)}} = \frac{[\boldsymbol{\Sigma}_{11}\mathbf{a}_k]_j}{\sigma_j \cdot 1}$$

onde $\sigma_j = \sqrt{\sigma_{jj}}$ é o desvio padrão de $X_j^{(1)}$, e usamos o fato de que $\text{Var}(U_k) = 1$. De forma análoga, para o segundo grupo:

$$\ell_{j,k}^{(2)} = \text{Corr}(X_j^{(2)}, V_k) = \frac{[\boldsymbol{\Sigma}_{22}\mathbf{b}_k]_j}{\sigma_j^{(2)}}$$

onde $\sigma_j^{(2)}$ é o desvio padrão de $X_j^{(2)}$.

> **Simplificação com Variáveis Padronizadas**
>
> Quando trabalhamos com variáveis padronizadas (média 0 e variância 1), temos $\sigma_j = 1$ para todas as variáveis. Nesse caso, as fórmulas se simplificam para:
>
> $$\boldsymbol{\ell}_k^{(1)} = \mathbf{R}_{11}\mathbf{a}_k, \quad \boldsymbol{\ell}_k^{(2)} = \mathbf{R}_{22}\mathbf{b}_k$$
>
> onde $\mathbf{R}_{11}$ e $\mathbf{R}_{22}$ são as matrizes de correlação dos grupos. Esta é a forma comumente encontrada na literatura e em softwares estatísticos.

Cargas elevadas (em módulo) indicam que a variável original tem grande influência na definição daquela dimensão canônica. Cargas são preferidas aos coeficientes pois têm escala padronizada (correlações, entre -1 e 1) e são mais estáveis numericamente.

12.2.2 Cargas Cruzadas

Enquanto as cargas canônicas medem a relação entre variáveis e suas *próprias* variáveis canônicas, as **cargas cruzadas** (*cross-loadings*) medem a correlação entre as variáveis de um grupo e a variável canônica do *outro* grupo. Essa medida é importante porque captura diretamente a associação entre os dois conjuntos de variáveis.

A carga cruzada da j-ésima variável do primeiro grupo com a k-ésima variável canônica do segundo grupo V_k é:

$$\ell_{j,k}^{(1\to 2)} = \text{Corr}(X_j^{(1)}, V_k) = \frac{[\boldsymbol{\Sigma}_{12}\mathbf{b}_k]_j}{\sigma_j}$$

Relação entre Cargas e Cargas Cruzadas

As cargas e cargas cruzadas estão relacionadas pela correlação canônica. Partindo da condição de primeira ordem $\boldsymbol{\Sigma}_{12}\mathbf{b}_k = \rho_k^*\boldsymbol{\Sigma}_{11}\mathbf{a}_k$, dividindo elemento a elemento por σ_j:

$$\ell_{j,k}^{(1\to 2)} = \rho_k^* \cdot \ell_{j,k}^{(1)}$$

Portanto, a carga cruzada é simplesmente a carga canônica multiplicada pela correlação canônica. Isso tem uma interpretação intuitiva: a associação entre uma variável de um grupo e a variável canônica do outro é "atenuada" pela força da correlação canônica.

As cargas cruzadas oferecem insight direto sobre quais variáveis de um conjunto estão mais fortemente associadas ao outro conjunto. Por exemplo, uma carga cruzada alta de $X_j^{(1)}$ com V_1 indica que essa variável contribui significativamente para a relação entre os dois grupos.

Exemplo 12.2. Suponha que, para o primeiro par canônico ($k = 1$) com $\rho_1^* = 0.80$, obtivemos as seguintes cargas canônicas para três variáveis do grupo 1:

Variável	Carga Canônica $\ell^{(1)}$	Carga Cruzada $\ell^{(1\to 2)}$
X_1	0.90	$0.80 \times 0.90 = 0.72$
X_2	0.45	$0.80 \times 0.45 = 0.36$
X_3	-0.70	$0.80 \times (-0.70) = -0.56$

Interpretação:

- X_1 tem forte associação positiva ($\ell = 0.90$) com a variável canônica U_1 e, consequentemente, associação moderada-alta ($\ell_{\text{cruz}} = 0.72$) com V_1.
- X_2 tem associação fraca com a dimensão canônica.
- X_3 tem associação negativa: valores altos de X_3 estão associados a valores baixos de U_1 (e, por extensão, de V_1).

Note que as cargas cruzadas são sempre menores em módulo que as cargas canônicas (exceto quando $\rho^* = 1$), refletindo a imperfeição da relação entre os grupos.

12.2.3 Índice de Redundância

O **índice de redundância** quantifica a proporção da variância de um conjunto de variáveis que é explicada pela variável canônica do *outro* conjunto. Ele é fundamental para avaliar a utilidade prática da ACC, pois uma correlação canônica alta não garante que muita variância seja explicada.

O índice de redundância para o grupo 1, explicado pela k-ésima dimensão canônica de V, é definido em duas etapas:

1. Variância Extraída: A proporção média da variância de $\mathbf{X}^{(1)}$ explicada pela *própria* variável canônica U_k:

$$\text{VE}_k^{(1)} = \frac{1}{p}\sum_{j=1}^{p}\left(\ell_{j,k}^{(1)}\right)^2$$

2. Índice de Redundância: A variância extraída é então ponderada pelo quadrado da correlação canônica:

$$\text{Red}_k^{(1)} = \text{VE}_k^{(1)} \times (\rho_k^*)^2$$

Equivalentemente, o índice de redundância pode ser calculado diretamente como a média dos quadrados das cargas cruzadas:

$$\text{Red}_k^{(1)} = \frac{1}{p}\sum_{j=1}^{p}\left(\ell_{j,k}^{(1\to 2)}\right)^2$$

Esta equivalência decorre da relação $\ell^{(1\to 2)} = \rho^* \cdot \ell^{(1)}$.

Observar o índice de redundância, e não apenas a correlação canônica, é crucial: uma correlação canônica alta ($\rho^* = 0.9$) com variância extraída baixa ($\text{VE} = 0.15$) resulta em redundância de apenas $0.15 \times 0.81 = 0.12$, indicando que, apesar da forte correlação entre as variáveis canônicas, V_k explica pouca variância das variáveis originais de $\mathbf{X}^{(1)}$.

Visualização. A visualização gráfica das cargas canônicas (através de *heliographs*, gráficos de barras, ou diagramas de dispersão das variáveis canônicas) facilita a identificação de padrões e a compreensão de quais variáveis originais "carregam" em cada dimensão canônica. Para uma aplicação completa incluindo cálculo, interpretação detalhada e visualização com dados reais, consulte o exemplo prático em Capítulo 20.

12.3 Inferência Estatística

Na prática, as matrizes populacionais $\boldsymbol{\Sigma}$ são desconhecidas. Trabalhamos com uma amostra de tamanho n e utilizamos os estimadores de máxima verossimilhança, substituindo $\boldsymbol{\Sigma}$ pela matriz de covariância amostral:

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{bmatrix}$$

As correlações canônicas amostrais $\hat{\rho}_k^*$ e os vetores de coeficientes $\hat{\mathbf{a}}_k, \hat{\mathbf{b}}_k$ são obtidos aplicando a SVD à matriz de coerência amostral $\hat{\mathbf{K}} = \mathbf{S}_{11}^{-1/2}\mathbf{S}_{12}\mathbf{S}_{22}^{-1/2}$.

> **Nota sobre Variáveis Padronizadas (Matriz de Correlação)**
>
> É comum realizar a análise utilizando a matriz de correlação $\mathbf{R}$ em vez da matriz de covariância $\mathbf{S}$, o que equivale a trabalhar com variáveis padronizadas (média 0, variância 1). As **correlações canônicas** são invariantes a mudanças de escala, logo são idênticas em ambos os casos. Os **coeficientes canônicos** terão escalas diferentes.
> A distinção entre padronização e branqueamento é importante:
>
> - **Padronização (via $\mathbf{R}$)**: Normaliza variâncias marginais para 1, mantendo correlações intra-grupo.
> - **Branqueamento (via $\mathbf{S}^{-1/2}$)**: Remove totalmente a correlação intra-grupo (covariância vira identidade), permitindo que a SVD foque na associação *entre* grupos.

Para testar a significância da associação entre os dois conjuntos, testamos a hipótese nula $H_0 : \mathbf{\Sigma}_{12} = \mathbf{0}$ (independência) contra $H_1 : \mathbf{\Sigma}_{12} \neq \mathbf{0}$. Assumindo que os dados provêm de uma distribuição Normal Multivariada, utilizamos o teste de razão de verossimilhança com a estatística Lambda de Wilks:

$$\Lambda = \prod_{i=1}^{p}(1 - \hat{\rho}_i^{*2})$$

Para grandes amostras, a estatística de Bartlett segue aproximadamente uma distribuição qui-quadrado:

$$-\left(n - 1 - \frac{p+q+1}{2}\right)\ln(\Lambda) \sim \chi^2_{pq}$$

Rejeitamos H_0 ao nível α se a estatística exceder o quantil $\chi^2_{pq}(\alpha)$. Se H_0 for rejeitada, podemos testar sequencialmente a significância das correlações canônicas remanescentes. Para testar se as correlações restantes após remover as k primeiras são zero, a estatística é:

$$\Lambda_k = \prod_{i=k+1}^{\min(p,q)}(1 - \hat{\rho}_i^{*2})$$

E a aproximação qui-quadrado utiliza $(p-k)(q-k)$ graus de liberdade:

$$-\left(n - 1 - \frac{p+q+1}{2}\right)\ln(\Lambda_k) \sim \chi^2_{(p-k)(q-k)}$$

12.4 Considerações Práticas

1. **Tamanho da Amostra**: A ACC requer tamanhos de amostra relativamente grandes para produzir resultados estáveis. Uma regra prática sugere pelo menos 10 observações por variável.
2. **Linearidade**: O método assume relações lineares entre os conjuntos de variáveis.
3. **Outliers**: A técnica é sensível a outliers, que devem ser identificados e tratados.
4. **Multicolinearidade**: Alta multicolinearidade dentro de um conjunto de variáveis pode tornar a inversão das matrizes instável.

Parte III

Exemplos

13 Exemplo manual: ACP

Neste exemplo, vamos detalhar passo a passo a aplicação da Análise de Componentes Principais (ACP) em um pequeno conjunto de dados. O objetivo é demonstrar manualmente todos os cálculos, desde a preparação dos dados até a interpretação dos resultados, seguindo a metodologia apresentada na Capítulo 7.

13.1 O Cenário

Vamos expandir o exemplo da intuição geométrica, que usava **Peso** e **Altura**. Adicionaremos uma terceira variável, **Renda** (em milhares de R$), para um grupo de 5 indivíduos. A ideia é que Peso e Altura sejam correlacionados, mas a Renda não tenha uma correlação forte com eles.

Nosso conjunto de dados inicial é:

Indivíduo	Peso (kg)	Altura (cm)	Renda (R$ 1000)
1	65	170	5.5
2	72	182	4.0
3	58	165	7.0
4	81	190	3.5
5	75	178	5.0

13.2 Passo 1: Preparação dos Dados

Conforme discutido no Capítulo 3, a ACP é sensível à escala das variáveis. Portanto, o primeiro passo é **padronizar** os dados. Isso envolve duas etapas: centralizar (subtrair a média) e escalonar (dividir pelo desvio padrão).

13.2.1 1.1. Calcular a Média e o Desvio Padrão

Primeiro, calculamos a média e o desvio padrão para cada variável.

$$\bar{x}_{peso} = \frac{65 + 72 + 58 + 81 + 75}{5} = 70.2\,\text{kg}$$
$$\bar{x}_{altura} = \frac{170 + 182 + 165 + 190 + 178}{5} = 177.0\,\text{cm}$$
$$\bar{x}_{renda} = \frac{5.5 + 4.0 + 7.0 + 3.5 + 5.0}{5} = 5.0\,(R\$1000)$$

Agora, os desvios padrão (usando a fórmula com denominador $n - 1$):

$$s_{peso} = \sqrt{\frac{(65 - 70.2)^2 + ... + (75 - 70.2)^2}{4}} = 8.64\,\text{kg}$$
$$s_{altura} = \sqrt{\frac{(170 - 177)^2 + ... + (178 - 177)^2}{4}} = 9.67\,\text{cm}$$
$$s_{renda} = \sqrt{\frac{(5.5 - 5.0)^2 + ... + (5.0 - 5.0)^2}{4}} = 1.35\,(R\$1000)$$

13.2.2 1.2. Padronizar os Dados

Com as médias e desvios padrão, podemos padronizar cada observação x_{ij} usando a fórmula $z_{ij} = (x_{ij} - \bar{x}_j)/s_j$.

Por exemplo, para o Indivíduo 1:

$$z_{1,peso} = \frac{65 - 70.2}{8.64} = -0.60$$
$$z_{1,altura} = \frac{170 - 177}{9.67} = -0.72$$
$$z_{1,renda} = \frac{5.5 - 5.0}{1.35} = 0.37$$

Aplicando isso a todos os dados, obtemos a matriz de dados padronizados $\mathbf{Z}$:

$$\mathbf{Z} = \begin{pmatrix} -0.60 & -0.72 & 0.37 \\ 0.21 & 0.52 & -0.74 \\ -1.41 & -1.24 & 1.48 \\ 1.25 & 1.34 & -1.11 \\ 0.56 & 0.10 & 0.00 \end{pmatrix}$$

13.3 Passo 2: Calcular a Matriz de Correlação

Como estamos trabalhando com dados padronizados, a ACP será realizada sobre a **matriz de correlação $\mathbf{R}$**. A matriz de correlação pode ser calculada como:

$$\mathbf{R} = \frac{1}{n-1}\mathbf{Z}'\mathbf{Z}$$

> **Por que essa fórmula funciona?**
>
> Essa fórmula só é válida porque $\mathbf{Z}$ contém dados **padronizados** (média 0 e variância 1). Para dados padronizados, a covariância amostral entre duas variáveis é igual à sua correlação. Para dados não padronizados, usaríamos $\mathbf{S} = \frac{1}{n-1}(\mathbf{X} - \bar{\mathbf{X}})'(\mathbf{X} - \bar{\mathbf{X}})$.

Calculando $\mathbf{Z}'\mathbf{Z}$:

$$\mathbf{Z}'\mathbf{Z} = \begin{pmatrix} 4.00 & 3.85 & -0.81 \\ 3.85 & 4.00 & -1.18 \\ -0.81 & -1.18 & 4.00 \end{pmatrix}$$

Dividindo por $n - 1 = 4$, obtemos a matriz de correlação $\mathbf{R}$:

$$\mathbf{R} = \begin{pmatrix} 1.00 & 0.96 & -0.20 \\ 0.96 & 1.00 & -0.29 \\ -0.20 & -0.29 & 1.00 \end{pmatrix}$$

Como esperado, a correlação entre Peso e Altura (0.96) é muito alta, enquanto a Renda tem uma correlação fraca e negativa com as outras duas variáveis.

13.4 Passo 3: Decomposição Espectral da Matriz de Correlação

O próximo passo é encontrar os autovalores (λ) e autovetores ($\mathbf{e}$) da matriz de correlação $\mathbf{R}$. Eles são a solução da equação $\mathbf{R}\mathbf{e} = \lambda\mathbf{e}$, que é equivalente a resolver $(\mathbf{R} - \lambda\mathbf{I})\mathbf{e} = \mathbf{0}$.

Isso requer encontrar as raízes do polinômio característico $det(\mathbf{R} - \lambda\mathbf{I}) = 0$.

$$det\begin{pmatrix} 1.00 - \lambda & 0.96 & -0.20 \\ 0.96 & 1.00 - \lambda & -0.29 \\ -0.20 & -0.29 & 1.00 - \lambda \end{pmatrix} = 0$$

Resolver este determinante cúbico manualmente é trabalhoso. Usando uma calculadora ou software, encontramos os seguintes autovalores:

$$\lambda_1 = 1.98 \quad \lambda_2 = 1.00 \quad \lambda_3 = 0.02$$

13.4.1 Interpretação dos Autovalores

A variância total no sistema é a soma dos autovalores (que é igual ao traço da matriz $\mathbf{R}$, ou seja, 3). - **Variância Total** = $1.98 + 1.00 + 0.02 = 3.00$

A proporção da variância explicada por cada componente é: - **CP1**: $\frac{1.98}{3.00} = 66.0\%$ - **CP2**: $\frac{1.00}{3.00} = 33.3\%$ - **CP3**: $\frac{0.02}{3.00} = 0.7\%$

Os dois primeiros componentes juntos explicam $66.0\% + 33.3\% = 99.3\%$ da variância total. Isso indica que podemos reduzir a dimensionalidade de 3 para 2 com uma perda mínima de informação.

13.5 Passo 4: Calcular os Autovetores

Agora, para cada autovalor, resolvemos o sistema $(\mathbf{R} - \lambda_i \mathbf{I})\mathbf{e}_i = \mathbf{0}$ para encontrar o autovetor correspondente $\mathbf{e}_i$.

- Para $\lambda_1 = 1.98$:

$$\begin{pmatrix} -0.98 & 0.96 & -0.20 \\ 0.96 & -0.98 & -0.29 \\ -0.20 & -0.29 & -0.98 \end{pmatrix} \begin{pmatrix} e_{11} \\ e_{12} \\ e_{13} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

  A solução, após normalização (para que $\mathbf{e}_1'\mathbf{e}_1 = 1$), é:

$$\mathbf{e}_1 = \begin{pmatrix} 0.69 \\ 0.71 \\ -0.15 \end{pmatrix}$$

- Para $\lambda_2 = 1.00$:

$$\begin{pmatrix} 0.00 & 0.96 & -0.20 \\ 0.96 & 0.00 & -0.29 \\ -0.20 & -0.29 & 0.00 \end{pmatrix} \begin{pmatrix} e_{21} \\ e_{22} \\ e_{23} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

  A solução normalizada é:

$$\mathbf{e}_2 = \begin{pmatrix} 0.18 \\ -0.22 \\ -0.96 \end{pmatrix}$$

- Para $\lambda_3 = 0.02$:

$$\begin{pmatrix} 0.98 & 0.96 & -0.20 \\ 0.96 & 0.98 & -0.29 \\ -0.20 & -0.29 & 0.98 \end{pmatrix} \begin{pmatrix} e_{31} \\ e_{32} \\ e_{33} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

  A solução normalizada é:

$$\mathbf{e}_3 = \begin{pmatrix} -0.70 \\ 0.67 \\ -0.24 \end{pmatrix}$$

13.6 Passo 5: Interpretação dos Componentes

Os autovetores (ou *loadings*) nos dizem como as variáveis originais se combinam para formar cada componente.

- **Componente Principal 1 (CP_1)**:

$$Y_1 = 0.69 \cdot Z_{peso} + 0.71 \cdot Z_{altura} - 0.15 \cdot Z_{renda}$$

  Este componente é basicamente uma média ponderada de Peso e Altura, com uma pequena contribuição negativa da Renda. Podemos interpretá-lo como um índice de **"Tamanho Corporal"**. As cargas altas e positivas para Peso e Altura confirmam a alta correlação entre essas variáveis.

- **Componente Principal 2 (CP_2)**:

$$Y_2 = 0.18 \cdot Z_{peso} - 0.22 \cdot Z_{altura} - 0.96 \cdot Z_{renda}$$

  Este componente é dominado pela Renda, com uma carga muito alta e negativa. As cargas para Peso e Altura são pequenas. Podemos interpretar o CP_2 como um índice de **"Status Socioeconômico Inverso"**, já que ele é quase que inteiramente uma representação da Renda (com sinal trocado).

13.7 Passo 6: Calcular os Scores dos Componentes

Finalmente, podemos calcular os valores (scores) dos componentes principais para cada indivíduo. Usamos a fórmula $\mathbf{Y} = \mathbf{ZE}$, onde $\mathbf{E}$ é a matriz cujas colunas são os autovetores (seguindo a notação de Capítulo 7).

$$\mathbf{E} = \begin{pmatrix} 0.69 & 0.18 & -0.70 \\ 0.71 & -0.22 & 0.67 \\ -0.15 & -0.96 & -0.24 \end{pmatrix}$$

Para o Indivíduo 1, com dados padronizados $(-0.60, -0.72, 0.37)$:

$$y_{11} = (-0.60)(0.69) + (-0.72)(0.71) + (0.37)(-0.15) = -0.98$$

$$y_{12} = (-0.60)(0.18) + (-0.72)(-0.22) + (0.37)(-0.96) = -0.31$$

Calculando para todos os indivíduos, obtemos a matriz de scores $\mathbf{Y}$:

Indivíduo	CP1 (Tamanho)	CP2 (Renda Inversa)
1	-0.98	-0.31
2	0.57	0.85
3	-2.19	-1.18
4	2.09	1.35
5	0.46	0.08

13.8 Conclusão

Este exemplo demonstra o poder da ACP. Começamos com três variáveis e, através de uma derivação passo a passo, conseguimos: 1. **Reduzir a dimensionalidade**: Mostramos que 99.3% da informação está contida em dois componentes. 2. **Criar variáveis não correlacionadas**: O CP_1 e o CP_2 são, por construção, ortogonais. 3. **Interpretar a estrutura latente**: Identificamos que a principal fonte de variação nos dados é o "Tamanho Corporal" (uma combinação de Peso e Altura), seguida pelo "Status Socioeconômico" (representado pela Renda).

A análise manual, embora trabalhosa, revela a mecânica exata da técnica, solidificando a compreensão teórica apresentada no capítulo principal.

14 Rotação fatorial em R

A Análise Fatorial (AF) é uma técnica estatística poderosa usada para identificar estruturas latentes (fatores) subjacentes a um conjunto de variáveis observadas. No entanto, a solução matemática inicial de uma AF raramente é interpretável. É aqui que a **rotação fatorial** se torna a etapa mais crítica do processo. A rotação transforma a matriz de cargas fatoriais inicial em uma solução mais simples e teoricamente mais significativa, sem alterar as propriedades matemáticas fundamentais da solução.

Este documento oferece um exemplo prático e didático, totalmente focado em demonstrar o impacto das diferentes estratégias de rotação. Usaremos o software R e o clássico conjunto de dados `bfi` (Big Five Inventory) do pacote `psych`.

Objetivos:

1. Comparar métodos de extração: Componentes Principais (PAF) e Máxima Verossimilhança (ML).
2. Demonstrar a dificuldade de interpretar uma solução fatorial **não rotacionada**.
3. Aplicar e interpretar uma **rotação ortogonal (Varimax)**, que assume fatores não correlacionados.
4. Aplicar e interpretar uma **rotação oblíqua (Promax)**, que permite a correlação entre os fatores.
5. Discutir como a escolha da rotação afeta a interpretação final e a validade teórica dos resultados.

14.1 Passo 1: Análise Descritiva e Adequação dos Dados

Primeiro, carregamos os pacotes necessários e o conjunto de dados `bfi`. Este dataset contém respostas de 2800 indivíduos a 25 itens de personalidade.

```
# Carregar pacotes
library(psych)
library(ggplot2)
library(dplyr)
library(tidyr)
library(corrplot)

# Carregar os dados do Big Five Inventory
data(bfi, package = "psych")

# Selecionar apenas as 25 variáveis de itens de personalidade
```

```
bfi_items <- bfi[, 1:25]

# Remover linhas com dados ausentes para simplificar
bfi_complete <- na.omit(bfi_items)

knitr::kable(head(bfi_complete))
```

Tabela 14.1: Exemplo de respostas no banco de dados Big Five

	A1	A2	A3	A4	A5	C1	C2	C3	C4	C5	E1	E2	E3	E4	E5	N1	N2	N3	N4	N5	O1	O2	O3	O4	O5
61617	2	4	3	4	4	2	3	3	4	4	3	3	3	4	4	3	4	2	2	3	3	6	3	4	3
61618	2	4	5	2	5	5	4	4	3	4	1	1	6	4	3	3	3	3	5	5	4	2	4	3	3
61620	5	4	5	4	4	4	5	4	2	5	2	4	4	4	5	4	5	4	2	3	4	2	5	5	2
61621	4	4	6	5	5	4	4	3	5	5	5	3	4	4	4	2	5	2	4	1	3	3	4	3	5
61622	2	3	3	4	5	4	4	5	3	2	2	2	5	4	5	2	3	4	4	3	3	3	4	3	3
61623	6	6	5	6	5	6	6	6	1	3	2	1	6	5	6	3	5	2	2	3	4	3	5	6	1

As 25 variáveis correspondem a 5 itens para cada um dos traços do "Big Five":

- **A1-A5:** Amabilidade (Agreeableness)
- **C1-C5:** Conscienciosidade (Conscientiousness)
- **E1-E5:** Extroversão (Extraversion)
- **N1-N5:** Neuroticismo (Neuroticism)
- **O1-O5:** Abertura à Experiência (Openness)

A hipótese teórica é que os 5 itens que medem o mesmo traço (e.g., N1 a N5) estarão altamente correlacionados entre si e se agruparão em um único fator latente (Neuroticismo).

Uma boa prática é verificar se os dados são fatorizáveis. Para isso, podemos usar o teste de Bartlett e a medida KMO.

```
# Teste de Bartlett
bartlett_test <- cortest.bartlett(bfi_complete)
```

```
R was not square, finding R from data
```

```
# Teste KMO
kmo_test <- KMO(bfi_complete)

# Exibindo os resultados de forma concisa
cat("Teste de Bartlett: p-valor =", bartlett_test$p.value, "\n")
```

```
Teste de Bartlett: p-valor = 0
```

```
cat("Medida KMO Geral (Overall MSA):", round(kmo_test$MSA, 2), "\n")
```

```
Medida KMO Geral (Overall MSA): 0.85
```

O p-valor de Bartlett próximo de zero e o KMO de 0.85 (“meritório”) sugerem que os dados têm correlações suficientes para justificar uma análise fatorial.

14.2 Passo 2: Extração Inicial dos Fatores e Escolha do Número de Fatores

Antes de rotacionar, precisamos extrair os fatores. Dois métodos comuns são a **Fatoração do Eixo Principal** (ou “componentes principais” para o modelo fatorial) e a **Máxima Verossimilhança**. Vamos extrair 5 fatores usando ambos os métodos (sem rotação) para ver a solução inicial.

```
# Extração via Fatoração do Eixo Principal (Principal Axis Factoring)
fa_pa <- fa(bfi_complete, nfactors = 5, rotate = "none", fm = "pa")
cat("Cargas - Fatoração do Eixo Principal (PA):\n")
```

```
Cargas - Fatoração do Eixo Principal (PA):
```

```
print(fa_pa$loadings, cutoff = 0.3)
```

```
Loadings:
   PA1    PA2    PA3    PA4    PA5
A1                              -0.371
A2  0.467                       0.340
A3  0.534  0.302
A4  0.417
A5  0.581
C1  0.343          0.446
C2  0.336          0.477
C3  0.319          0.351  0.310
C4 -0.465         -0.452
C5 -0.493
E1 -0.408
```

```
E2 -0.619                         0.323
E3  0.527  0.328
E4  0.599        -0.329
E5  0.513
N1 -0.441  0.636
N2 -0.423  0.616
N3 -0.407  0.611
N4 -0.528  0.416
N5 -0.345  0.413
O1  0.328              -0.360
O2                      0.370
O3  0.407              -0.446
O4
O5                      0.412

                 PA1   PA2   PA3   PA4   PA5
SS loadings    4.600 2.268 1.549 1.218 0.956
Proportion Var 0.184 0.091 0.062 0.049 0.038
Cumulative Var 0.184 0.275 0.337 0.385 0.424
```

```r
# Extração via Máxima Verossimilhança (Maximum Likelihood)
fa_ml <- fa(bfi_complete, nfactors = 5, rotate = "none", fm = "ml")
cat("\nCargas - Máxima Verossimilhança (ML):\n")
```

```
Cargas - Máxima Verossimilhança (ML):
```

```r
print(fa_ml$loadings, cutoff = 0.3)
```

```
Loadings:
   ML1    ML2    ML3    ML4    ML5
A1                              -0.322
A2 -0.396  0.354                  0.334
A3 -0.462  0.401                  0.321
A4 -0.386
A5 -0.546
C1                0.465
C2                0.511
C3                0.404
C4  0.441        -0.512
```

```
C5  0.485        -0.358
E1  0.355 -0.309
E2  0.585                       0.336
E3 -0.446  0.436
E4 -0.552  0.333
E5 -0.409  0.429
N1  0.609  0.566
N2  0.587  0.543
N3  0.533  0.479
N4  0.591
N5  0.421
O1                     -0.409
O2                      0.388
O3 -0.329  0.349       -0.491
O4                     -0.307  0.311
O5                      0.433

                ML1   ML2   ML3   ML4   ML5
SS loadings    4.451 2.379 1.546 1.221 0.977
Proportion Var 0.178 0.095 0.062 0.049 0.039
Cumulative Var 0.178 0.273 0.335 0.384 0.423
```

As duas soluções iniciais são numericamente diferentes, mas conceitualmente iguais: são **ininterpretáveis**. Um primeiro fator geral domina, e as variáveis se distribuem de forma confusa nos demais. Isso reforça a necessidade da rotação.

Para determinar o número de fatores a extrair de forma mais objetiva, usamos a Análise Paralela.

```
fa.parallel(bfi_complete, fa = "fa", fm = "pa")
```

```
Parallel analysis suggests that the number of factors =  6  and the number of components =  NA
```

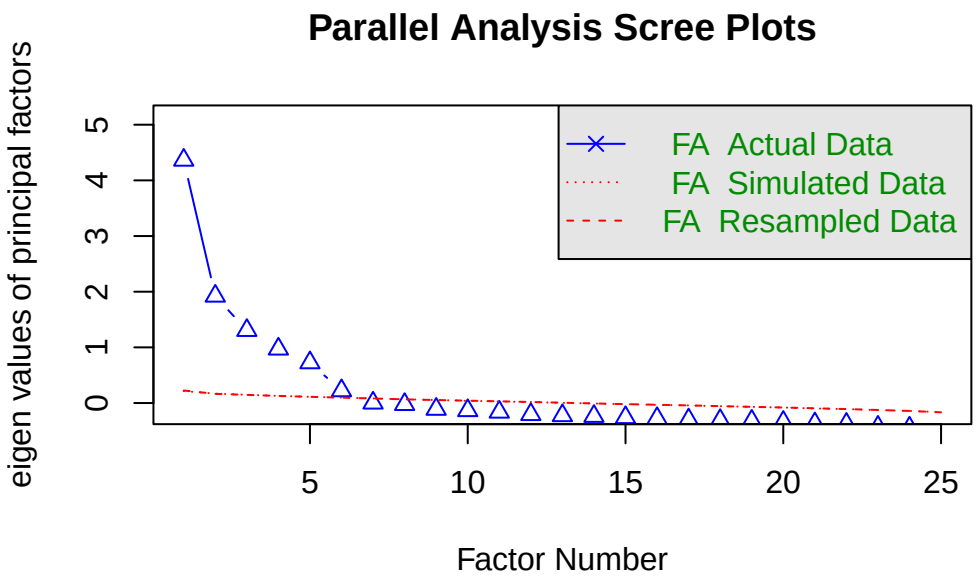

Figura 14.1: Análise Paralela sugerindo a extração de 6 fatores.

A Análise Paralela (Figura 14.1) sugere 6 fatores. No entanto, como nosso objetivo é testar a teoria dos Big Five, **prosseguiremos com a extração de 5 fatores**, uma decisão comum quando a teoria é forte.

14.3 Passo 3: Rotação Ortogonal (Varimax)

A rotação **Varimax** "limpa" a estrutura sob a suposição de que **os fatores não são correlacionados entre si**.

```
# Análise Fatorial com rotação Varimax
fa_varimax <- factanal(bfi_complete,
                       factors = 5,
                       rotation = "varimax")

cat("Cargas Fatoriais (AF) - Rotação Varimax:\n")
```

```
Cargas Fatoriais (AF) - Rotação Varimax:
```

```
print(fa_varimax$loadings, cutoff = 0.3, sort = TRUE)
```

```
Loadings:
   Factor1 Factor2 Factor3 Factor4 Factor5
N1  0.816
N2  0.787
N3  0.714
N4  0.562  -0.367
N5  0.518
E1         -0.587
E2         -0.674
E4          0.613           0.363
C1                  0.533
C2                  0.624
C3                  0.554
C4                 -0.653
C5                 -0.573
A2                          0.601
A3                          0.662
A5          0.351           0.580
O1                                  0.524
O3                                  0.614
O5                                 -0.512
A1                         -0.393
A4                          0.454
E3          0.490           0.315   0.313
E5          0.491   0.310
O2                                 -0.454
O4                                  0.368

               Factor1 Factor2 Factor3 Factor4 Factor5
SS loadings      2.687   2.320   2.034   1.978   1.557
Proportion Var   0.107   0.093   0.081   0.079   0.062
Cumulative Var   0.107   0.200   0.282   0.361   0.423
```

A estrutura agora é muito mais "simples" e alinhada com a teoria. Podemos visualizar essa transformação de forma clara comparando o círculo de correlações *antes* e *depois* da rotação.

Primeiro, a solução não rotacionada (Figura 14.2). Note como as variáveis (vetores) se espalham pelo espaço fatorial sem um padrão claro. É difícil traçar os eixos (fatores) de forma que representem grupos distintos de variáveis.

```r
library(ggrepel)

# Extrair cargas da solução NÃO ROTACIONADA (ml) para um dataframe
loadings_unrotated_df <- as.data.frame(unclass(fa_ml$loadings))
loadings_unrotated_df$Variable <- rownames(loadings_unrotated_df)

# Selecionar algumas variáveis para anotar e evitar poluição
vars_to_label <- c("N1", "N3", "E2", "E4", "A1", "C1", "O1")
annotations_df_unrotated <- loadings_unrotated_df %>% filter(Variable %in% vars_to_label)

# Criar dados para o círculo unitário
circle <- data.frame(
  angle = seq(-pi, pi, length = 100),
  x = sin(seq(-pi, pi, length = 100)),
  y = cos(seq(-pi, pi, length = 100))
)

# Gerar o gráfico com ggplot2
ggplot(data = loadings_unrotated_df, aes(x = ML1, y = ML2)) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "gray70") +
  geom_vline(xintercept = 0, linetype = "dashed", color = "gray70") +
  geom_path(data = circle, aes(x = x, y = y), inherit.aes = FALSE, color = "gray60") +
  geom_segment(aes(x = 0, y = 0, xend = ML1, yend = ML2),
               arrow = arrow(length = unit(0.1, "inches")),
               color = "steelblue") +
  geom_text_repel(data = annotations_df_unrotated, aes(label = Variable), min.segment.len
  coord_fixed(ratio = 1, xlim = c(-1.1, 1.1), ylim = c(-1.1, 1.1)) +
  labs(title = "Círculo de Correlações - Solução Não Rotacionada",
       x = "Fator 1",
       y = "Fator 2") +
  theme_minimal()
```

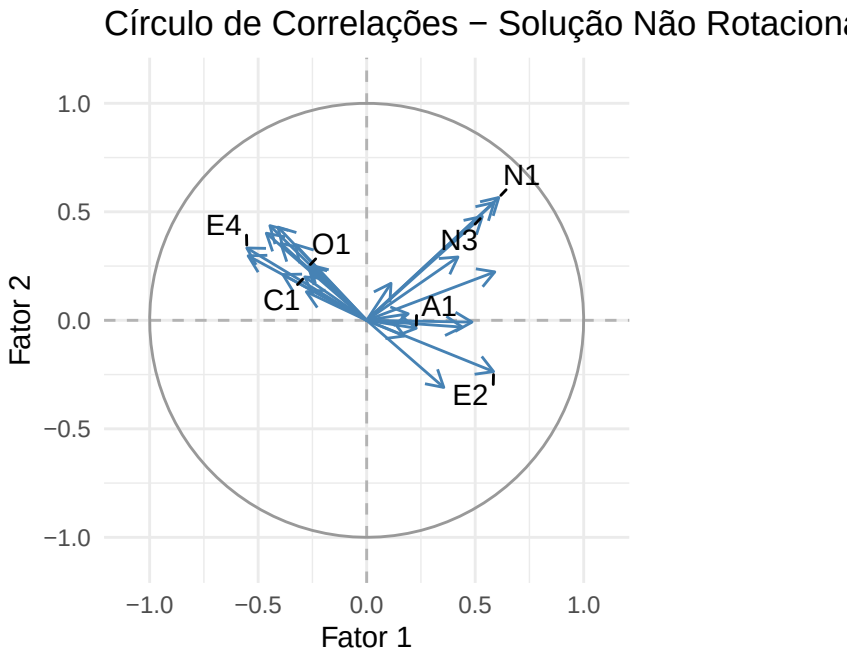

Figura 14.2: Círculo de correlações da solução não rotacionada. Os vetores não estão alinhados com os eixos.

Agora, veja o resultado após a rotação Varimax (Figura 14.3). A rotação funcionou como um ajuste dos eixos, alinhando-os com os agrupamentos de variáveis.

```
# Extrair cargas da solução ROTACIONADA (Varimax) para um dataframe
loadings_rotated_df <- as.data.frame(unclass(fa_varimax$loadings))
loadings_rotated_df$Variable <- rownames(loadings_rotated_df)

# Selecionar as mesmas variáveis para anotar
annotations_df_rotated <- loadings_rotated_df %>% filter(Variable %in% vars_to_label)

# Calcular a variância explicada para os eixos
ss_loadings <- colSums(fa_varimax$loadings2)
prop_variance <- ss_loadings / ncol(bfi_complete)
xlab_text <- sprintf("Fator 1 (%.2f%% da variância)", prop_variance[1] * 100)
ylab_text <- sprintf("Fator 2 (%.2f%% da variância)", prop_variance[2] * 100)

# Gerar o gráfico com ggplot2
ggplot(data = loadings_rotated_df, aes(x = Factor1, y = Factor2)) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "gray70") +
  geom_vline(xintercept = 0, linetype = "dashed", color = "gray70") +
```

```r
  geom_path(data = circle, aes(x = x, y = y), inherit.aes = FALSE, color = "gray60") +
  geom_segment(aes(x = 0, y = 0, xend = Factor1, yend = Factor2),
               arrow = arrow(length = unit(0.1, "inches")),
               color = "steelblue") +
  geom_text_repel(data = annotations_df_rotated, aes(label = Variable), min.segment.lengt
  coord_fixed(ratio = 1, xlim = c(-1.1, 1.1), ylim = c(-1.1, 1.1)) +
  labs(title = "Círculo de Correlações - Rotação Varimax",
       x = xlab_text,
       y = ylab_text) +
  theme_minimal()
```

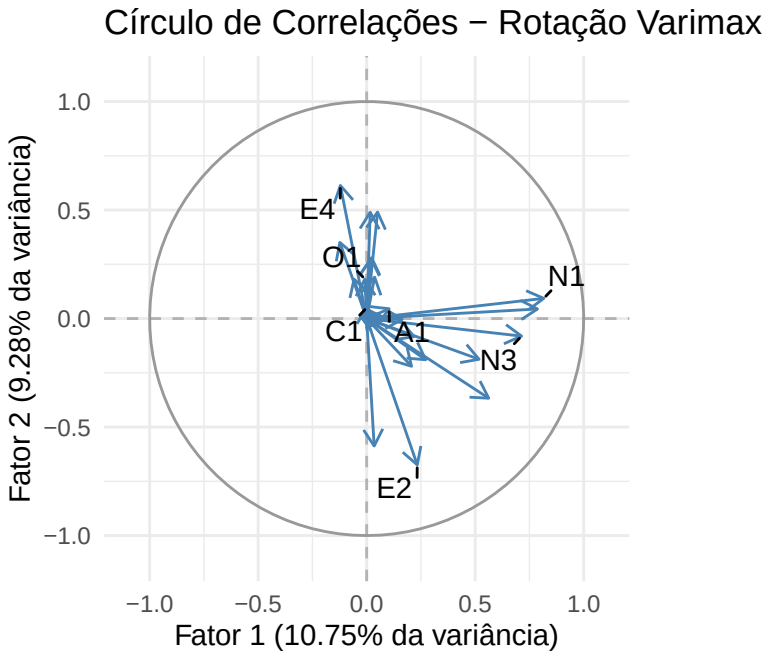

Figura 14.3: Círculo de correlações após a rotação Varimax. A rotação alinhou os vetores com os eixos, revelando uma estrutura simples.

O resultado é uma "estrutura simples", onde os itens de Neuroticismo (como N1 e N3) carregam quase exclusivamente no Fator 1 (correlação próxima de 1 ou -1 em um eixo e de 0 no outro), e os itens de Extroversão (como E2 e E4) carregam no Fator 2. A interpretação se torna direta.

14.4 Passo 4: Rotação Oblíqua (Promax)

A rotação **Promax** é mais flexível, pois **permite que os fatores sejam correlacionados**. Isso costuma ser mais realista em psicologia.

```
# Análise Fatorial com rotação Promax (oblíqua)
fa_promax <- fa(bfi_complete,
                nfactors = 5,
                rotate = "promax",
                fm = "pa")
```

```
Loading required namespace: GPArotation
```

```
cat("Cargas Fatoriais (Pattern Matrix) - Rotação Promax:\n")
```

```
Cargas Fatoriais (Pattern Matrix) - Rotação Promax:
```

```
print(fa_promax$loadings, cutoff = 0.3, sort = TRUE)
```

```

Loadings:
   PA2    PA1    PA3    PA5    PA4
N1  0.835
N2  0.791
N3  0.741
N4  0.533 -0.311
N5  0.529
E1        -0.636
E2        -0.711
E3         0.545
E4         0.660
C1                0.567
C2                0.697
C3                0.597
C4               -0.652
C5               -0.561
A2                       0.611
A3                       0.620
O3                              0.576
O5                             -0.543
```

```
A1                      -0.463
A4                       0.411
A5          0.332        0.489
E5          0.498
O1                               0.491
O2                              -0.484
O4                               0.370

                 PA2   PA1   PA3   PA5   PA4
SS loadings    2.704 2.486 2.050 1.638 1.461
Proportion Var 0.108 0.099 0.082 0.066 0.058
Cumulative Var 0.108 0.208 0.290 0.355 0.414
```

A matriz de cargas é similar à da Varimax, mas a grande vantagem é poder examinar a **matriz de correlação entre os fatores**.

```r
# Matriz de correlação entre os fatores
factor_correlations <- fa_promax$Phi

cat("Matriz de Correlação entre os Fatores (Promax):\n")
```

```
Matriz de Correlação entre os Fatores (Promax):
```

```r
print(round(factor_correlations, 2))
```

```
      PA2   PA1   PA3   PA5  PA4
PA2  1.00 -0.26 -0.22 -0.01 0.04
PA1 -0.26  1.00  0.40  0.35 0.14
PA3 -0.22  0.40  1.00  0.24 0.19
PA5 -0.01  0.35  0.24  1.00 0.16
PA4  0.04  0.14  0.19  0.16 1.00
```

```r
# Visualização da matriz de correlação
corrplot(factor_correlations, method = "color", type = "upper",
         addCoef.col = "black", tl.col = "black", tl.srt = 45, diag = FALSE)
```

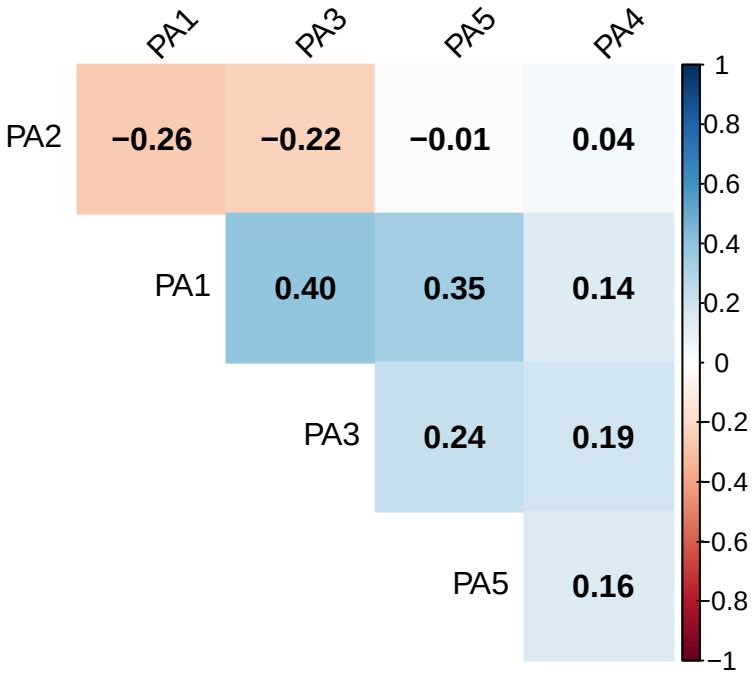

Figura 14.4: Correlações entre os fatores da solução Promax.

A matriz de correlação (Figura 14.4) mostra que **Neuroticismo (ML1)** se correlaciona negativamente com **Conscienciosidade (ML3)** (-0.33) e **Extroversão (ML2)** (-0.24). Essas relações são teoricamente plausíveis e seriam perdidas em uma rotação ortogonal.

14.5 Conclusão: Qual Rotação Escolher?

- **Solução Não Rotacionada:** É apenas um ponto de partida matemático. Dificilmente é possível tirar interpretações úteis dela.
- **Rotação Ortogonal (Varimax):** É a melhor escolha quando há fortes razões teóricas para acreditar que os fatores são independentes. Oferece uma solução mais simples (parcimoniosa).
- **Rotação Oblíqua (Promax):** É uma escolha mais realista nas ciências sociais. **A decisão final deve ser baseada na matriz de correlação dos fatores.** Se as correlações forem substanciais, a solução oblíqua é superior.

Neste exemplo, a solução **Promax (oblíqua)** é a mais apropriada. Ela não apenas recupera a estrutura dos Big Five, mas também fornece insights sobre como esses traços se relacionam, oferecendo uma visão mais rica e fiel da realidade psicológica.

15 Análise de agrupamentos

Neste exemplo, aplicaremos o procedimento de análise de agrupamentos proposto na Seção 9.3.2. O objetivo é ilustrar como a combinação de métodos hierárquicos e não hierárquicos pode levar a uma solução de agrupamento robusta e interpretável.

Utilizaremos o conjunto de dados `USArrests`, que contém estatísticas de crimes para cada um dos 50 estados dos EUA em 1973. As variáveis são:

- `Murder`: Assassinatos (por 100.000 habitantes).
- `Assault`: Agressões (por 100.000 habitantes).
- `UrbanPop`: Porcentagem da população que vive em áreas urbanas.
- `Rape`: Estupros (por 100.000 habitantes).

Nosso objetivo é agrupar os estados com base em seus perfis de criminalidade e urbanização.

15.1 Preparação dos Dados

O primeiro passo em qualquer análise de agrupamento baseada em distância é a **padronização** dos dados. As variáveis no nosso conjunto de dados têm escalas muito diferentes (`Assault` varia na casa das centenas, enquanto `Murder` varia na casa das dezenas). Se não padronizarmos, a variável `Assault` dominará o cálculo da distância, e o agrupamento será baseado quase inteiramente nela.

Padronizamos as variáveis para que tenham média 0 e desvio padrão 1.

15.1.1 Análise Descritiva

Antes de iniciar o agrupamento, é sempre útil explorar a distribuição das variáveis. A figura abaixo mostra os histogramas para cada uma das quatro variáveis do conjunto de dados.

```
fig, axes = plt.subplots(2, 2, figsize=(7, 5))

sns.histplot(data['Murder'], ax=axes[0, 0], kde=True)
axes[0, 0].set_title('Assassinatos')

sns.histplot(data['Assault'], ax=axes[0, 1], kde=True)
```

Tabela 15.1: Cabeçalho dos dados padronizados

```
import pandas as pd
from sklearn.preprocessing import StandardScaler
import statsmodels.api as sm
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import dendrogram, linkage
from scipy.cluster.hierarchy import fcluster
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
import seaborn as sns
from tabulate import tabulate

# Carregando o conjunto de dados
data = pd.read_csv('../../dados/USArrests.csv', index_col='rownames')

# Separando os dados e os nomes dos estados
X = data.values
states = data.index

# Padronizando os dados
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Criando um DataFrame com os dados padronizados para facilitar a manipulação
X_scaled_df = pd.DataFrame(X_scaled, index=states, columns=data.columns)

print(tabulate(X_scaled_df.head(), headers='keys', tablefmt='pipe'))
```

rownames	Murder	Assault	UrbanPop	Rape
Alabama	1.25518	0.790787	-0.526195	-0.00345116
Alaska	0.513019	1.11806	-1.22407	2.50942
Arizona	0.0723607	1.49382	1.00912	1.05347
Arkansas	0.234708	0.233212	-1.08449	-0.186794
California	0.281093	1.27564	1.77678	2.08881

```
axes[0, 1].set_title('Agressões')

sns.histplot(data['UrbanPop'], ax=axes[1, 0], kde=True)
axes[1, 0].set_title('População Urbana (%)')

sns.histplot(data['Rape'], ax=axes[1, 1], kde=True)
axes[1, 1].set_title('Estupros')

plt.tight_layout(rect=[0, 0.03, 1, 0.95])
plt.show()
```

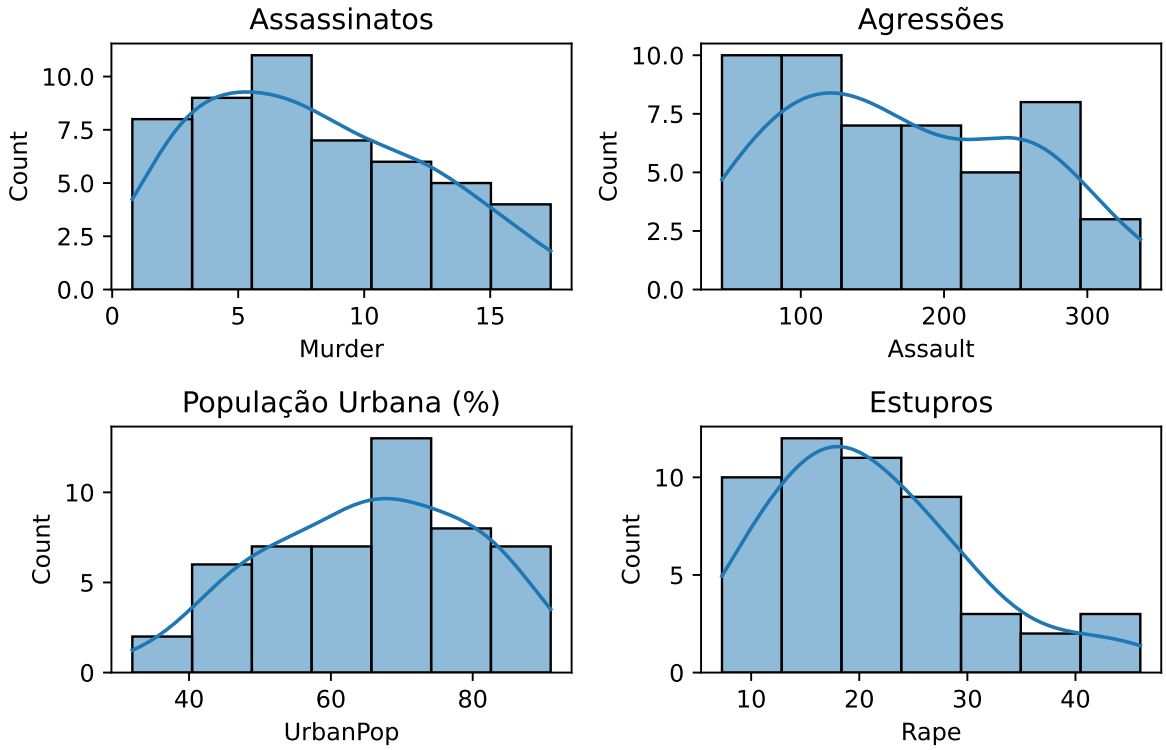

Figura 15.1: Distribuição das variáveis do dataset USArrests.

Observamos que as variáveis de crime (`Murder`, `Assault`, `Rape`) parecem ter uma leve assimetria à direita, com a maioria dos estados concentrados em valores mais baixos. A variável `UrbanPop` tem uma distribuição mais simétrica, quase uniforme, indicando uma boa variedade nos níveis de urbanização entre os estados.

Além dos histogramas, podemos visualizar a matriz de correlação entre as variáveis para entender suas relações lineares.

```
corr_matrix = data.corr()
plt.figure(figsize=(7, 5))
sns.heatmap(corr_matrix, annot=True, cmap='Greys', fmt='.2f')
plt.show()
```

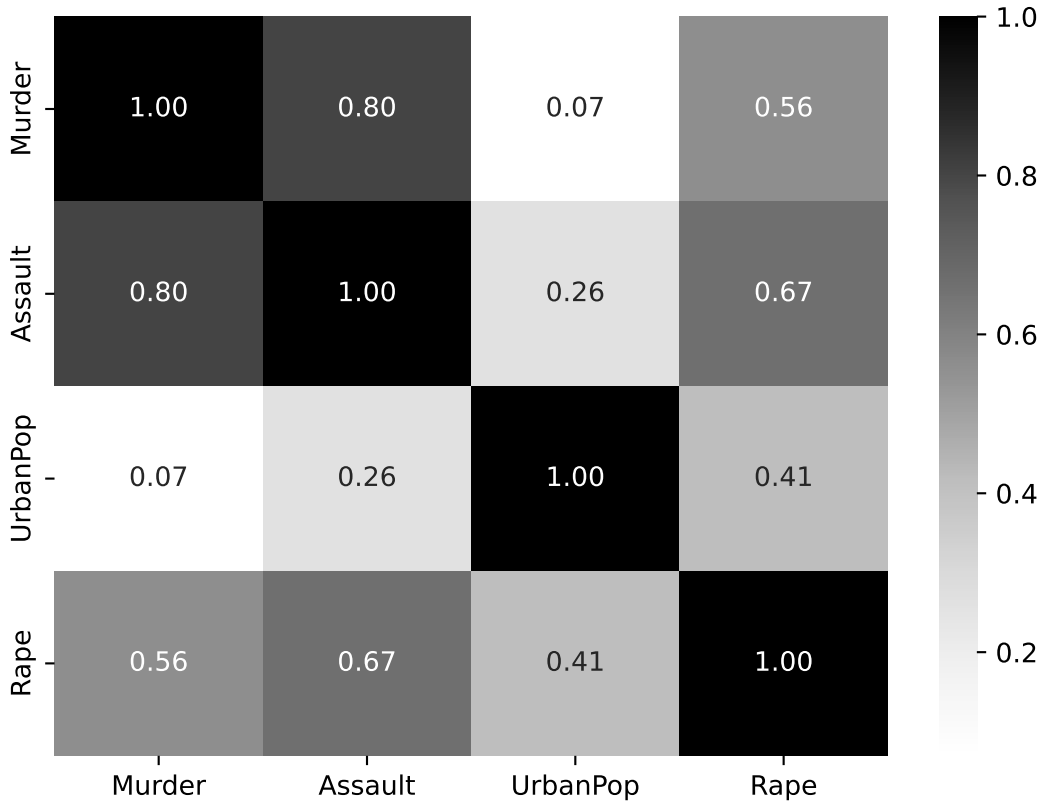

Figura 15.2: Matriz de Correlação entre as variáveis.

A matriz de correlação na Figura 20.1 mostra, como esperado, uma forte correlação positiva entre as três variáveis de crime (`Murder`, `Assault`, `Rape`). `UrbanPop` tem uma correlação positiva mais fraca com as outras variáveis, sugerindo que o efeito da urbanização no aumento da criminalidade geral é moderado.

15.2 Agrupamento Hierárquico e Escolha de K

Agora, aplicamos o agrupamento hierárquico aglomerativo usando o **método de Ward**, que busca minimizar a variância dentro dos grupos a cada fusão. Em seguida, plotamos o dendrograma para nos

ajudar a decidir o número ideal de grupos, K.

```
# Realizando o agrupamento hierárquico com o método de Ward
linked = linkage(X_scaled, method='ward')

# Plotando o dendrograma
plt.figure(figsize=(7, 5))
dendrogram(linked,
           orientation='top',
           labels=states,
           distance_sort='descending',
           show_leaf_counts=True,
           color_threshold=5.2)
plt.xlabel('Estados')
plt.ylabel('Distância de Ward')
plt.axhline(y=5.2, color='r', linestyle='-.', label="Corte 1 (K=4)")
plt.axhline(y=10.5, color='grey', linestyle='--', label="Corte 2 (K=2)")
plt.legend(loc="upper left")
plt.show()
```

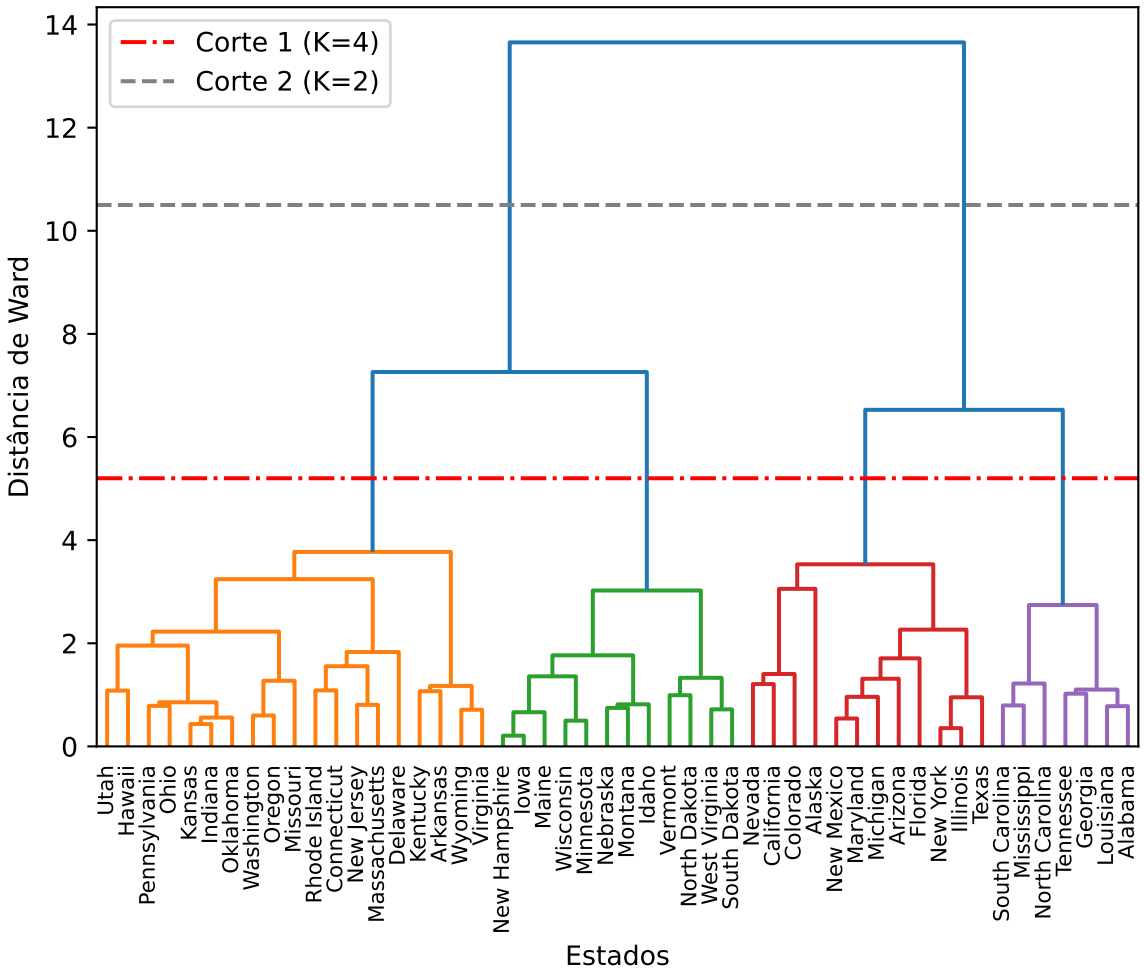

Figura 15.3: Dendrograma para o conjunto de dados USArrests usando o método de Ward.

Analisando o Figura 15.3, procuramos por um corte que cruze o maior espaço vertical possível. Vemos duas opções razoáveis, indicadas pelas linhas tracejadas. O "Corte 2" (cinza) sugere uma partição em $K = 2$ grupos, separando os estados em dois grandes blocos. O "Corte 1" (vermelho), mais abaixo, sugere uma partição mais granular de $K = 4$ grupos. Uma solução com 4 grupos nos dará um entendimento mais detalhado dos perfis dos estados. Portanto, iniciaremos a análise com $K = 4$ e, ao final deste exemplo, exploraremos a solução mais simples com $K = 2$ para fins de comparação.

Para verificar a robustez dessa escolha, podemos comparar o resultado com o de outro método de ligação, como a **ligação completa**.

```
linked_complete = linkage(X_scaled, method='complete')
```

```
plt.figure(figsize=(7, 5))
dendrogram(linked_complete,
           orientation='top',
           labels=states,
           distance_sort='descending',
           show_leaf_counts=True)
plt.xlabel('Estados')
plt.ylabel('Distância')
plt.show()
```

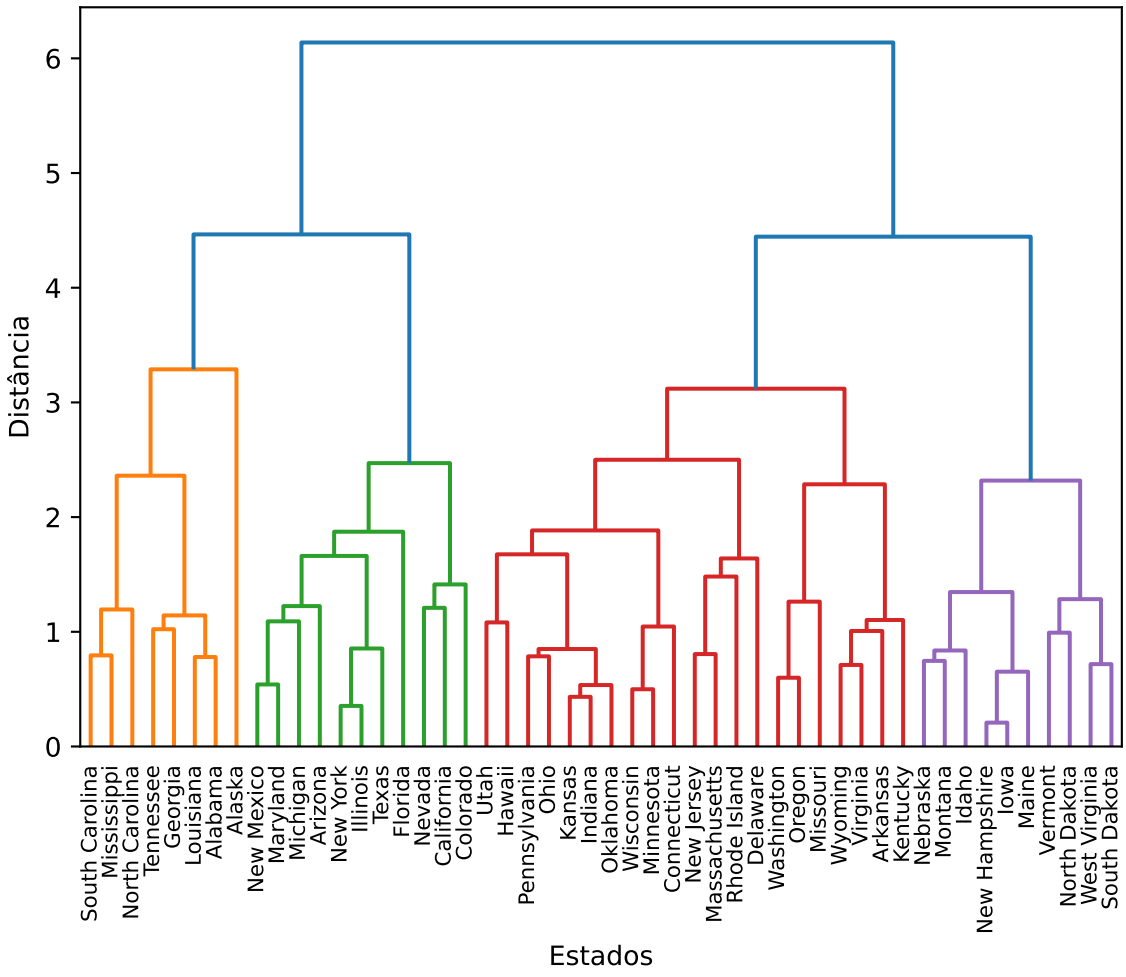

Figura 15.4: Dendrograma para o conjunto de dados USArrests usando o método de Ligação Completa.

O dendrograma de ligação completa também sugere uma partição de 2 ou 4 grupos como as mais

sensatas.

15.3 K-Médias com Centroides Hierárquicos

Seguindo o nosso procedimento, agora usaremos o resultado do agrupamento hierárquico para informar o algoritmo K-médias.

1. Obtemos as 4 partições (grupos) do método de Ward.
2. Calculamos o centroide (média) de cada um desses 4 grupos.
3. Executamos o K-médias com $K = 4$, usando os centroides calculados como pontos de partida.

Isso ajuda o K-médias a evitar mínimos locais e a convergir para uma solução mais estável e significativa.

```
# 1. Obter os 4 grupos do modelo hierárquico
k = 4
hierarchical_grupos = fcluster(linked, k, criterion='maxclust')

# Adicionar ao DataFrame para calcular os centroides
X_scaled_df['hierarchical_grupo'] = hierarchical_grupos

# 2. Calcular os centroides iniciais
initial_centroids = X_scaled_df.groupby('hierarchical_grupo').mean().values

# 3. Executar o K-médias com os centroides iniciais
kmeans = KMeans(n_clusters=k, init=initial_centroids, n_init=1, random_state=42)
kmeans.fit(X_scaled_df.drop('hierarchical_grupo', axis=1))

# Obter os grupos finais
final_grupos = kmeans.labels_

# Adicionar os grupos finais ao DataFrame original (não padronizado)
data['grupo'] = final_grupos
```

15.4 Interpretação e Visualização dos Grupos

Com os grupos finais definidos, o passo mais importante é a interpretação. Calculamos a média de cada variável para cada grupo para criar um "perfil".

A Tabela 15.2 nos permite caracterizar cada grupo:

Tabela 15.2: Perfil dos grupos: médias das variáveis para cada grupo.

```
# Calcular as médias por grupo
grupo_profile = data.groupby('grupo').mean()
print(tabulate(grupo_profile, headers='keys', tablefmt='pipe'))
```

grupo	Murder	Assault	UrbanPop	Rape
0	13.9375	243.625	53.75	21.4125
1	10.9667	264	76.5	33.6083
2	3.6	78.5385	52.0769	12.1769
3	5.85294	141.176	73.6471	19.3353

- **Grupo 0 (Estados Perigosos):** Este grupo tem os maiores índices de assassinatos e índices também altos de agressões e estupros. A população urbana é uma das mais baixas. Podemos nomeá-lo "Estados Violentos e Rurais".
- **Grupo 1 (Estados Urbanizados e Perigosos):** Este grupo tem alta urbanização e níveis de criminalidade também muito altos, especialmente quanto a estupros e agressões. Um bom nome seria "Grandes Centros Urbanos Perigosos".
- **Grupo 2 (Estados Seguros e Rurais):** Este grupo se destaca bastante dos anteriores. Apresenta os menores índices em todas as categorias de crime. A população urbana também é a mais baixa. Inclui estados como Dakota do Norte, Vermont e Iowa. Poderíamos chamá-lo de "Estados Seguros e Rurais".
- **Grupo 3 (Estados Intermediários):** Este grupo é formado por estados urbanizados com menores índices de criminalidade, quando comparados aos grupos 0 e 1. Nele, nenhum extremo se destaca. Podemos chamá-lo de "Estados na Média".

Para visualizar a separação, usamos a Análise de Componentes Principais (ACP) para reduzir a dimensionalidade dos dados para 2D e plotamos os estados, colorindo-os por grupo.

15.4.1 Interpretando os Componentes Principais

Os eixos (componentes principais) são combinações lineares das variáveis originais. Podemos inspecionar os pesos (loadings) de cada variável para interpretar o significado de cada componente.

A tabela Tabela 15.3 mostra as cargas das variáveis nos dois primeiros fatores.

O primeiro componente (PC1) explica aproximadamente 62% da variância total, enquanto o segundo (PC2) explica cerca de 25%. Juntos, eles capturam 87% da informação original, o que é excelente para uma visualização 2D.

Tabela 15.3: Cargas (loadings) dos componentes principais.

```
# Reduzindo a dimensionalidade com ACP
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)

# Variância explicada
explained_variance = pca.explained_variance_ratio_

# Loadings
loadings = pca.components_.T
loadings_df = pd.DataFrame(loadings, columns=['PC1', 'PC2'], index=data.columns[:-1])
print(tabulate(loadings_df, headers='keys', tablefmt='pipe'))
```

	PC1	PC2
Murder	0.535899	-0.418181
Assault	0.583184	-0.187986
UrbanPop	0.278191	0.872806
Rape	0.543432	0.167319

- **Componente Principal 1 (PC1):** Todas as quatro variáveis têm cargas positivas, com destaque para as variáveis associadas à criminalidade. Isso significa que ele representa uma medida geral de "Criminalidade".
- **Componente Principal 2 (PC2):** Este componente mostra um contraste. Ele tem uma carga positiva forte para `UrbanPop` e uma carga negativa para `Murder`. Isso significa que PC2 separa estados urbanizados com menor índice de assassinatos (scores altos) de estados rurais com mais assassinatos (scores baixos).

Com essa interpretação, podemos agora visualizar os grupos de forma mais informativa.

```
# Criando um DataFrame para o plot
pca_df = pd.DataFrame(data=X_pca, columns=['PC1', 'PC2'])
pca_df['grupo'] = final_grupos
pca_df['state'] = states

# Plotando
plt.figure(figsize=(7, 5))
sns.scatterplot(x='PC1', y='PC2', hue='grupo', data=pca_df, palette='viridis', s=100)

# Adicionando os nomes dos estados ao gráfico
for i in range(pca_df.shape[0]):
    plt.text(x=pca_df.PC1[i]+0.05, y=pca_df.PC2[i], s=pca_df.state[i],
             fontdict=dict(color='black',size=8))

plt.title('Grupos de Estados dos EUA (Visualização com ACP)')
plt.xlabel(f'Componente Principal 1 ({explained_variance[0]:.1%})')
plt.ylabel(f'Componente Principal 2 ({explained_variance[1]:.1%})')
plt.legend(title='Grupo')
plt.grid(True)
plt.show()
```

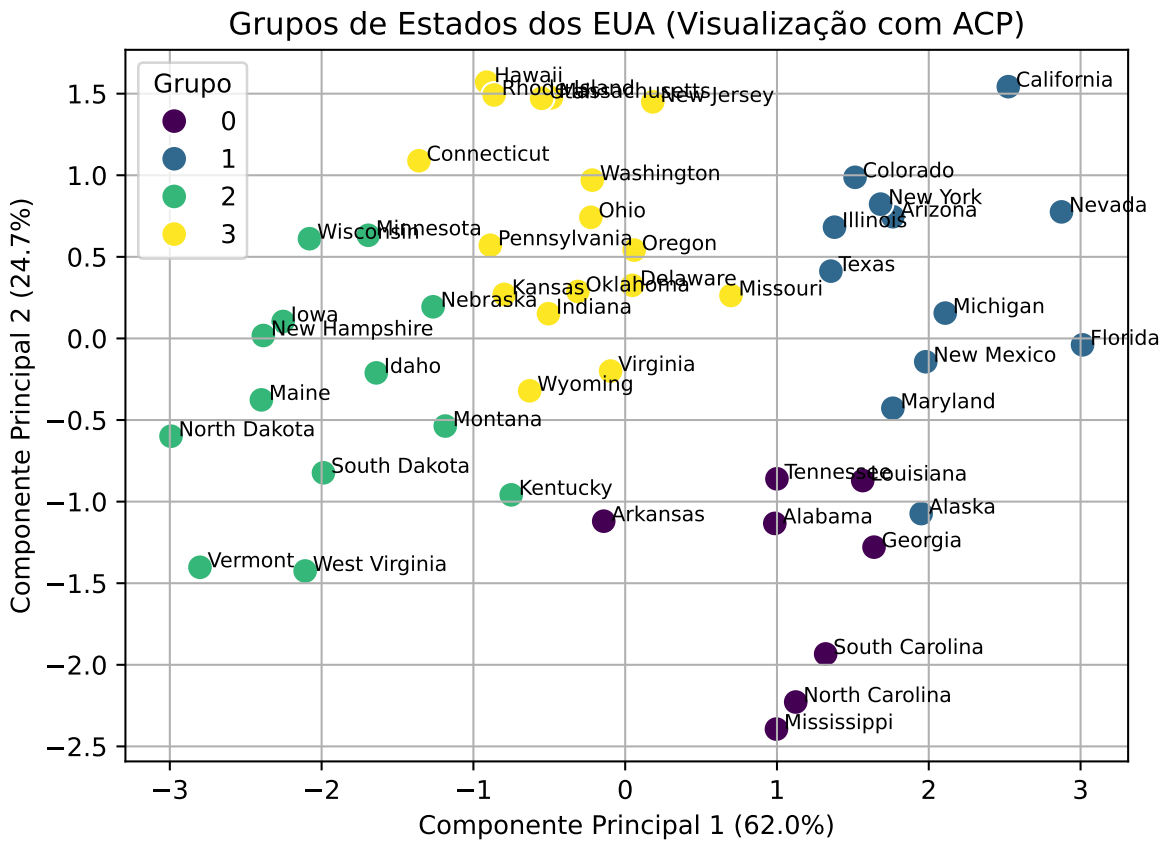

Figura 15.5: Visualização dos grupos no espaço dos dois primeiros componentes principais.

O gráfico na Figura 15.5 mostra uma separação clara dos grupos. O primeiro componente principal (PC1, eixo horizontal) efetivamente separa os estados com base na "Criminalidade Geral", com os grupos mais violentos (0 e 1) à direita e os mais seguros (2 e 3) à esquerda. O segundo componente principal (PC2, eixo vertical) está relacionado à "Urbanização", posicionando os grupos mais urbanizados (1 e 3) na parte superior e os mais rurais (0 e 2) na parte inferior. A análise combinada forneceu uma partição clara e interpretável dos estados dos EUA com base em seus dados sociais de 1973.

15.5 Comparação com K=2 Grupos

Como vimos no dendrograma, uma solução com $K = 2$ também é uma escolha justificável e representa a divisão de mais alto nível nos dados. Vamos repetir a etapa final do nosso procedimento para $K = 2$ e analisar o resultado.

Com $K = 2$, a partição é bem mais simples:

Tabela 15.4: Perfil dos grupos para a solução com K=2.

```
# 1. Obter os 2 grupos do modelo hierárquico
k = 2
hierarchical_grupos_k2 = fcluster(linked, k, criterion='maxclust')

# Adicionar ao DataFrame para calcular os centroides
X_scaled_df['hierarchical_grupo_k2'] = hierarchical_grupos_k2
initial_centroids_k2 = X_scaled_df.drop(columns="hierarchical_grupo").groupby('hierarchic

# 2. Executar K-médias
kmeans_k2 = KMeans(n_clusters=k, init=initial_centroids_k2, n_init=1, random_state=42)
kmeans_k2.fit(X_scaled_df.drop(['hierarchical_grupo', 'hierarchical_grupo_k2'], axis=1))
final_grupos_k2 = kmeans_k2.labels_

# 3. Calcular e exibir o perfil
data['grupo_k2'] = final_grupos_k2
grupo_profile_k2 = data.groupby('grupo_k2').mean().drop('grupo', axis=1)
print(tabulate(grupo_profile_k2, headers='keys', tablefmt='pipe'))
```

grupo_k2	Murder	Assault	UrbanPop	Rape
0	12.165	255.25	68.4	29.165
1	4.87	114.433	63.6333	15.9433

- **Grupo 0:** Agrega os estados que possuem níveis de criminalidade e urbanização mais elevados.
- **Grupo 1:** Engloba os estados mais seguros e rurais.

Essa divisão é útil para uma visão macro, mas perde a granularidade que a solução com 4 grupos nos proporcionou, como a distinção entre os estados "intermediários" e os "grandes centros urbanos". A visualização no espaço dos componentes principais ilustra isso claramente.

```
pca_df['grupo_k2'] = final_grupos_k2

plt.figure(figsize=(7, 5))
sns.scatterplot(x='PC1', y='PC2', hue='grupo_k2', data=pca_df, palette='viridis', s=100)

# Adicionando os nomes dos estados ao gráfico
for i in range(pca_df.shape[0]):
    plt.text(x=pca_df.PC1[i]+0.05, y=pca_df.PC2[i], s=pca_df.state[i],
             fontdict=dict(color='black',size=8))

plt.title('Grupos de Estados dos EUA (K=2, Visualização com ACP)')
plt.xlabel(f'Componente Principal 1 ({explained_variance[0]:.1%})')
plt.ylabel(f'Componente Principal 2 ({explained_variance[1]:.1%})')
plt.legend(title='Grupo')
plt.grid(True)
plt.show()
```

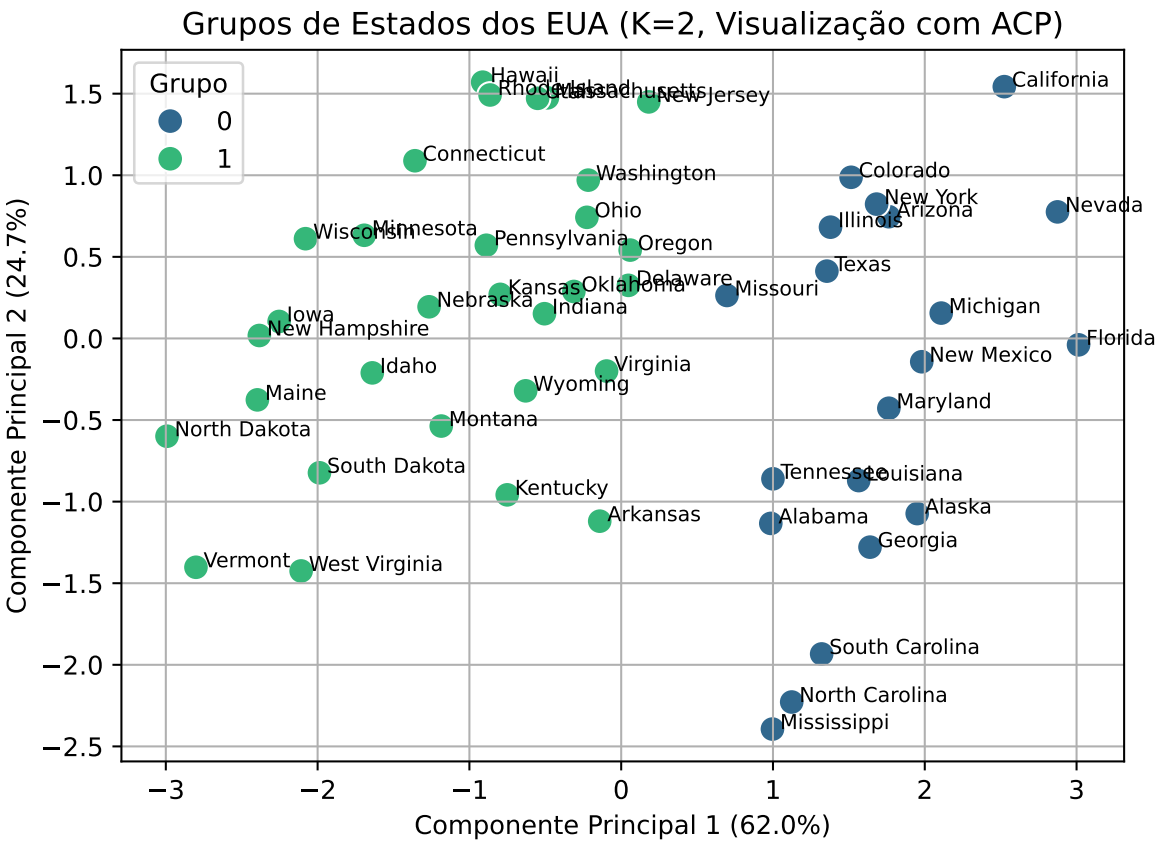

Figura 15.6: Visualização dos grupos (K=2) no espaço dos componentes principais.

16 Agrupamento com Modelos de Mistura Gaussiana (GMM)

Dando sequência à análise de agrupamentos do conjunto de dados `USArrests` iniciada na Capítulo 15, vamos agora aplicar uma abordagem baseada em modelos: o **Modelo de Mistura Gaussiana (GMM)**. Enquanto o exemplo anterior utilizou métodos hierárquicos e K-médias, aqui exploraremos as vantagens de uma abordagem probabilística.

Além disso, mostraremos passo a passo como o algoritmo de *Expectation-Maximization* (EM) ajusta iterativamente os parâmetros das distribuições Gaussianas para se adequar aos dados.

16.1 Preparação dos Dados

A preparação dos dados é a mesma do exemplo anterior. Carregamos o conjunto de dados `USArrests` e padronizamos as variáveis para que tenham média 0 e desvio padrão 1.

16.2 Visualizando o Algoritmo EM

Com base na análise exploratória do exemplo anterior (Capítulo 15), que sugeriu uma estrutura de 4 grupos, vamos configurar nosso GMM com $K = 4$ e visualizar como o algoritmo EM funciona.

Primeiro, vamos definir uma função auxiliar para desenhar as elipses que representam as distribuições de probabilidade de cada componente Gaussiano e reduzir a dimensionalidade dos dados para 2D usando a Análise de Componentes Principais (ACP), apenas para fins de plotagem.

```
def draw_ellipse(position, covariance, ax=None, **kwargs):
    """Desenha uma elipse representando a covariância de um componente GMM."""
    ax = ax or plt.gca()

    # Decomposição da matriz de covariância
    if covariance.shape == (2, 2):
        U, s, Vt = np.linalg.svd(covariance)
        angle = np.degrees(np.arctan2(U[1, 0], U[0, 0]))
        width, height = 2 * np.sqrt(s)
```

Tabela 16.1: Cabeçalho dos dados padronizados.

```
import pandas as pd
from sklearn.preprocessing import StandardScaler
import statsmodels.api as sm
import matplotlib.pyplot as plt
import matplotlib as mpl
from sklearn.mixture import GaussianMixture
from sklearn.decomposition import PCA
import seaborn as sns
import numpy as np
from tabulate import tabulate

# Carregando o conjunto de dados
data = pd.read_csv('../../dados/USArrests.csv', index_col='rownames')

# Padronizando os dados
scaler = StandardScaler()
X_scaled = scaler.fit_transform(data)
X_scaled_df = pd.DataFrame(X_scaled, index=data.index, columns=data.columns)

print(tabulate(X_scaled_df.head(), headers='keys', tablefmt='pipe'))
```

rownames	Murder	Assault	UrbanPop	Rape
Alabama	1.25518	0.790787	-0.526195	-0.00345116
Alaska	0.513019	1.11806	-1.22407	2.50942
Arizona	0.0723607	1.49382	1.00912	1.05347
Arkansas	0.234708	0.233212	-1.08449	-0.186794
California	0.281093	1.27564	1.77678	2.08881

```
    else:
        angle = 0
        width, height = 2 * np.sqrt(covariance)

    # Desenha a elipse
    for nsig in range(1, 4):
        ax.add_patch(mpl.patches.Ellipse(position, nsig * width, nsig * height,
                                         angle=angle, **kwargs))

# Reduzindo a dimensionalidade com ACP para visualização
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)
```

Agora, vamos observar o algoritmo EM em ação a partir de uma inicialização aleatória.

```
# Modelo GMM único com warm_start para visualização
gmm = GaussianMixture(n_components=4, covariance_type='full', n_init=1,
                      init_params='random', random_state=42, warm_start=True)

fig, axes = plt.subplots(2, 2, figsize=(7, 7), sharex=True, sharey=True)

# --- Iteração 0: Inicialização ---
gmm.max_iter = 0
gmm.fit(X_pca) # Apenas inicializa os parâmetros

axes[0, 0].scatter(X_pca[:, 0], X_pca[:, 1], s=15, cmap='viridis')
for pos, covar, w in zip(gmm.means_, gmm.covariances_, gmm.weights_):
    draw_ellipse(pos, covar, ax=axes[0, 0], alpha=0.2, fill=True, facecolor='gray')
axes[0, 0].set_title('Iteração 0: Inicialização Aleatória')

# --- Iteração 1 ---
gmm.max_iter = 1
gmm.fit(X_pca) # Executa 1 iteração EM

labels1 = gmm.predict(X_pca)
axes[0, 1].scatter(X_pca[:, 0], X_pca[:, 1], c=labels1, s=15, cmap='viridis')
for pos, covar, w in zip(gmm.means_, gmm.covariances_, gmm.weights_):
    draw_ellipse(pos, covar, ax=axes[0, 1], alpha=0.2, fill=True, facecolor='gray')
axes[0, 1].set_title('Iteração 1: Após 1 passo EM')

# --- Iteração 10 ---
gmm.max_iter = 9 # Executa mais 9 iterações (total 10)
```

```
gmm.fit(X_pca)

labels5 = gmm.predict(X_pca)
axes[1, 0].scatter(X_pca[:, 0], X_pca[:, 1], c=labels5, s=15, cmap='viridis')
for pos, covar, w in zip(gmm.means_, gmm.covariances_, gmm.weights_):
    draw_ellipse(pos, covar, ax=axes[1, 0], alpha=0.2, fill=True, facecolor='gray')
axes[1, 0].set_title('Iteração 10')

# --- Convergência Final ---
gmm.max_iter = 95 # Executa mais 95 iterações (total 100)
gmm.fit(X_pca)
gmm_final_random = gmm # Guarda o modelo final da inicialização aleatória

labels_final = gmm_final_random.predict(X_pca)
axes[1, 1].scatter(X_pca[:, 0], X_pca[:, 1], c=labels_final, s=15, cmap='viridis')
for pos, covar, w in zip(gmm_final_random.means_, gmm_final_random.covariances_, gmm_fina
    draw_ellipse(pos, covar, ax=axes[1, 1], alpha=0.2, fill=True, facecolor='gray')
axes[1, 1].set_title('Convergência Final')

for ax in axes.flat:
    ax.set_xticks([])
    ax.set_yticks([])

plt.tight_layout()
plt.show()
```

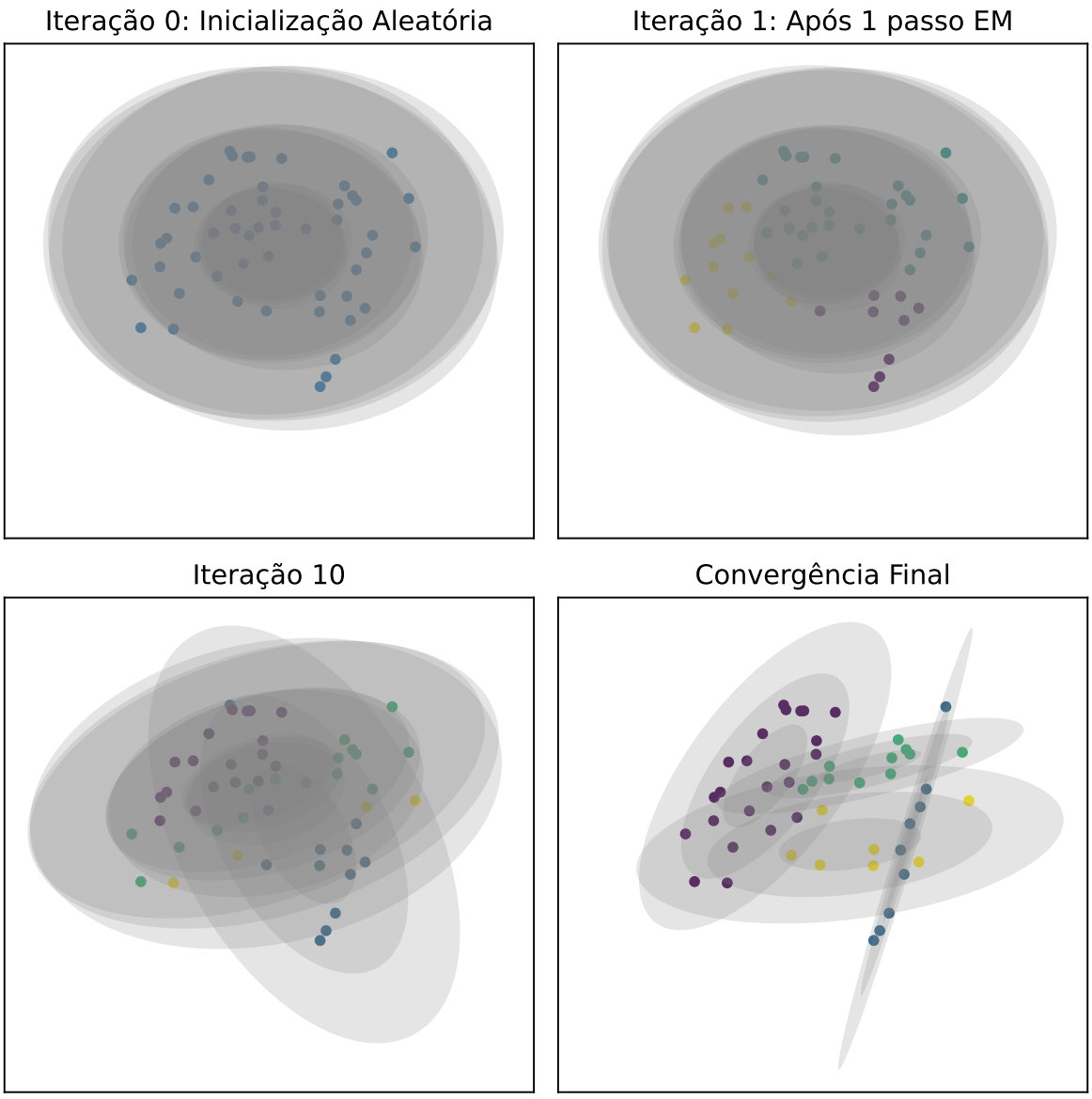

Figura 16.1: Visualização das iterações do algoritmo EM partindo de uma inicialização aleatória.

A Figura 16.1 ilustra o processo de ajuste do GMM a partir de uma inicialização aleatória, mostrando como as elipses se ajustam aos dados a cada iteração.

16.3 Efeito da Inicialização

A inicialização do algoritmo EM pode impactar tanto a velocidade de convergência quanto a solução final encontrada. Uma inicialização pobre pode levar o algoritmo a um mínimo local subótimo. Vamos comparar a solução convergida da nossa inicialização aleatória com uma inicialização informada, usando os centroides do K-médias, que é o padrão do `sklearn`.

```
# Modelo GMM com inicialização K-médias
gmm_kmeans_init = GaussianMixture(n_components=4, covariance_type='full', n_init=1,
                                  init_params='kmeans', random_state=42)
gmm_kmeans_init.fit(X_pca)

# --- Plot de Comparação ---
fig, axes = plt.subplots(1, 2, figsize=(7, 5), sharex=True, sharey=True)

# Plot Esquerda: Inicialização Aleatória (resultado do gmm_final_random)
labels_random = gmm_final_random.predict(X_pca)
axes[0].scatter(X_pca[:, 0], X_pca[:, 1], c=labels_random, s=15, cmap='viridis')
for pos, covar, w in zip(gmm_final_random.means_, gmm_final_random.covariances_, gmm_fina
    draw_ellipse(pos, covar, ax=axes[0], alpha=0.2, fill=True, facecolor='gray')
axes[0].set_title('Final: Inicialização Aleatória')

# Plot Direita: Inicialização K-médias
labels_kmeans = gmm_kmeans_init.predict(X_pca)
axes[1].scatter(X_pca[:, 0], X_pca[:, 1], c=labels_kmeans, s=15, cmap='viridis')
for pos, covar, w in zip(gmm_kmeans_init.means_, gmm_kmeans_init.covariances_, gmm_kmeans
    draw_ellipse(pos, covar, ax=axes[1], alpha=0.2, fill=True, facecolor='gray')
axes[1].set_title('Final: Inicialização K-médias')

for ax in axes.flat:
    ax.set_xticks([])
    ax.set_yticks([])

plt.tight_layout()
plt.show()
```

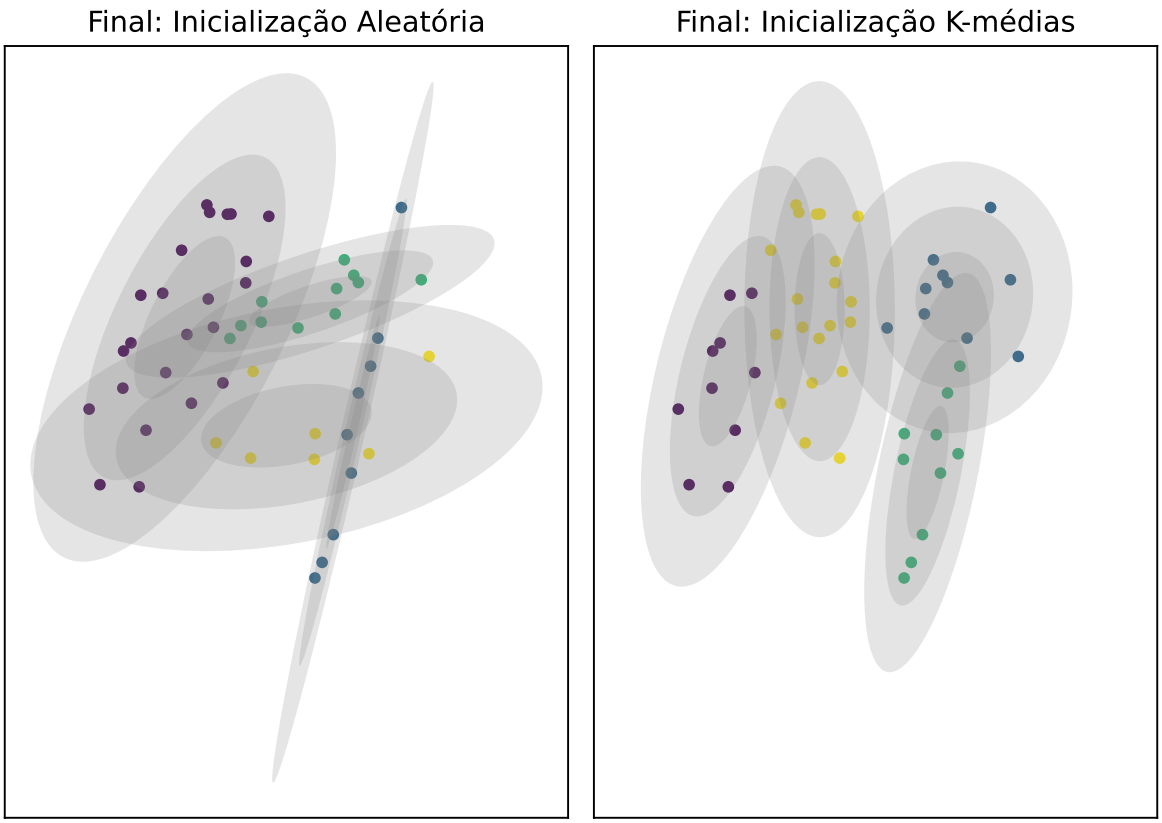

Figura 16.2: Comparação da convergência final do GMM com inicialização aleatória (esquerda) vs. K-médias (direita).

A Figura 16.2 revela como a escolha do método de inicialização pode impactar o resultado final do GMM. A inicialização aleatória (esquerda) e a inicialização baseada em K-médias (direita) levaram a agrupamentos visivelmente distintos. Isso ocorre porque o algoritmo EM, sensível às condições iniciais, pode convergir para um ótimo local em vez do ótimo global. A inicialização com K-médias, que posiciona os centróides iniciais de forma mais informada, tende a produzir soluções mais estáveis e reprodutíveis. Por essa razão, e para manter a consistência com a análise do capítulo anterior, usaremos o resultado do modelo inicializado com K-médias para a interpretação dos grupos.

16.4 Interpretação dos Grupos

Agora, com o modelo estimado utilizando o K-médias para inicialização, podemos analisar os grupos no espaço dos componentes principais. Note que os grupos são muito similares aos observados na Figura 15.5 do exemplo anterior. Por isso vamos omitir aqui a análise de perfis.

```
# Usando o modelo com inicialização k-médias para os grupos finais
data['gmm_grupo'] = gmm_kmeans_init.predict(X_pca)

# Criando um DataFrame para o plot
pca_df = pd.DataFrame(data=X_pca, columns=['PC1', 'PC2'])
pca_df['gmm_grupo'] = data['gmm_grupo'].values
pca_df['state'] = data.index

# Plotando
plt.figure(figsize=(7, 5))
sns.scatterplot(x='PC1', y='PC2', hue='gmm_grupo', data=pca_df, palette='viridis', s=100)

# Adicionando os nomes dos estados ao gráfico
for i in range(pca_df.shape[0]):
    plt.text(x=pca_df.PC1[i]+0.05, y=pca_df.PC2[i], s=pca_df.state[i],
             fontdict=dict(color='black',size=8))

# Interpretando os eixos da ACP
explained_variance = pca.explained_variance_ratio_
plt.title('Grupos GMM de Estados dos EUA (Visualização com ACP)')
plt.xlabel(f'Componente Principal 1 (Criminalidade Geral) ({explained_variance[0]:.1%})')
plt.ylabel(f'Componente Principal 2 (Urbanização) ({explained_variance[1]:.1%})')
plt.legend(title='Grupo')
plt.grid(True)
plt.show()
```

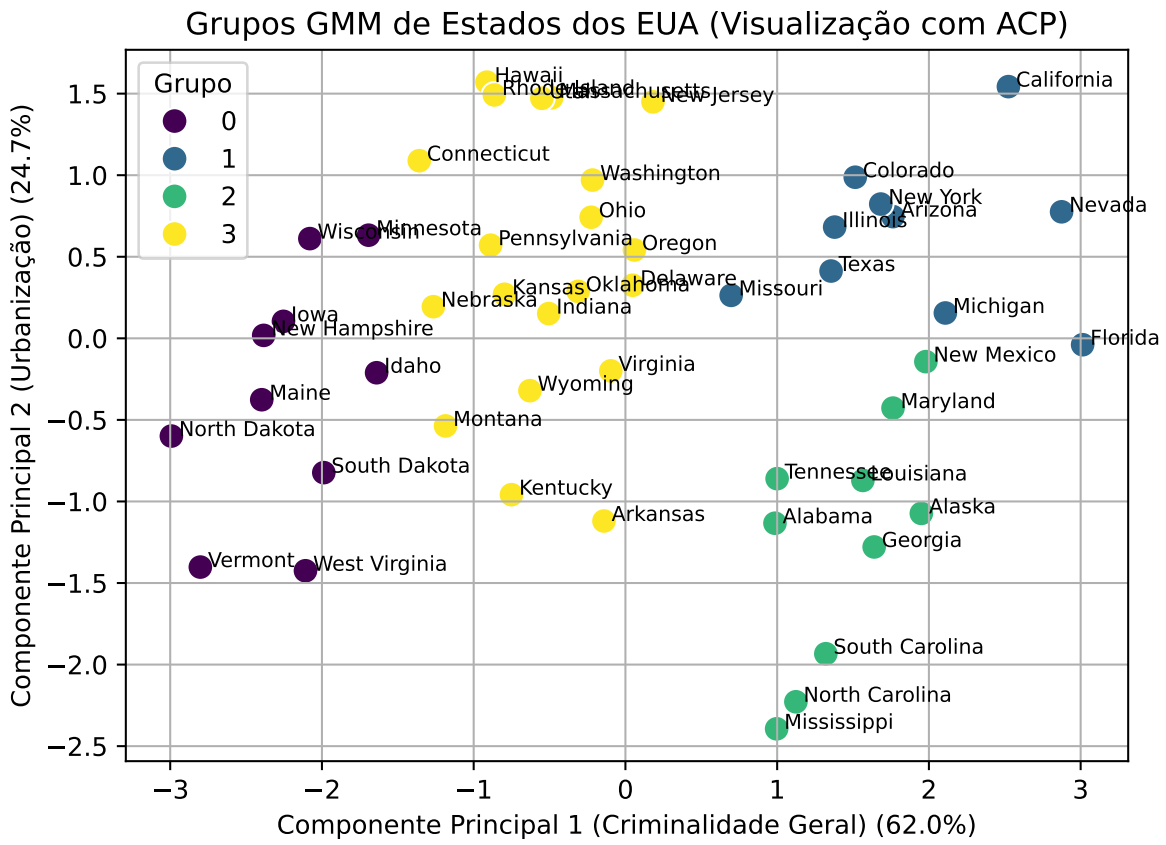

Figura 16.3: Visualização dos grupos GMM (inicialização K-médias) no espaço dos componentes principais.

16.5 Alocação Probabilística

Uma diferença fundamental entre o K-médias e o GMM é que o K-médias realiza uma alocação "rígida", onde cada ponto de dado pertence exclusivamente a um único grupo. Em contraste, o GMM realiza uma alocação "suave". Na figura anterior, ignoramos essa propriedade e simplesmente alocamos cada estado ao grupo de maior probabilidade.

Essa alocação suave significa que, para cada ponto, o GMM calcula a probabilidade de ele pertencer a *cada um* dos grupos. Essa abordagem nos dá uma medida de incerteza sobre a classificação de cada ponto. Um ponto equidistante de dois centros de grupo pode ter uma probabilidade de 50% para cada um, enquanto um ponto no coração de um grupo terá uma probabilidade próxima de 100% para aquele grupo.

Tabela 16.2: Probabilidades de pertencimento a cada grupo para estados selecionados.

```
# Obter as probabilidades de pertencimento
probs = gmm_kmeans_init.predict_proba(X_pca)

# Criar um DataFrame com as probabilidades
prob_df = pd.DataFrame(probs, index=data.index, columns=[f'Grupo {i}' for i in range(gmm_

# Selecionar estados de interesse
estados_selecionados = prob_df.loc[['California', 'New Mexico', 'Missouri']]

# Imprimir a tabela formatada
print(tabulate(estados_selecionados, headers='keys', tablefmt='pipe', floatfmt=".4f"))
```

rownames	Grupo 0	Grupo 1	Grupo 2	Grupo 3
California	0.0000	0.9961	0.0039	0.0000
New Mexico	0.0000	0.4966	0.5034	0.0000
Missouri	0.0000	0.8131	0.0002	0.1867

Vamos inspecionar as probabilidades de alguns estados específicos para ver isso em ação. Selecionamos estratégicamente para fins didáticos os estados da Califórnia, Novo México e Missouri.

A Tabela 16.2 mostra resultados interessantes. Notamos visualmente pela Figura 16.3 que a Califórnia é um estado bastante distante dos grupos 0, 2 e 3. Isso se reflete na probabilidade muito próxima de 1 de pertencimento ao grupo 1 para esse estado.

Já os estados do Missouri e do Novo México estão na "fronteira" entre dois grupos. Por isso, essas observações têm probabilidades de pertencimento distribuídas entre mais de um grupo. O caso mais extremo é o do Novo México, que tem probabilidades quase idênticas de pertencer aos grupos 1 e 2. Logo, podemos dizer que esse estado compartilha características com ambos os grupos.

16.6 Onde o K-médias Falha e o GMM Brilha

No exemplo anterior com os dados `USArrests`, vimos que os resultados do K-médias e do GMM foram bastante similares. Isso ocorreu porque os grupos naquele conjunto de dados são razoavelmente esféricos e bem definidos. Contudo, a verdadeira vantagem do GMM sobre o K-médias surge em cenários onde essa suposição de esfericidade não se sustenta.

A principal delas é a **flexibilidade de formato** do GMM: como cada componente Gaussiano possui sua própria matriz de covariância, o modelo pode se adaptar a grupos com diferentes formas (elípticas) e

orientações. Para ilustrar essa diferença, vamos criar um exemplo onde o K-médias falha precisamente por sua rigidez.

Vamos criar um exemplo clássico com dados alongados (anisotrópicos) para demonstrar onde o GMM, com sua capacidade de modelar covariâncias, supera o K-médias.

```
from sklearn.datasets import make_blobs
from sklearn.cluster import KMeans
from sklearn.metrics.cluster import contingency_matrix
from scipy.optimize import linear_sum_assignment

# Gerar dados isotrópicos
X, y_true = make_blobs(n_samples=400, centers=4,
                       cluster_std=0.60, random_state=0)
X = X[:, ::-1] # Trocar eixos para melhor visualização

# Transformar para criar grupos alongados e rotacionados
rng = np.random.RandomState(13)
transformation = rng.normal(size=(2, 2))
X_aniso = np.dot(X, transformation)

# Ajustar K-médias
kmeans = KMeans(n_clusters=4, random_state=42, n_init=10)
kmeans.fit(X_aniso)
y_kmeans = kmeans.predict(X_aniso)

# Ajustar GMM
gmm = GaussianMixture(n_components=4, covariance_type='full', random_state=42)
gmm.fit(X_aniso)
y_gmm = gmm.predict(X_aniso)

# --- Corrigir a ordem das cores para consistência ---
def remap_labels(y_true, y_pred):
    cm = contingency_matrix(y_true, y_pred)
    row_ind, col_ind = linear_sum_assignment(-cm)
    y_pred_remapped = np.zeros_like(y_pred)
    for i in range(y_pred.max() + 1):
        y_pred_remapped[y_pred == col_ind[i]] = row_ind[i]
    return y_pred_remapped

y_kmeans_remapped = remap_labels(y_true, y_kmeans)
y_gmm_remapped = remap_labels(y_true, y_gmm)
```

```
# --- Plotar os resultados ---
layout = [
    ['.', 'A', 'A', '.'],
    ['B', 'B', 'C', 'C']
]

fig, axs = plt.subplot_mosaic(layout, figsize=(8, 6), sharex=True, sharey=True)

# Plot A: Grupos Teóricos
axs['A'].scatter(X_aniso[:, 0], X_aniso[:, 1], c=y_true, s=20, cmap='viridis')
axs['A'].set_title('Grupos Teóricos')

# Plot B: K-médias
axs['B'].scatter(X_aniso[:, 0], X_aniso[:, 1], c=y_kmeans_remapped, s=20, cmap='viridis')
axs['B'].set_title('K-médias (Falha)')

# Plot C: GMM
axs['C'].scatter(X_aniso[:, 0], X_aniso[:, 1], c=y_gmm_remapped, s=20, cmap='viridis')
axs['C'].set_title('GMM (Sucesso)')
for pos, covar, w in zip(gmm.means_, gmm.covariances_, gmm.weights_):
    draw_ellipse(pos, covar, ax=axs['C'], alpha=0.2, fill=True, facecolor='gray')

# Remover ticks dos eixos
for ax_id in ['A', 'B', 'C']:
    axs[ax_id].set_xticks([])
    axs[ax_id].set_yticks([])

plt.tight_layout()
plt.show()
```

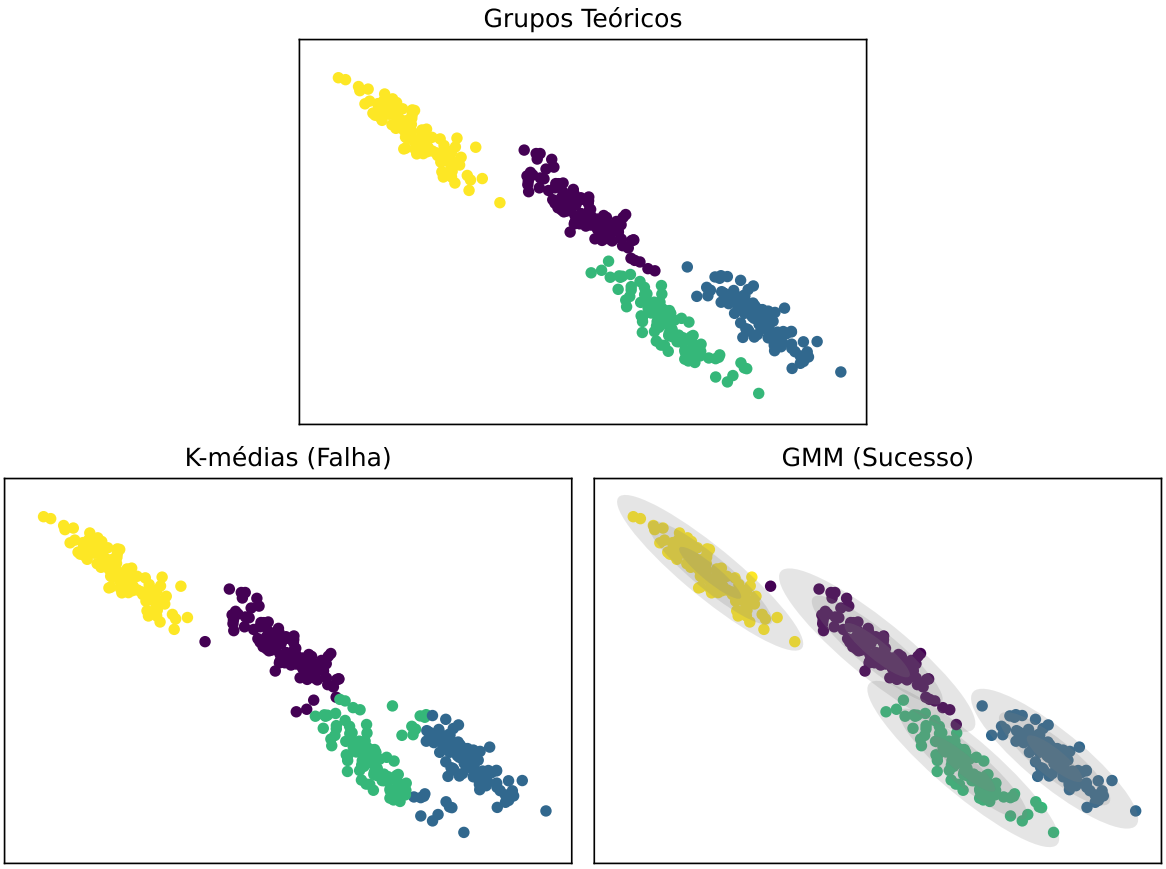

Figura 16.4: Comparação entre K-médias e GMM em grupos não esféricos.

A Figura 16.4 ilustra perfeitamente a diferença. O K-médias, buscando minimizar a distância para o centroide de maneira uniforme em todas as direções, divide os grupos alongados ao meio. Ele não consegue reconhecer que os pontos em um mesmo grupo alongado estão mais próximos "seguindo a forma" do grupo do que de um centroide de outro grupo.

Em contraste, o GMM tem sucesso. Cada componente Gaussiano tem sua própria matriz de covariância, o que permite ao modelo aprender a forma elíptica e a orientação de cada grupo. As elipses cinzas na figura da direita representam as distribuições de probabilidade aprendidas, que se alinham perfeitamente com a estrutura real dos dados. Este exemplo visual destaca a superioridade dos modelos probabilísticos como o GMM em cenários de agrupamento com estruturas de covariância mais complexas.

17 Análise Discriminante Exploratória

Neste exemplo, realizaremos uma Análise Discriminante com um dos conjuntos de dados mais clássicos da estatística: o dataset *Iris*, popularizado por R.A. Fisher em seu artigo de 1936, que introduziu a Análise Discriminante Linear.

O nosso foco será **exploratório**: queremos entender como as medidas das flores (variáveis) ajudam a distinguir as três espécies de íris (grupos). Em vez de focar na precisão da classificação de novas flores, vamos usar a LDA para encontrar as "visões" (projeções) que melhor separam os grupos e interpretar o que essas visões significam.

O conjunto de dados contém 150 observações de flores de íris, divididas igualmente em três espécies:

1. **Setosa**
2. **Versicolor**
3. **Virginica**

Para cada flor, quatro características foram medidas (em centímetros):

- Comprimento da sépala (`sepal_length`)
- Largura da sépala (`sepal_width`)
- Comprimento da pétala (`petal_length`)
- Largura da pétala (`petal_width`)

17.1 Análise Descritiva

Queremos entender a distribuição de cada variável e como elas se relacionam, especialmente quando observamos os diferentes grupos. Vamos carregar os dados e visualizar um resumo descritivo e um gráfico de pares.

```
# Gerar o gráfico de pares

sns.pairplot(iris, hue='especie', palette='viridis', height=2, aspect=1)
plt.show()
```

Tabela 17.1: Primeiras linhas do banco de dados Iris

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris

# Carregar o dataset Iris
iris_data = load_iris()
iris = pd.DataFrame(data=iris_data.data, columns=iris_data.feature_names)
iris['especie'] = iris_data.target_names[iris_data.target]

# Renomear colunas para o português, para facilitar a interpretação
iris.columns = ['comp_sepala', 'larg_sepala', 'comp_petala', 'larg_petala', 'especie']

# Exibir as primeiras linhas
print(iris.head())
```

```
   comp_sepala  larg_sepala  comp_petala  larg_petala especie
0          5.1          3.5          1.4          0.2  setosa
1          4.9          3.0          1.4          0.2  setosa
2          4.7          3.2          1.3          0.2  setosa
3          4.6          3.1          1.5          0.2  setosa
4          5.0          3.6          1.4          0.2  setosa
```

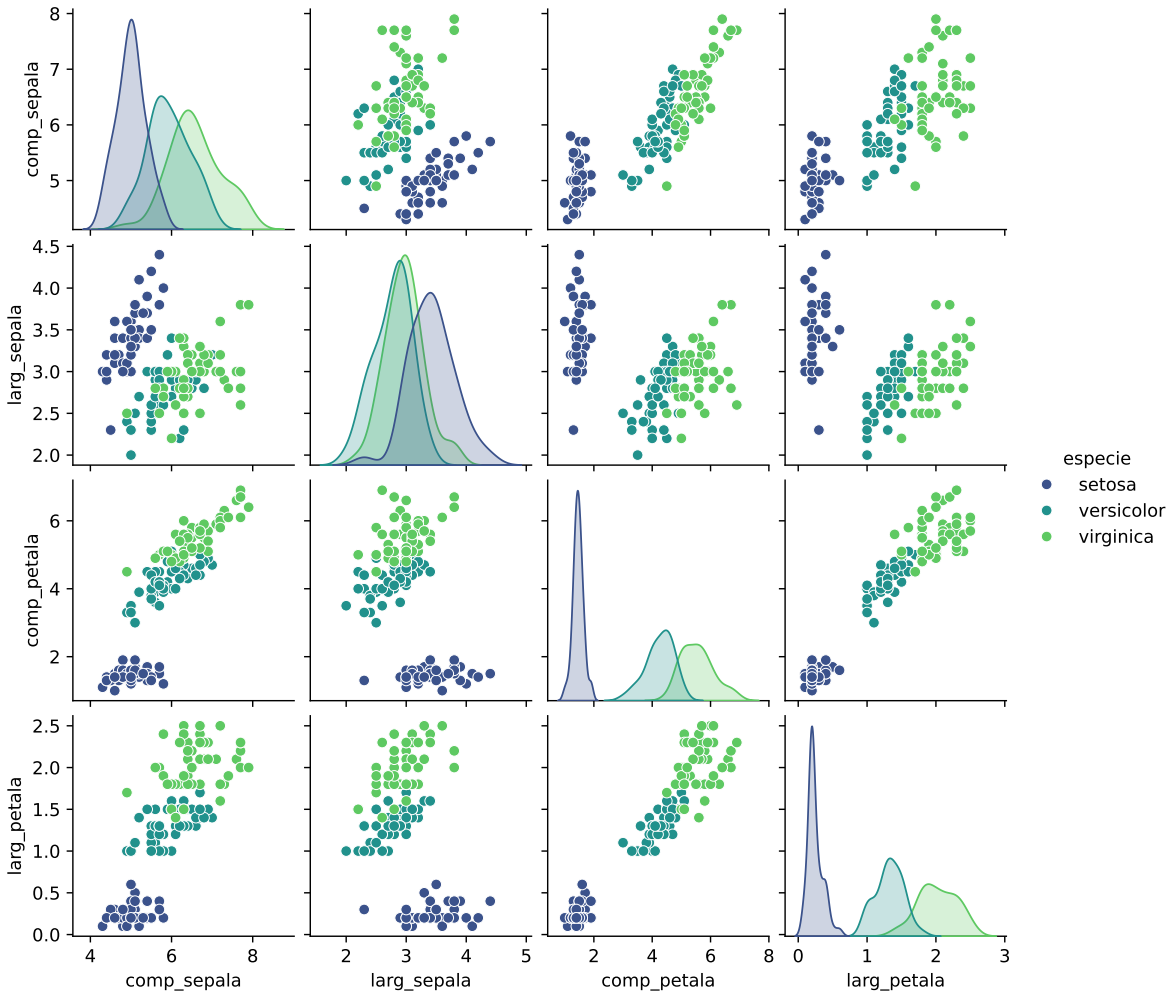

Figura 17.1: Gráfico de pares das variáveis do dataset Iris, separadas por espécie.

Na figura, observamos algumas características interessantes:

1. **Separação Clara da Setosa:** A espécie *Setosa* é facilmente distinguível das outras duas. Suas pétalas são muito menores (comprimento e largura) e suas sépalas são mais largas e mais curtas. A separação é tão clara que um simples corte em `comp_petala` ou `larg_petala` poderia classificar quase todas as *Setosas* perfeitamente.

2. **Sobreposição entre Versicolor e Virginica:** As espécies *Versicolor* e *Virginica* são mais parecidas. Embora a *Virginica* tenda a ter pétalas e sépalas maiores que a *Versicolor*, há uma sobreposição considerável em todas as variáveis. É aqui que uma técnica multivariada como a Análise Discriminante se torna útil, pois pode encontrar uma combinação de variáveis que melhore a separação.

3. **Correlações:** O comprimento e a largura da pétala (`comp_petala` e `larg_petala`) são altamente correlacionados. O comprimento da sépala (`comp_sepala`) também tem uma correlação positiva com ambas.

17.2 Análise Discriminante Linear (LDA)

A análise discriminante linear tem como suposição a homocedasticidade dos grupos. Ou seja, ela pressupõe que as matrizes de covariâncias de todos os grupos são iguais. A Figura 17.1 nos dá indícios de que tal suposição não é razoável. Em particular, a variância da largura e do comprimento da sépala do grupo *Setosa* parece bem inferior as demais. Ainda assim, vamos prosseguir com a LDA por razões didáticas.

A LDA busca encontrar as combinações lineares das variáveis originais — as **funções discriminantes** — que maximizam a razão entre a variância *entre* os grupos e a variância *dentro* dos grupos. Como temos $k = 3$ grupos e $p = 4$ variáveis, podemos encontrar no máximo $\min(p, k-1) = \min(4, 2) = 2$ funções discriminantes.

Essas duas funções, LD1 e LD2, criarão um novo espaço 2D onde a separação dos grupos é otimizada.

```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis

# Separar variáveis preditoras (X) e a variável resposta (y)
X = iris[['comp_sepala', 'larg_sepala', 'comp_petala', 'larg_petala']]
y = iris['especie']

# Criar e ajustar o modelo LDA
lda = LinearDiscriminantAnalysis(n_components=2)
X_lda = lda.fit_transform(X, y)

# Criar um DataFrame com os resultados
lda_df = pd.DataFrame(data=X_lda, columns=['LD1', 'LD2'])
lda_df['especie'] = y

# Plotar o resultado
plt.figure(figsize=(8, 6))
sns.scatterplot(x='LD1', y='LD2', hue='especie', data=lda_df, palette='viridis', s=70, al
plt.title('Projeção da Análise Discriminante Linear')
plt.xlabel('Função Discriminante 1 (LD1)')
plt.ylabel('Função Discriminante 2 (LD2)')
plt.grid(True, linestyle='--', alpha=0.5)
plt.legend()
plt.show()
```

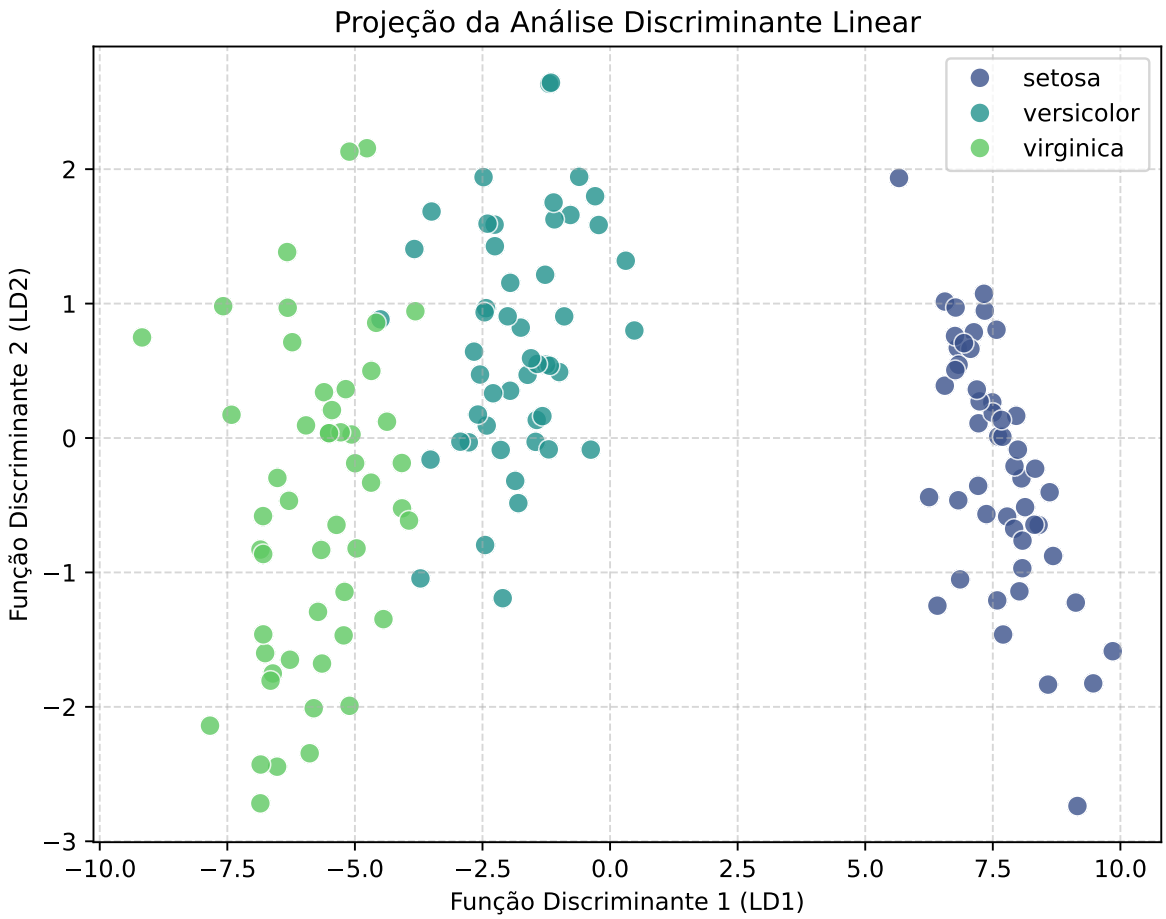

Figura 17.2: Projeção dos dados do Iris no espaço discriminante. Os eixos LD1 e LD2 são as combinações lineares que melhor separam as espécies.

O gráfico da projeção LDA é bem informativo:

- **LD1 (Eixo Horizontal):** Esta primeira função discriminante separa a *Setosa* das outras duas espécies. As flores da espécie *Setosa* aparecem agrupadas a direita, com uma grande lacuna os separando dos outros grupos. Vemos ainda que essa dimensão separa também os grupos *Versicolor* e *Virginica* relativamente bem.
- **LD2 (Eixo Vertical):** Esta segunda função discriminante acaba não sendo tão informativa, principalmente pelo fato da primeira dimensão já apresentar uma ótima separabilidade dos grupos. Ainda assim, podemos observar que as flores *Virginica* tendem a ter valores mais baixos em LD2, enquanto a *Versicolor* tem valores mais altos.

Juntas, as duas funções criam um mapa claro da estrutura dos grupos. Mas o que exatamente são LD1 e LD2? Para entender isso, precisamos olhar para os coeficientes (ou “loadings”) de cada função.

```
# Coeficientes das funções discriminantes
coeficientes = pd.DataFrame(lda.scalings_, index=X.columns, columns=['LD1', 'LD2'])
print("Coeficientes das Funções Discriminantes (Loadings):")
print(coeficientes)
```

```
Coeficientes das Funções Discriminantes (Loadings):
                  LD1       LD2
comp_sepala  0.829378 -0.024102
larg_sepala  1.534473 -2.164521
comp_petala -2.201212  0.931921
larg_petala -2.810460 -2.839188
```

Interpretando os Coeficientes:

- **LD1:** Os coeficientes de maior magnitude são `larg_petala` (-2.81) e `comp_petala` (-2.20), ambos negativos, que se contrapõem aos coeficientes positivos de `larg_sepala` (1.53) e `comp_sepala` (0.83). Isso significa que a **LD1 é uma medida de contraste entre o tamanho das pétalas e o tamanho das sépalas**. Escores baixos em LD1 (valores negativos) são associados a flores com pétalas grandes, enquanto escores altos (valores positivos) estão associados a flores com pétalas pequenas e sépalas largas. Olhando o gráfico, vemos que a *Setosa* tem escores altos em LD1, o que é consistente com suas pétalas pequenas. *Virginica* e *Versicolor* têm escores negativos, correspondendo às suas pétalas maiores.
- **LD2:** Esta função é dominada pelo contraste entre `larg_sepala` (-2.16) e `larg_petala` (-2.84), ambos com grandes coeficientes negativos, e `comp_petala` (0.93), com coeficiente positivo. Essencialmente, a **LD2 separa flores com base em uma "forma" que opõe o comprimento da pétala à sua largura e à largura da sépala**. Esta função é a principal responsável por separar a *Versicolor* (que tende a ter escores positivos em LD2) da *Virginica* (que tende a ter escores negativos), capturando as diferenças mais sutis entre essas duas espécies.

17.3 Comparação das Fronteiras de Decisão (LDA vs. QDA)

Enquanto a LDA assume que todos os grupos têm a mesma matriz de covariância (levando a fronteiras de decisão lineares), a **Análise Discriminante Quadrática (QDA)** relaxa essa suposição, permitindo fronteiras quadráticas mais flexíveis.

Vamos visualizar as fronteiras de decisão de ambos os modelos usando apenas duas variáveis: `comp_petala` e `larg_petala`. O objetivo disso é poder observar a diferença dos métodos visualmente de forma clara em duas dimensões. Ilustramos as regiões de discriminação na figura abaixo.

```
from sklearn.discriminant_analysis import QuadraticDiscriminantAnalysis
from matplotlib.colors import ListedColormap

# Usar apenas as duas variáveis mais importantes para visualização
X_petal = iris[['comp_petala', 'larg_petala']]
y_petal = iris['especie']

# Mapear espécies para números para o plot
species_map = {species: i for i, species in enumerate(y_petal.unique())}
y_numeric = y_petal.map(species_map)

# Ajustar modelos LDA e QDA
lda_petal = LinearDiscriminantAnalysis()
lda_petal.fit(X_petal, y_petal)

qda_petal = QuadraticDiscriminantAnalysis()
qda_petal.fit(X_petal, y_petal)

# Criar uma malha de pontos para plotar as fronteiras
x_min, x_max = X_petal.iloc[:, 0].min() - 1, X_petal.iloc[:, 0].max() + 1
y_min, y_max = X_petal.iloc[:, 1].min() - 1, X_petal.iloc[:, 1].max() + 1
xx, yy = np.meshgrid(np.arange(x_min, x_max, 0.02),
                     np.arange(y_min, y_max, 0.02))

# Criar um DataFrame com os pontos da malha para evitar o warning
grid_points = pd.DataFrame(np.c_[xx.ravel(), yy.ravel()], columns=X_petal.columns)

# Prever em cada ponto da malha
Z_lda = lda_petal.predict(grid_points)
Z_lda = np.array([species_map[label] for label in Z_lda]).reshape(xx.shape)

Z_qda = qda_petal.predict(grid_points)
Z_qda = np.array([species_map[label] for label in Z_qda]).reshape(xx.shape)

# Plotar
plt.figure(figsize=(8, 6))

# Plotar as áreas de decisão da QDA
cmap_light = ListedColormap(['#DAF7A6', '#FFC300', '#FF5733'])
plt.contourf(xx, yy, Z_qda, alpha=0.2, cmap=cmap_light)

# Plotar as fronteiras da LDA
```

```
plt.contour(xx, yy, Z_lda, colors='black', linestyles='--', linewidths=2)

# Plotar os pontos de dados
sns.scatterplot(x=X_petal.iloc[:, 0], y=X_petal.iloc[:, 1], hue=y_petal,
                palette='viridis', alpha=0.9, edgecolor='k', s=80)

plt.title('Fronteiras de Decisão: LDA (tracejado) vs. QDA (fundo colorido)')
plt.xlabel('Comprimento da Pétala (cm)')
plt.ylabel('Largura da Pétala (cm)')
plt.legend(title='Espécie')
plt.show()
```

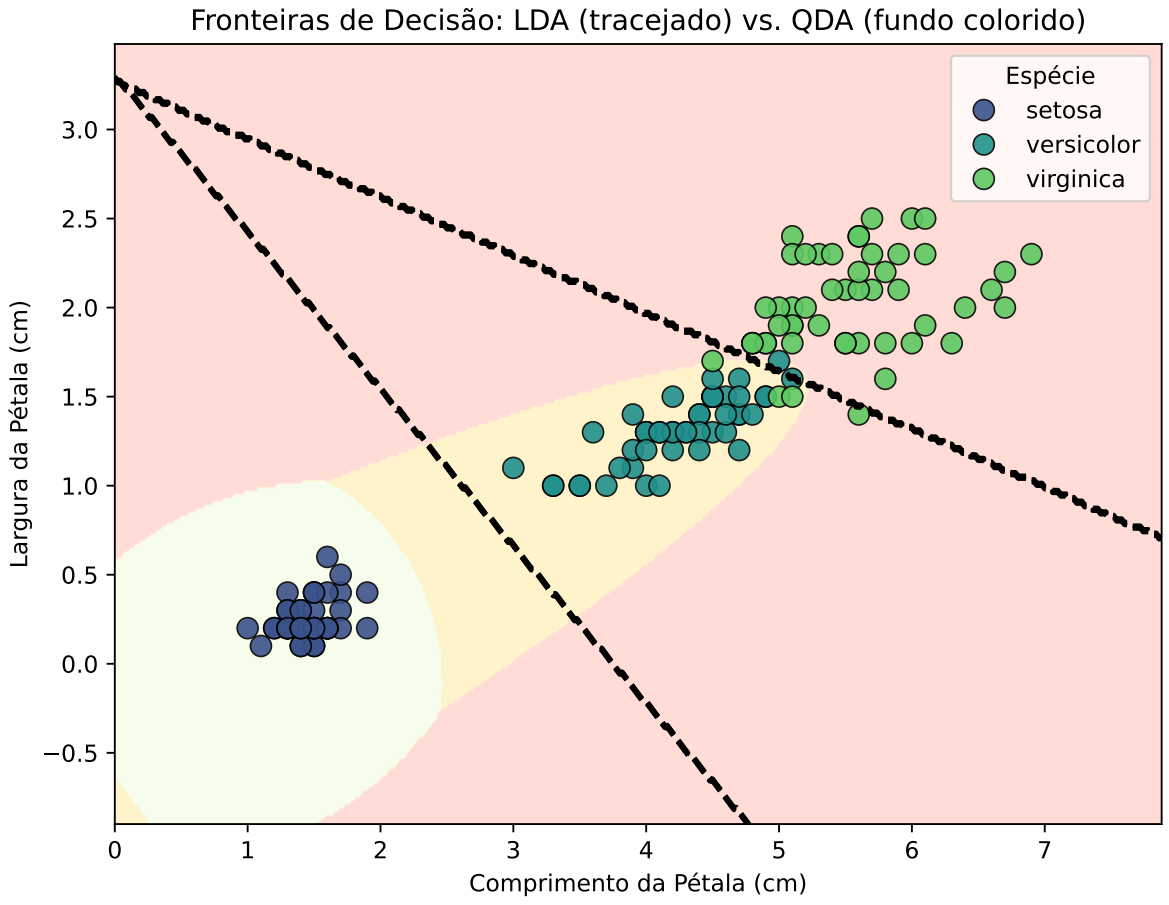

Figura 17.3: Fronteiras de decisão para LDA (linhas retas) e QDA (curvas) usando as variáveis da pétala.

A visualização das fronteiras de decisão nos mostra:

- **LDA (Linhas Tracejadas):** Como esperado, a LDA cria fronteiras perfeitamente retas entre os grupos. Ela faz um excelente trabalho ao isolar a *Setosa*, mas a linha que separa *Versicolor* e *Virginica* não mostra uma separação tão clara.
- **QDA (Fundo Colorido):** A QDA cria fronteiras curvas (cônicas). Note como a fronteira entre *Versicolor* e *Virginica* é uma parábola. Essa flexibilidade permite que a QDA se ajuste um pouco melhor à forma específica da nuvem de pontos de cada grupo.

17.4 Conclusão

Nesta análise exploratória, usamos a Análise Discriminante não apenas como um classificador, mas como uma ferramenta para **redução de dimensionalidade e interpretação**.

1. A **Análise Descritiva** nos deu a intuição inicial de que a *Setosa* era muito diferente e que a distinção entre *Versicolor* e *Virginica* seria o principal desafio.
2. A **Análise Discriminante Linear** confirmou isso e foi além: ela quantificou a importância de cada variável para a separação. A LD1, principal eixo de separação, revelou que a distinção dos grupos se baseia em uma medida de **contraste entre o tamanho das pétalas e o das sépalas**. A LD2 capturou uma diferença de forma mais sutil, opondo o comprimento da pétala à sua largura e à largura da sépala, o que foi fundamental para separar as espécies mais parecidas, *Versicolor* e *Virginica*.
3. A comparação com a **QDA** ilustrou a diferença fundamental entre os modelos: a flexibilidade das fronteiras de decisão. Para este dataset, onde a separação já é bastante clara, as vantagens da QDA em termos de poder de classificação podem ser pequenas, mas a visualização de suas fronteiras nos ajuda a entender a estrutura não-linear que a LDA não pode capturar.

18 Análise de Correspondência

Neste exemplo, vamos aplicar a Análise de Correspondência (CA) a um conjunto de dados real para explorar a associação entre duas variáveis categóricas. Usaremos o famoso conjunto de dados `HairEyeColor`, que contém as frequências de combinações de cor de cabelo e cor dos olhos. O ponto de partida é uma tabela de contingência. Carregamos os dados e os formatamos como uma tabela onde as linhas representam a cor do cabelo e as colunas, a cor dos olhos.

```
import pandas as pd
import numpy as np

# Dados do HairEyeColor (soma de homens e mulheres)
# Fonte: Fisher, R. A. (1940) The precision of discriminant functions.
# Annals of Eugenics, 10, 422-429.
data = {
    'Brown': [68, 119, 26, 7],
    'Blue': [20, 84, 17, 94],
    'Hazel': [15, 54, 14, 10],
    'Green': [5, 29, 14, 16]
}
index = ['Black', 'Brown', 'Red', 'Blond']
columns = ['Brown', 'Blue', 'Hazel', 'Green']

N = pd.DataFrame(data, index=index, columns=columns)
N
```

Tabela 18.1: Tabela de contingência: Cor do Cabelo vs. Cor dos Olhos

	Brown	Blue	Hazel	Green
Black	68	20	15	5
Brown	119	84	54	29
Red	26	17	14	14
Blond	7	94	10	16

A tabela acima é a nossa matriz de contagens $\mathbf{N}$.

18.1 Análise e Decomposição das Matrizes

A partir da tabela de contagens, calculamos as matrizes fundamentais para a CA. Primeiro, dividimos cada contagem pelo total geral para obter a matriz de proporções, ou **matriz de correspondência (P)**. Em seguida, calculamos os perfis médios (marginais de linha **r** e de coluna **c**) para obter a **matriz de independência (E)**, que representa as proporções esperadas se não houvesse associação entre as variáveis.

```
P = N / N.values.sum()
P.style.format("{:.4f}")

r = P.sum(axis=1)
c = P.sum(axis=0)
E = pd.DataFrame(np.outer(r, c), index=N.index, columns=N.columns)
```

Tabela 18.2: Matriz de correspondência (P)

	Brown	Blue	Hazel	Green
Black	0.1149	0.0338	0.0253	0.0084
Brown	0.2010	0.1419	0.0912	0.0490
Red	0.0439	0.0287	0.0236	0.0236
Blond	0.0118	0.1588	0.0169	0.0270

Tabela 18.3: Matriz de independência (E)

	Brown	Blue	Hazel	Green
Black	0.0678	0.0663	0.0287	0.0197
Brown	0.1795	0.1755	0.0759	0.0522
Red	0.0446	0.0436	0.0188	0.0130
Blond	0.0797	0.0779	0.0337	0.0232

A diferença entre $\mathbf{P}$ e $\mathbf{E}$ é a associação que a CA busca explicar. Para isso, decompomos essa associação em "tabelas de fatores" ortogonais ($\mathbf{T}_1, \mathbf{T}_2, \ldots$). A reconstrução da matriz original é dada por:

$$\mathbf{P} - \mathbf{E} \approx \mathbf{T}_1 + \mathbf{T}_2 + \ldots$$

Abaixo, calculamos as matrizes para os dois primeiros e mais importantes fatores.

```
# Realizando a análise de correspondência
S = np.diag(r**-0.5) @ (P - E).values @ np.diag(c**-0.5)
U, s, Vt = np.linalg.svd(S, full_matrices=False)
V = Vt.T

# Função para calcular a tabela de um fator k
def get_factor_table(k, U, s, V, r, c):
    R_hat_k = s[k] * np.outer(U[:, k], V[:, k])
    T_k = np.diag(r**0.5) @ R_hat_k @ np.diag(c**0.5)
    return pd.DataFrame(T_k, index=N.index, columns=N.columns)

# Calculando as tabelas dos fatores
T1 = get_factor_table(0, U, s, V, r, c)
T2 = get_factor_table(1, U, s, V, r, c)

# Matrizes reconstruídas
P_hat1 = E + T1
P_hat2 = E + T1 + T2
```

Tabela 18.4: Matriz do primeiro fator (T1)

	Brown	Blue	Hazel	Green
Black	0.0368	-0.0401	0.0067	-0.0035
Brown	0.0287	-0.0312	0.0052	-0.0027
Red	0.0062	-0.0068	0.0011	-0.0006
Blond	-0.0717	0.0780	-0.0131	0.0069

Tabela 18.5: Matriz do segundo fator (T2)

	Brown	Blue	Hazel	Green
Black	0.0086	0.0079	-0.0069	-0.0096
Brown	-0.0035	-0.0032	0.0028	0.0039
Red	-0.0084	-0.0077	0.0068	0.0094
Blond	0.0033	0.0030	-0.0026	-0.0037

18.2 Visualização e Interpretação

Para entender o que esses fatores representam, recorremos à visualização. Os heatmaps, por exemplo, são uma excelente forma de comparar a matriz original com as suas reconstruções.

```
import matplotlib.pyplot as plt
import seaborn as sns

fig, axes = plt.subplots(2, 2, figsize=(12, 10), sharex=True, sharey=True)
vmin = P.values.min()
vmax = P.values.max()

sns.heatmap(P, ax=axes[0, 0], annot=True, fmt=".3f", cmap="viridis", vmin=vmin, vmax=vmax
axes[0, 0].set_title("Original (P)")

sns.heatmap(E, ax=axes[0, 1], annot=True, fmt=".3f", cmap="viridis", vmin=vmin, vmax=vmax
axes[0, 1].set_title("Independência (E)")

sns.heatmap(P_hat1, ax=axes[1, 0], annot=True, fmt=".3f", cmap="viridis", vmin=vmin, vmax
axes[1, 0].set_title(r"Reconstrução com 1 Fator $\mathbf{\hat{P}}_1 = \mathbf{E} + \mathb

sns.heatmap(P_hat2, ax=axes[1, 1], annot=True, fmt=".3f", cmap="viridis", vmin=vmin, vmax
axes[1, 1].set_title(r"Reconstrução com 2 Fatores $\mathbf{\hat{P}}_2 = \mathbf{E} + \mat

plt.tight_layout(rect=[0, 0, 1, 0.96])
plt.show()
```

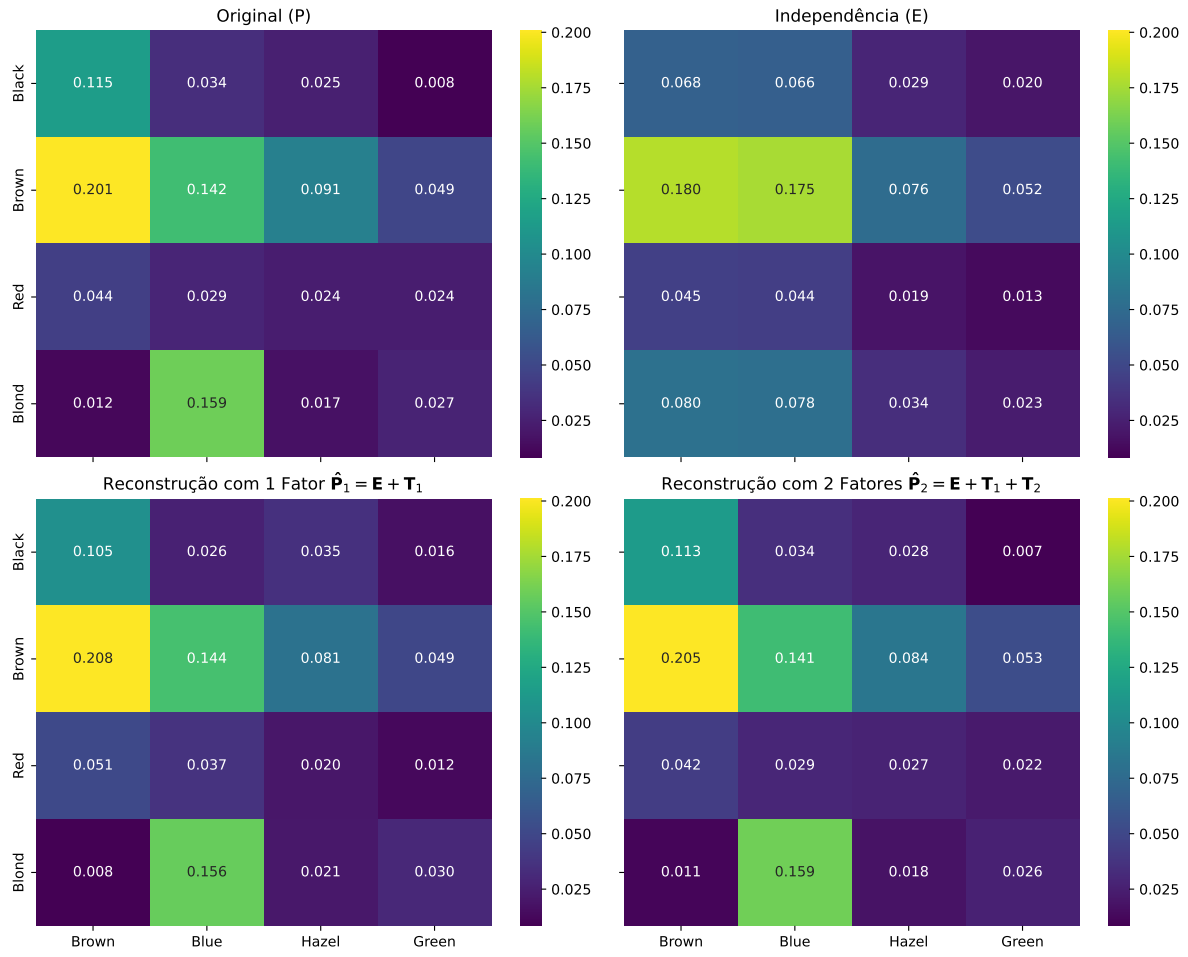

Figura 18.1: Comparação da reconstrução da matriz P

O mapa de calor da reconstrução com 2 fatores já é visualmente muito similar ao original, mostrando que as duas primeiras dimensões capturam a maior parte da associação. Se adicionarmos o terceiro (e último) par de fatores, isto é $\mathbf{P} - \mathbf{E} = \mathbf{T}_1 + \mathbf{T}_2 + \mathbf{T}_3$, teriamos uma reconstrução perfeita.

Finalmente, o **biplot** nos permite interpretar a natureza dessa associação. Ele projeta as categorias de linha e coluna em um espaço de baixa dimensão (geralmente 2D), onde a proximidade entre pontos indica uma associação mais forte.

```
# Coordenadas principais
Lambda = np.diag(s)
F = np.diag(r**-0.5) @ U @ Lambda
G = np.diag(c**-0.5) @ V @ Lambda
```

```
F_df = pd.DataFrame(F[:, :2], index=N.index, columns=['Dim 1', 'Dim 2'])
G_df = pd.DataFrame(G[:, :2], index=N.columns, columns=['Dim 1', 'Dim 2'])

# Inércia explicada
total_inertia = np.sum(s**2)
inertia_explained = (s**2) / total_inertia

# Plot
fig, ax = plt.subplots(figsize=(7, 5))

# Pontos das linhas (cor de cabelo)
ax.scatter(F_df['Dim 1'], F_df['Dim 2'], c='red', marker='s', s=100, label='Cor do Cabelo
for txt in F_df.index:
    ax.text(F_df.loc[txt, 'Dim 1'] + 0.01, F_df.loc[txt, 'Dim 2'], txt, color='red', font

# Pontos das colunas (cor dos olhos)
ax.scatter(G_df['Dim 1'], G_df['Dim 2'], c='blue', marker='o', s=100, label='Cor dos Olho
for txt in G_df.index:
    ax.text(G_df.loc[txt, 'Dim 1'] + 0.01, G_df.loc[txt, 'Dim 2'], txt, color='blue', fon

ax.axhline(0, color='gray', linestyle='--')
ax.axvline(0, color='gray', linestyle='--')
ax.grid(True, linestyle='--', alpha=0.6)
ax.set_xlabel(f"Dimensão 1 ({inertia_explained[0]:.1%})")
ax.set_ylabel(f"Dimensão 2 ({inertia_explained[1]:.1%})")
ax.legend()

plt.show()
```

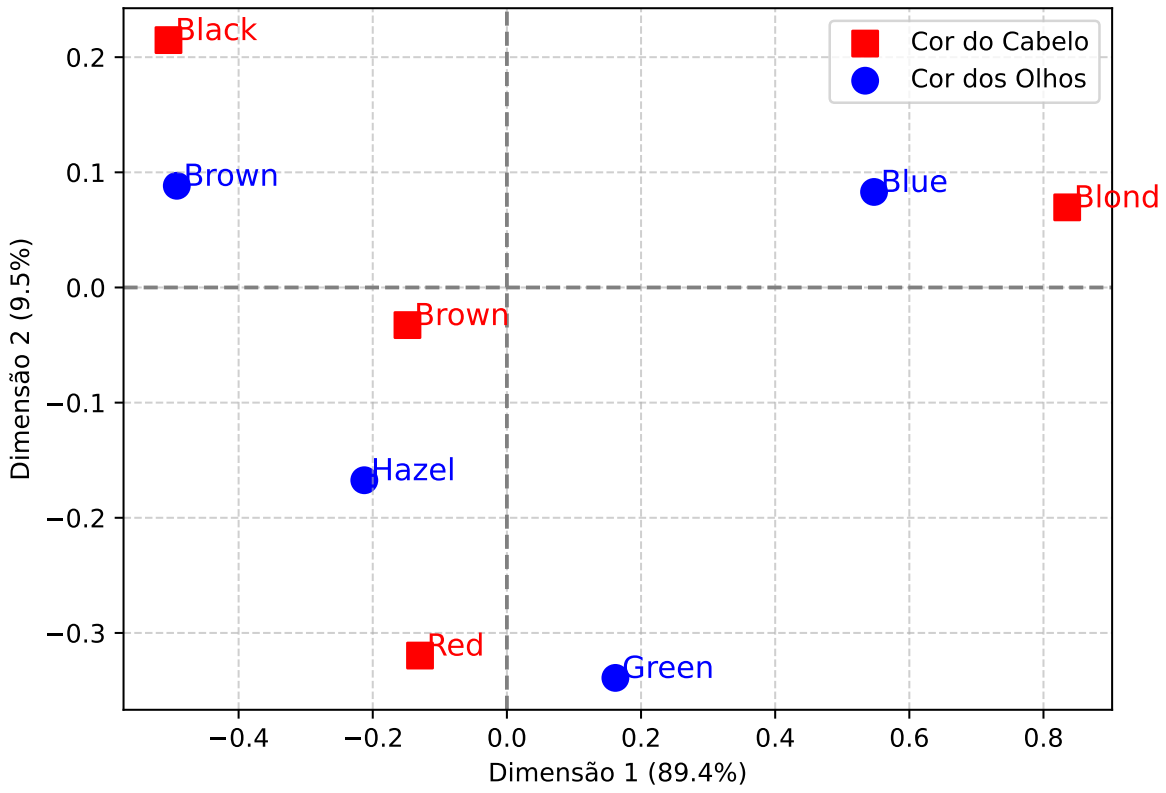

Figura 18.2: Biplot da associação entre cor de cabelo e cor dos olhos

A interpretação visual do biplot é poderosa, mas para uma análise rigorosa, devemos examinar as métricas de qualidade e contribuição que discutimos na Capítulo 11.

```
# Distances to origin (squared) for full space
row_dist2 = np.sum(F**2, axis=1)
col_dist2 = np.sum(G**2, axis=1)

# Point inertias (contribution to total inertia)
row_inertia = r.values * row_dist2
col_inertia = c.values * col_dist2

# Quality of representation (cos2) for the first 2 dims
row_cos2 = F[:, :2]**2 / row_dist2[:, np.newaxis]
col_cos2 = G[:, :2]**2 / col_dist2[:, np.newaxis]

# Relative contributions to the first 2 dims' inertia
row_contrib = (np.diag(r.values) @ F[:, :2]**2) / (s[:2]**2)
```

```
col_contrib = (np.diag(c.values) @ G[:, :2]**2) / (s[:2]**2)

# Create pandas DataFrames for nice display
row_metrics = pd.DataFrame({
    'Inércia': row_inertia,
    'cos2_mapa': row_cos2.sum(axis=1),
    'ctr_dim1': row_contrib[:, 0],
    'ctr_dim2': row_contrib[:, 1]
}, index=N.index)
row_metrics.index.name = "Cor do Cabelo"


col_metrics = pd.DataFrame({
    'Inércia': col_inertia,
    'cos2_mapa': col_cos2.sum(axis=1),
    'ctr_dim1': col_contrib[:, 0],
    'ctr_dim2': col_contrib[:, 1]
}, index=N.columns)
col_metrics.index.name = "Cor dos Olhos"

# Displaying with style
styled_rows = row_metrics.style.format({
    'Inércia': '{:.4f}',
    'cos2_mapa': '{:.3f}',
    'ctr_dim1': '{:.1%}',
    'ctr_dim2': '{:.1%}'
}).set_caption("Métricas para Cor do Cabelo")

styled_cols = col_metrics.style.format({
    'Inércia': '{:.4f}',
    'cos2_mapa': '{:.3f}',
    'ctr_dim1': '{:.1%}',
    'ctr_dim2': '{:.1%}'
}).set_caption("Métricas para Cor dos Olhos")

from IPython.display import display
display(styled_rows)
display(styled_cols)
```

O biplot e as métricas quantitativas nos permitem construir uma interpretação detalhada:

1. **Qualidade da Representação (`cos2_mapa`):** As tabelas mostram que a maioria das categorias

Tabela 18.6: Métricas de interpretação para as categorias de cabelo e olhos.

(a) Métricas para Cor do Cabelo

	Inércia	cos2_mapa	ctr_dim1	ctr_dim2
Cor do Cabelo				
Black	0.0554	0.990	22.2%	37.9%
Brown	0.0123	0.906	5.1%	2.8%
Red	0.0151	0.945	1.0%	55.1%
Blond	0.1508	1.000	71.7%	4.2%

(b) Métricas para Cor dos Olhos

	Inércia	cos2_mapa	ctr_dim1	ctr_dim2
Cor dos Olhos				
Brown	0.0931	0.998	43.1%	13.0%
Blue	0.1113	1.000	52.1%	11.2%
Hazel	0.0131	0.879	3.4%	19.8%
Green	0.0161	0.948	1.4%	55.9%

tem uma excelente qualidade de representação no mapa (cos2 > 0.8), com destaque para `Black`, `Blond` e `Blue`. A categoria `Hazel` (olhos) é a única com uma representação mais fraca, portanto sua posição no mapa é menos precisa e deve ser interpretada com cautela.

2. **Significado da Dimensão 1 (89.4% da inércia):**

   - **Contribuições (`ctr_dim1`):** As tabelas de contribuição mostram que este eixo é definido primariamente pelas cores de cabelo `Black` e `Blond` e pelas cores de olho `Brown` e `Blue`.
   - **Interpretação:** A Dimensão 1 representa o contraste "cores escuras vs. claras". À esquerda do eixo, temos as categorias de cabelo e olhos escuros; à direita, as claras. Isso confirma que a associação mais forte nos dados é entre cabelo/olhos escuros e cabelo/olhos claros.

3. **Significado da Dimensão 2 (9.5% da inércia):**

   - **Contribuições (`ctr_dim2`):** A segunda dimensão é majoritariamente definida pelo cabelo `Red` e pelos olhos `Green` e `Hazel`.
   - **Interpretação:** Este eixo secundário separa as cores "intermediárias", contrastando o cabelo ruivo e os olhos verdes/avelã das demais categorias.

4. **Pontos mais influentes (`Inércia`):** As categorias com maior inércia, e portanto maior impacto na associação geral, são `Blond` (cabelo) e `Blue` (olhos). Seus perfis são os que mais se desviam da média, impulsionando a maior parte da estrutura encontrada.

Em resumo, a análise quantitativa confirma e aprofunda a interpretação visual. Ela revela uma estrutura muito clara nos dados: a dimensão mais forte de associação é o gradiente de cores claras para escuras, seguida por uma dimensão secundária que isola as combinações de cabelo ruivo com olhos verdes/avelã.

19 Análise de Correspondência Múltipla (ACM)

A Análise de Correspondência Múltipla (ACM) é uma extensão da Análise de Correspondência Simples (ACS) para mais de duas variáveis categóricas. Nesta seção, vamos explorar a ACM através de dois exemplos práticos que ilustram as duas principais abordagens da técnica:

1. **ACM sobre a Tabela Indicadora:** Ideal para conjuntos de dados menores, onde a visualização e interpretação dos **indivíduos** é tão importante quanto a das variáveis. Usaremos o conjunto de dados `poison` para ilustrar este caso.
2. **ACM com foco na Tabela de Burt:** Adequada quando o foco principal é entender as **associações entre as variáveis**, ou quando a visualização dos indivíduos se torna impraticável. Usaremos o conjunto de dados `tea` para ilustrar esta abordagem.

Para as análises, usaremos os pacotes R `FactoMineR` para os cálculos e `factoextra` para as visualizações.

```
library(tidyverse)
library(FactoMineR)
library(factoextra)
library(patchwork)
```

19.1 Caso 1: ACM na Tabela Indicadora com o dataset `poison`

Esta abordagem é mais útil quando temos um número de observações pequeno o suficiente para que a interpretação da posição de cada indivíduo (ou de pequenos grupos de indivíduos) seja viável e informativa. Frequentemente, esse tipo de análise pode servir como um pré-processamento para uma análise secundária, como o agrupamento de indivíduos.

Vamos utilizar o conjunto de dados `poison`, disponível no pacote `FactoMineR`. Ele contém dados sobre 55 crianças que foram diagnosticadas com intoxicação alimentar. As variáveis registram os sintomas apresentados (`Nausea`, `Vomiting`, `Diarrhoea`, etc.) e os alimentos consumidos (`Potato`, `Fish`, `Mayo`, etc.). O objetivo é identificar perfis de sintomas e associá-los aos alimentos ingeridos.

```r
data(poison)

# Transformar em tibble para melhor visualização
poison_df <- as_tibble(poison)

# Exibir as primeiras linhas e a estrutura
knitr::kable(head(poison_df))
# Verificar a estrutura dos dados (fatores)
str(poison_df)
```

```
tibble [55 x 15] (S3: tbl_df/tbl/data.frame)
 $ Age       : int [1:55] 9 5 6 9 7 72 5 10 5 11 ...
 $ Time      : int [1:55] 22 0 16 0 14 9 16 8 20 12 ...
 $ Sick      : Factor w/ 2 levels "Sick_n","Sick_y": 2 1 2 1 2 2 2 2 2 2 ...
 $ Sex       : Factor w/ 2 levels "F","M": 1 1 1 1 2 2 1 1 2 2 ...
 $ Nausea    : Factor w/ 2 levels "Nausea_n","Nausea_y": 2 1 1 1 1 1 1 2 2 1 ...
 $ Vomiting  : Factor w/ 2 levels "Vomit_n","Vomit_y": 1 1 2 1 2 1 2 2 1 2 ...
 $ Abdominals: Factor w/ 2 levels "Abdo_n","Abdo_y": 2 1 2 1 2 2 2 2 2 1 ...
 $ Fever     : Factor w/ 2 levels "Fever_n","Fever_y": 2 1 2 1 2 2 2 2 2 2 ...
 $ Diarrhae  : Factor w/ 2 levels "Diarrhea_n","Diarrhea_y": 2 1 2 1 2 2 2 2 2 2 ...
 $ Potato    : Factor w/ 2 levels "Potato_n","Potato_y": 2 2 2 2 2 2 2 2 2 2 ...
 $ Fish      : Factor w/ 2 levels "Fish_n","Fish_y": 2 2 2 2 2 1 2 2 2 2 ...
 $ Mayo      : Factor w/ 2 levels "Mayo_n","Mayo_y": 2 2 2 1 2 2 2 2 2 2 ...
 $ Courgette : Factor w/ 2 levels "Courg_n","Courg_y": 2 2 2 2 2 2 2 2 2 2 ...
 $ Cheese    : Factor w/ 2 levels "Cheese_n","Cheese_y": 2 1 2 2 2 2 2 2 2 2 ...
 $ Icecream  : Factor w/ 2 levels "Icecream_n","Icecream_y": 2 2 2 2 2 2 2 2 2 2 ...
```

Tabela 19.1: Amostra dos dados do dataset poison.

Age	Time	Sick	Sex	Nausea	Vomiting	Abdominals	Fever	Diarrhae	Potato	Fish	Mayo	Courgette	Cheese	Icecream
9	22	Sick_y	F	Nausea_y	Vomit_n	Abdo_y	Fever_y	Diarrhea_y	Potato_y	Fish_y	Mayo_y	Courg_y	Cheese_y	Icecream_y
5	0	Sick_n	F	Nausea_y	Vomit_n	Abdo_n	Fever_n	Diarrhea_y	Potato_y	Fish_y	Mayo_y	Courg_y	Cheese_n	Icecream_y
6	16	Sick_y	F	Nausea_y	Vomit_y	Abdo_y	Fever_y	Diarrhea_y	Potato_y	Fish_y	Mayo_y	Courg_y	Cheese_y	Icecream_y
9	0	Sick_n	F	Nausea_y	Vomit_n	Abdo_n	Fever_n	Diarrhea_y	Potato_y	Fish_y	Mayo_y	Courg_y	Cheese_y	Icecream_y
7	14	Sick_y	M	Nausea_y	Vomit_y	Abdo_y	Fever_y	Diarrhea_y	Potato_y	Fish_y	Mayo_y	Courg_y	Cheese_y	Icecream_y
72	9	Sick_y	M	Nausea_y	Vomit_n	Abdo_y	Fever_y	Diarrhea_y	Potato_y	Fish_n	Mayo_y	Courg_y	Cheese_y	Icecream_y

As variáveis `Age` e `Sex` serão tratadas como suplementares, o que significa que elas não participarão da construção dos eixos fatoriais, mas serão projetadas no mapa para auxiliar na interpretação. Além delas, as variáveis binárias `Sick` e `Sex` também serão tratadas como suplementares. A variável `Sick` indica

se o indivíduo foi diagnosticado com intoxicação alimentar. Ou seja, essa variável é importantíssima para a análise.

```
# Executar a ACM
# graph = FALSE evita que o gráfico padrão do FactoMineR seja plotado imediatamente
mca_poison <- MCA(poison, quanti.sup = 1:2, quali.sup = 3:4, graph = FALSE)

# O scree plot mostra a inércia de cada dimensão
# A linha vermelha indica a inércia média esperada se não houvesse associação
fviz_screeplot(mca_poison,
    repel=TRUE,
    addlabels = TRUE,
    ggtheme = theme_minimal(),
    ylim = c(0, 35)
)
```

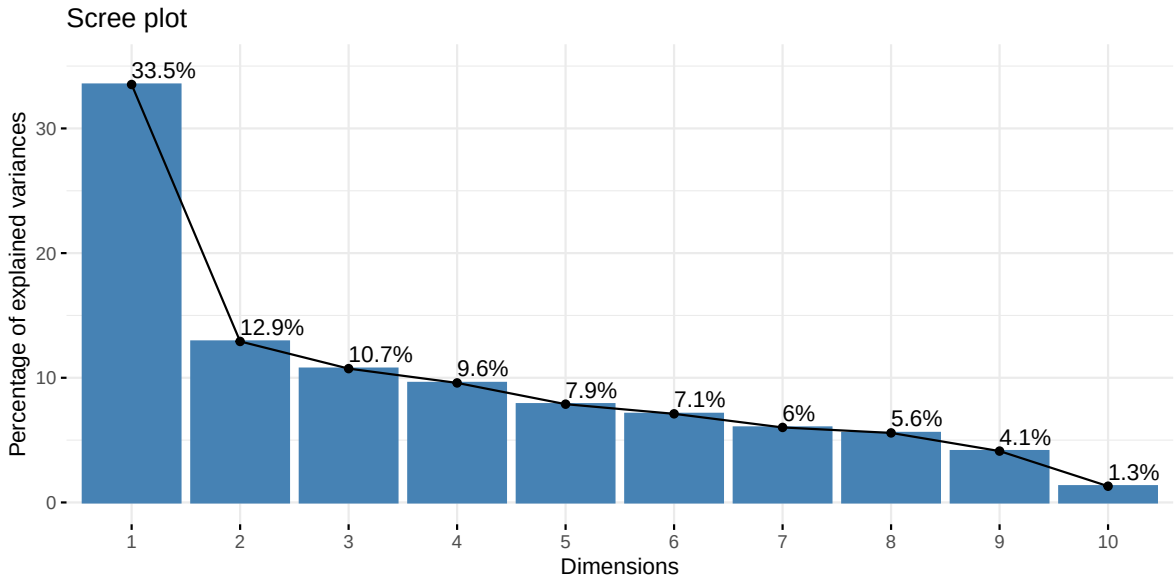

Figura 19.1: Scree plot para a ACM do dataset poison. As dimensões acima da linha pontilhada são as mais relevantes.

O *scree plot* indica que a primeira dimensão captura a maior parte da informação, e está bem acima do limiar de inércia média. Vamos focar nossa análise majoritariamente nela.

O biplot é a principal ferramenta de visualização. Ele mostra as categorias e os indivíduos no mesmo mapa, permitindo identificar perfis.

```
# Criar o biplot de indivíduos e variáveis
fviz_mca_biplot(mca_poison,
    repel = TRUE, # Evita sobreposição de texto
    ggtheme = theme_minimal(),
)
```

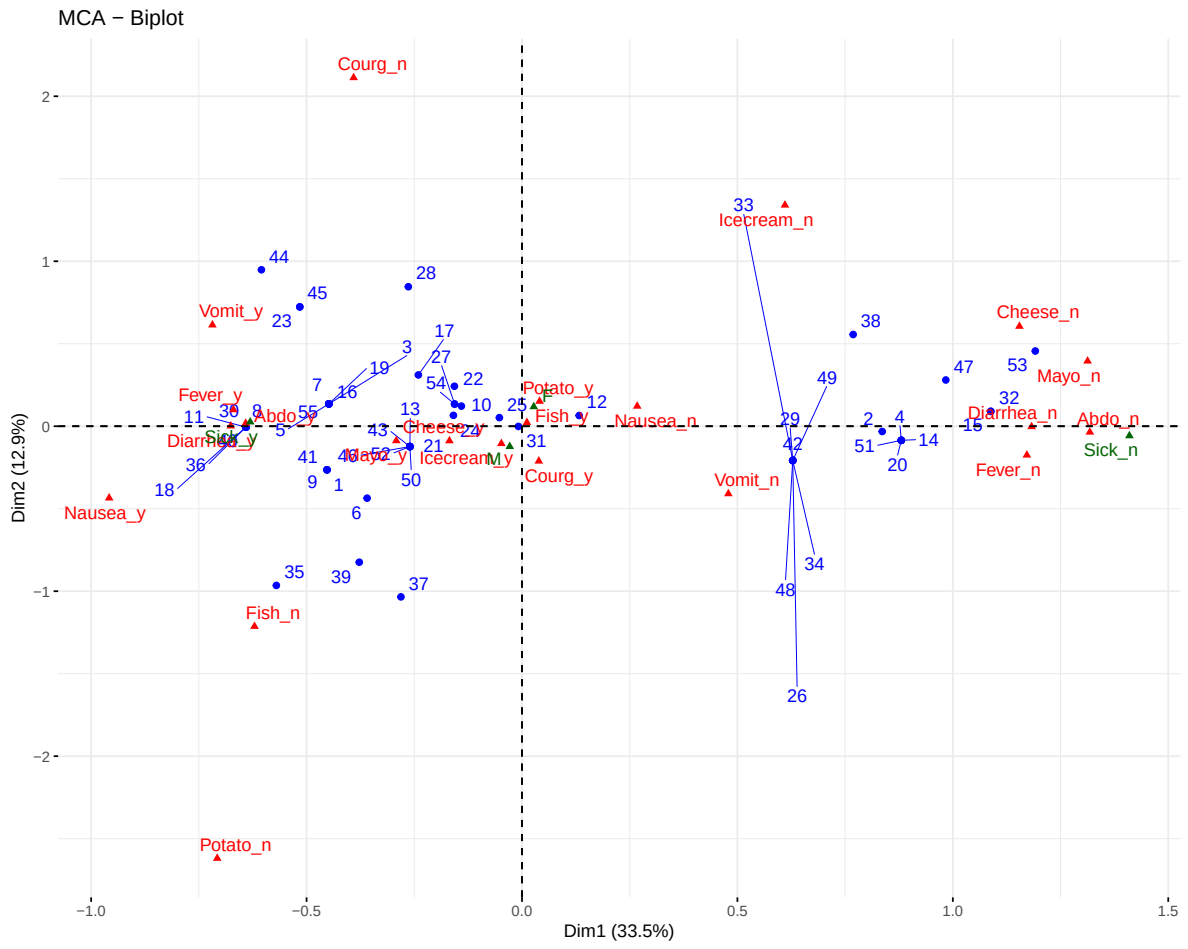

Figura 19.2: Biplot da ACM para o dataset poison. Indivíduos coloridos por presença de vômito.

Interpretação do Biplot:

O mapa revela uma estrutura clara, muitas vezes chamada de "efeito de tamanho" ou estrutura em V, comum quando há um grupo muito distinto (neste caso, os doentes vs. saudáveis).

- **Dimensão 1 (Eixo Horizontal):** É o eixo da **Saúde**.

  - **Polo Esquerdo:** Agrupa todas as categorias de sintomas (`Diarrhoea_sim`, `Fever_sim`, `Vomiting_sim`, `Nausea_sim`) e o estado `Sick_sim`.
  - **Polo Direito:** Agrupa a ausência de sintomas e o estado `Sick_nao`.
  - Basicamente, este eixo separa as crianças doentes das saudáveis.

- **Dimensão 2 (Eixo Vertical):** Diferencia nuances nos tipos de sintomas, mas é **menos importante** para a análise global.
  - Possui inércia significativamente menor que a primeira dimensão.
  - As variáveis mais importantes para definir os grupos de interesse (`Sick_y`/`Sick_n`) não estão fortemente associadas a este eixo.

Os gráficos de contribuição e qualidade ($\cos^2$) confirmam a importância das variáveis na definição dos eixos.

```
# Gráfico de contribuições para a Dimensão 1
p1 <- fviz_contrib(mca_poison, "var", axes = 1, top = 10)

# Gráfico de contribuições para a Dimensão 2
p2 <- fviz_contrib(mca_poison, "var", axes = 2, top = 10)

# Combinar os gráficos
p1 + p2
```

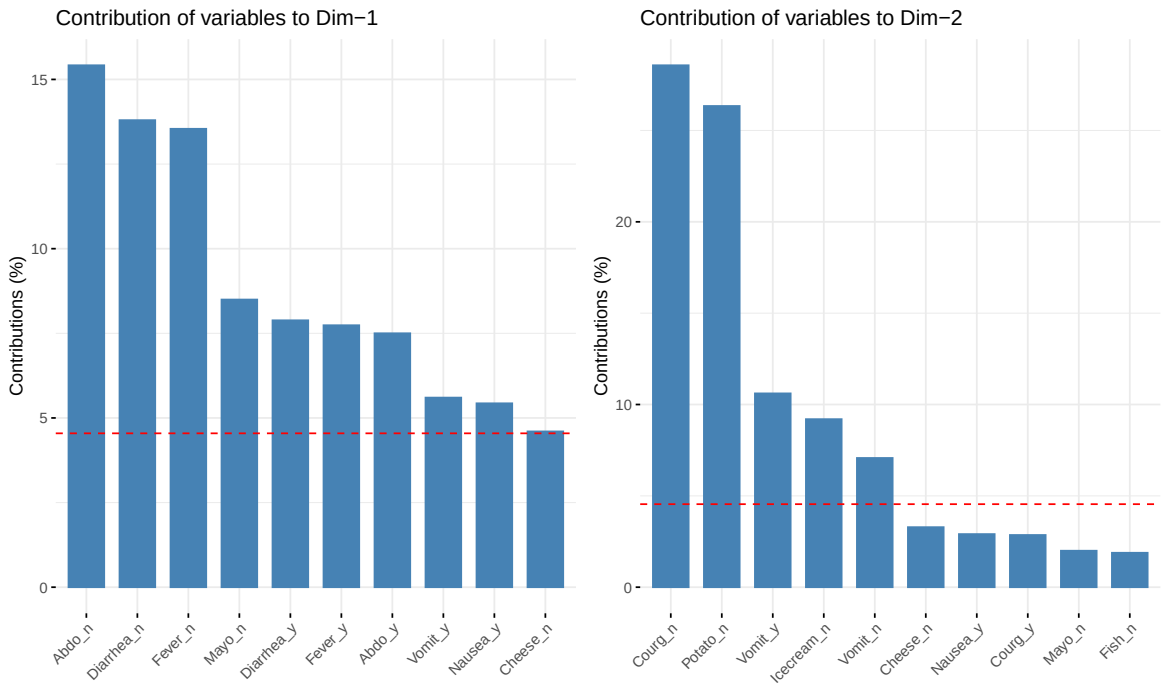

Figura 19.3: Contribuições das categorias para as Dimensões 1 e 2 (dataset poison).

A análise sobre a tabela indicadora permite ver essa estrutura "Doente vs Saudável" como a principal fonte de variação nos dados.

> Dica: O Efeito Guttman (Ferradura)
>
> Em análises de correspondência, é comum observar uma disposição dos pontos em forma de parábola ou ferradura (Horseshoe effect). Isso geralmente indica que a primeira dimensão é dominante e representa um gradiente latente forte (como "gravidade da doença" ou "tempo"), e a segunda dimensão é uma função quadrática da primeira. Se isso ocorrer, a interpretação deve focar principalmente na ordem ao longo da curva.

19.1.1 Agrupamento de Indivíduos (HCPC)

Uma extensão natural da ACM é o agrupamento (clustering) dos indivíduos. Como a ACM transforma as variáveis categóricas em coordenadas numéricas, podemos usar essas coordenadas para calcular distâncias entre indivíduos e agrupa-los.

O pacote `FactoMineR` oferece a função `HCPC` (Hierarchical Clustering on Principal Components) para isso.

```
# Realizar o agrupamento hierárquico sobre os resultados da ACM
res_hcpc <- HCPC(mca_poison, nb.clust = 3, graph = FALSE)

# Visualizar os clusters no mapa da ACM
fviz_cluster(res_hcpc,
             repel = TRUE,            # Evitar sobreposição
             show.clust.cent = TRUE,  # Mostrar o centroide de cada grupo
             palette = "jco",         # Paleta de cores
             ggtheme = theme_minimal(),
             main = "Agrupamento Hierárquico (HCPC) sobre ACM")
```

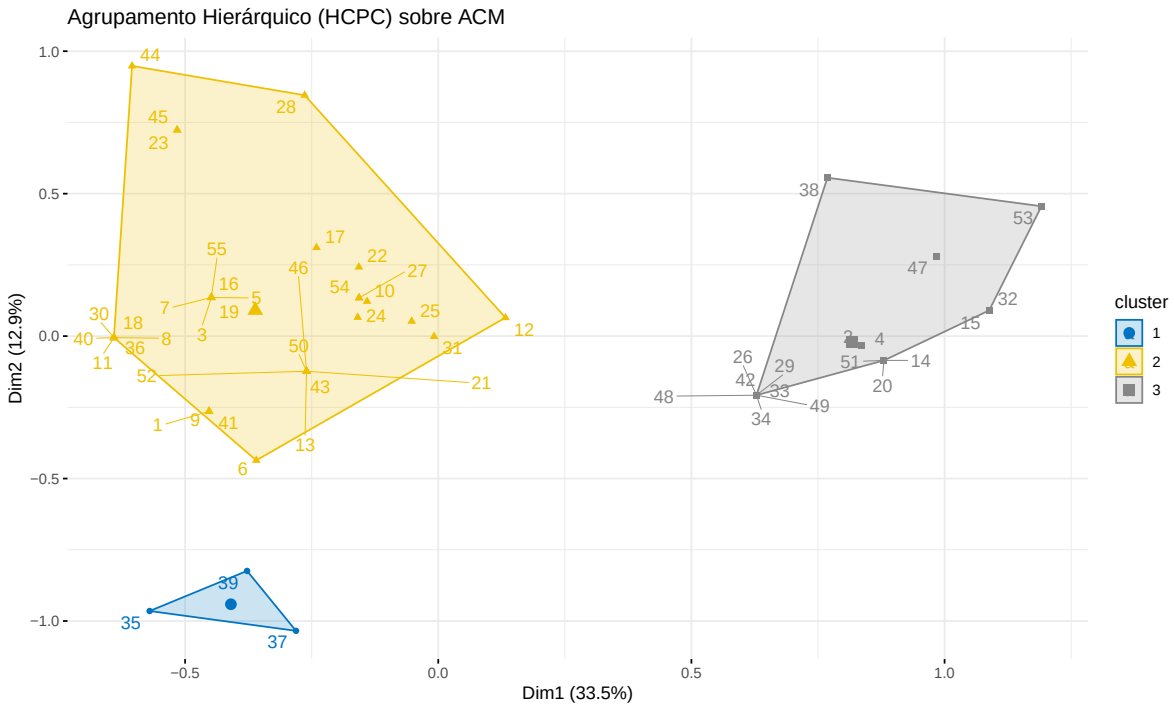

Figura 19.4: Dendrograma e mapa de fatores com os clusters identificados.

Fica como tarefa ao leitor identificar os clusters e interpretar sua significância.

19.2 Caso 2: ACM com foco na Tabela de Burt com o dataset `tea`

Quando o número de indivíduos é grande (centenas ou milhares), o biplot de indivíduos se torna um borrão de pontos impossível de interpretar. Nesses casos, o foco da análise muda: em vez de analisar perfis de indivíduos, buscamos entender a **estrutura de associações entre as variáveis**.

Essa abordagem é conceitualmente ligada à **Tabela de Burt**, que é uma matriz de todas as tabelas de contingência de duas vias entre as variáveis. A ACM na tabela indicadora é matematicamente equivalente a uma Análise de Correspondência Simples (ACS) na tabela de Burt, e o mapa de variáveis resultante é o mesmo (com ajuste na escala).

Vamos usar o dataset `tea`, também do `FactoMineR`, que contém respostas de 300 indivíduos sobre seus hábitos de consumo de chá. Para facilitar a interpretação, vamos selecionar apenas **6 variáveis-chave** que capturam os principais aspectos do comportamento de consumo:

```
data(tea)

# Selecionar apenas 5 variáveis-chave para facilitar interpretação:
# - Tea: tipo de chá preferido (black, green, Earl Grey)
# - How: como prepara (tea bag, unpackaged, etc)
# - how: forma de beber (alone, milk, lemon, etc)
# - where: onde compra (tea shop, chain store)
# - price: importância do preço (p_unknown, p_variable, p_branded, etc)
tea_ativa <- tea[, c("Tea", "How", "how", "where", "price")]

head(as_tibble(tea_ativa)) |>
  knitr::kable()
# Ver resumo das categorias
summary(tea_ativa)
```

```
       Tea         How                      how
 black    : 74   alone:195   tea bag           :170
 Earl Grey:193   lemon: 33   tea bag+unpackaged: 94
 green    : 33   milk : 63   unpackaged        : 36
                 other:  9


                  where                 price
 chain store         :192   p_branded      : 95
 chain store+tea shop: 78   p_cheap        :  7
 tea shop            : 30   p_private label: 21
                            p_unknown      : 12
                            p_upscale      : 53
                            p_variable     :112
```

Tabela 19.2: Amostra dos dados selecionados do dataset tea.

Tea	How	how	where	price
black	alone	tea bag	chain store	p_unknown
black	milk	tea bag	chain store	p_variable
Earl Grey	alone	tea bag	chain store	p_variable
Earl Grey	alone	tea bag	chain store	p_variable
Earl Grey	alone	tea bag	chain store	p_variable
Earl Grey	alone	tea bag	chain store	p_private label

Por que apenas 5 variáveis?

O dataset `tea` original contém 18 variáveis ativas. Para este exemplo, selecionamos apenas 5 variáveis-chave por razões pedagógicas:

1. **Interpretabilidade:** Com menos variáveis, o mapa de categorias fica mais limpo e fácil de interpretar
2. **Foco:** As variáveis escolhidas (`Tea`, `How`, `how`, `where`, `price`) capturam os principais aspectos do comportamento de consumo
3. **Demonstração:** Facilita a visualização dos conceitos fundamentais da ACM via Tabela de Burt

Em uma análise real de mercado, você usaria todas as variáveis relevantes disponíveis.

Com 300 indivíduos, plotar pontos individuais seria inviável. Nosso objetivo é mapear as **associações entre hábitos de consumo**.

A seguir, visualizamos a tabela de Burt para o dataset `tea`. Ela nada mais é do que o cruzamento de todas as variáveis com todas as variáveis. Matematicamente, a Tabela de Burt é $\mathbf{B} = \mathbf{Z}'\mathbf{Z}$, onde $\mathbf{Z}$ é a matriz indicadora.

```
# Executar a ACM sobre a tabela indicadora
mca_tea <- MCA(tea_ativa, graph = FALSE)

# A função MCA internamente utiliza a matriz indicadora,
# mas podemos extrair ou calcular a tabela de Burt

# Criar matriz indicadora manualmente para demonstração
library(FactoMineR)
tea_matrix <- tab.disjonctif(tea_ativa)

# Calcular a tabela de Burt (Z'Z)
```

```r
burt_table <- t(tea_matrix) %*% tea_matrix

# Visualizar a estrutura da tabela de Burt como heatmap
library(pheatmap)
pheatmap(burt_table,
         display_numbers = FALSE,
         cluster_rows = FALSE,
         cluster_cols = FALSE,
         color = colorRampPalette(c("white", "steelblue", "darkblue"))(50),
         main = "Tabela de Burt - Dataset Tea",
         fontsize = 7,
         border_color = "grey90")
```

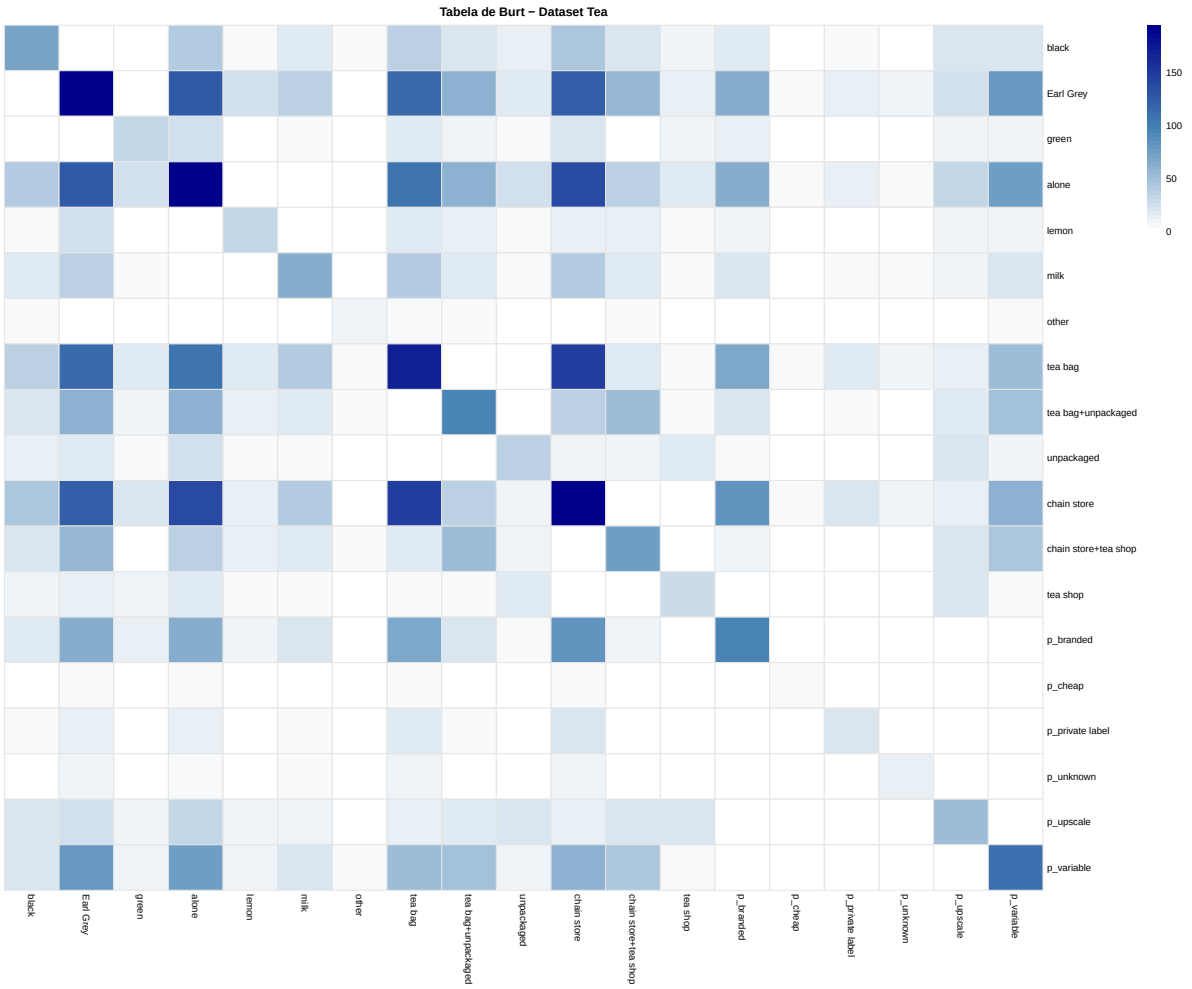

Figura 19.5: Visualização da estrutura da Tabela de Burt para o dataset tea. Células mais escuras indicam maior frequência de co-ocorrência entre categorias.

Interpretação da Tabela de Burt:

A tabela de Burt é uma matriz simétrica de dimensão $J \times J$ (onde J é o total de categorias), construída como $\mathbf{B} = \mathbf{Z}'\mathbf{Z}$. Sua estrutura é organizada em blocos:

- **Blocos na diagonal:** Contêm as frequências marginais de cada categoria (diagonal dentro de cada bloco indica quantos indivíduos têm aquela característica)
- **Blocos fora da diagonal:** Mostram as co-ocorrências entre categorias de variáveis diferentes - essas são as associações que buscamos explorar!

O que o heatmap revela:

Observando o padrão de cores (células escuras = alta frequência de co-ocorrência):

1. **Blocos bem definidos:** Há regiões com concentrações claras de cor escura, sugerindo que certas combinações de hábitos são muito comuns. Por exemplo, categorias relacionadas a consumo prático (tea bag, supermercados) co-ocorrem frequentemente entre si.
2. **Estrutura de blocos:** A matriz não é uniforme - há "blocos escuros" indicando clusters de categorias que tendem a aparecer juntas, separados por regiões mais claras que indicam associações fracas ou negativas.
3. **Simetria:** Como esperado, a matriz é perfeitamente simétrica (a co-ocorrência de A com B é igual à de B com A), o que facilita a interpretação visual.

A ACM que executaremos a seguir decompõe essa tabela complexa de co-ocorrências em dimensões interpretáveis, permitindo visualizar os padrões em um espaço de baixa dimensão.

19.2.1 ACM e Interpretação da Inércia

Agora executamos a ACM propriamente dita e analisamos a decomposição da inércia:

```
# O scree plot mostra a inércia de cada dimensão
fviz_screeplot(mca_tea,
               addlabels = TRUE,
               ggtheme = theme_minimal(),
               ylim = c(0, 20))
```

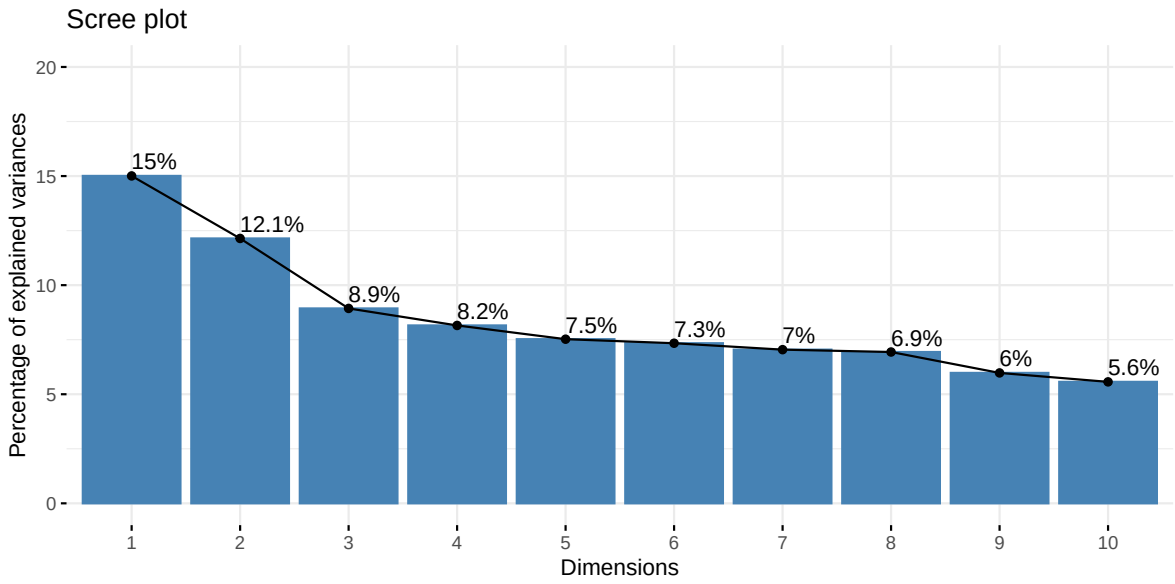

Figura 19.6: Scree plot para a ACM do dataset tea. A linha pontilhada vermelha representa o limiar de inércia média (critério de Benzécri).

Interpretação do Scree Plot:

Como esperado em ACM, as porcentagens de inércia explicada parecem baixas à primeira vista. A primeira dimensão explica cerca de 15% da inércia total, e a segunda explica 12.1%. **Isso NÃO significa que a análise é ruim!** É uma característica intrínseca da ACM sobre a matriz indicadora.

Análise dos autovalores:

- Com 5 variáveis ativas, a inércia média esperada sob independência seria $1/Q = 1/5 = 0.20$ (ou 20%)
- As primeiras 2-3 dimensões têm autovalores claramente **acima** desse limiar, indicando que capturam estrutura real (não apenas ruído)
- O "cotovelo" no gráfico ocorre após a segunda dimensão, sugerindo que as duas primeiras são suficientes para capturar os principais padrões de associação
- As dimensões posteriores mostram decréscimo gradual, típico de variação de fundo

Conclusão: Apesar das porcentagens aparentemente baixas, as duas primeiras dimensões são **altamente significativas** pelo critério de Benzécri. Juntas, capturam os principais eixos de variação nos hábitos de consumo de chá.

> **Particularidade da Inércia na ACM**
>
> Como discutido na Seção 11.7, a inércia total na ACM via matriz indicadora é fixa e depende do número de variáveis e categorias: Inércia Total $= \frac{J-Q}{Q}$, onde J é o total de categorias e Q o número de variáveis.
> Com 5 variáveis e 19 categorias totais, temos inércia total "estrutural" que dilui as porcentagens. O critério de Benzécri corrige essa distorção: uma dimensão é informativa se seu autovalor > $1/Q$ (neste caso, > 0.20 ou 20%). Isso nos diz que devemos focar na **magnitude relativa** dos autovalores (acima vs. abaixo da média), não nas porcentagens absolutas.

19.2.2 Mapa de Categorias e Interpretação das Associações

Com 300 indivíduos, plotar todos os pontos seria inviável. Vamos focar exclusivamente no **mapa de categorias**, que nos permite entender a estrutura de associação entre as variáveis:

```
# Plotar apenas as variáveis (categorias), coloridas por qualidade de representação
fviz_mca_var(mca_tea,
             repel = TRUE,
             ggtheme = theme_minimal(),
             col.var = "cos2",  # Cor por qualidade de representação
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             title = "Mapa de Categorias - ACM Dataset Tea")
```

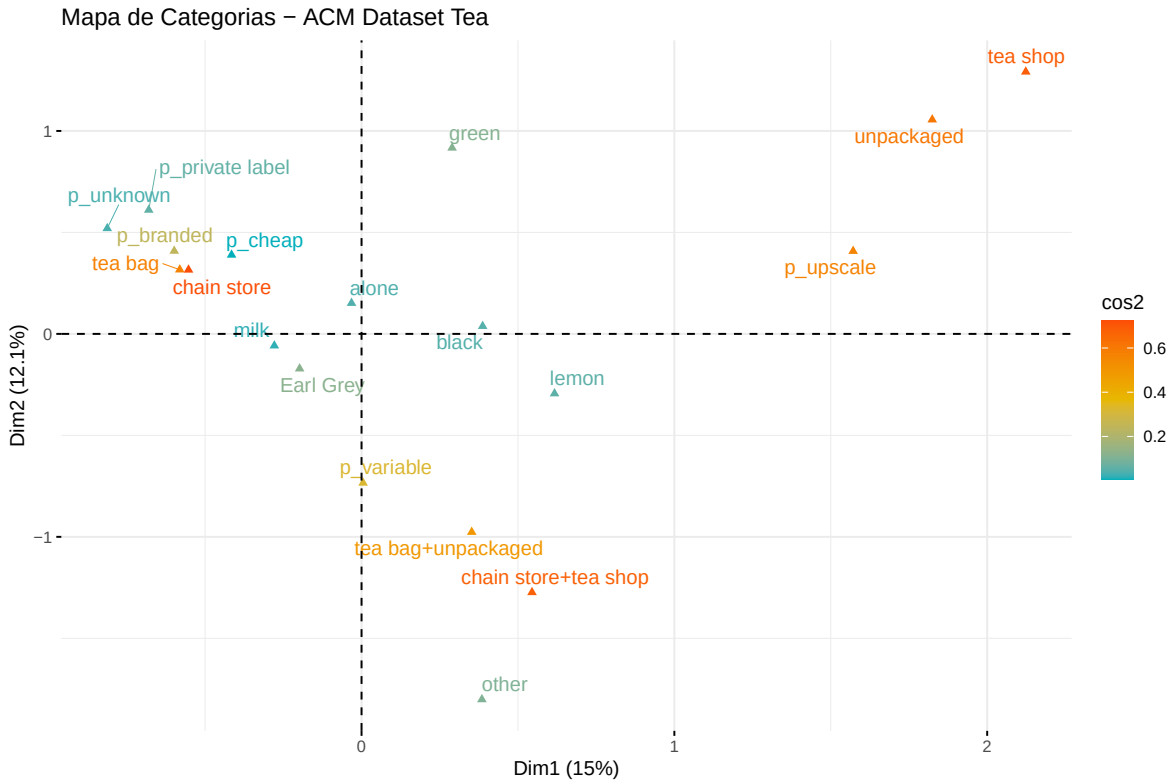

Figura 19.7: Mapa de categorias (ACM) para o dataset tea. A cor indica a qualidade de representação ($\cos^2$) de cada categoria no plano.

Interpretação do Mapa de Categorias:

O mapa revela a estrutura de associação entre os hábitos de consumo de chá dos 300 respondentes. As cores indicam a qualidade de representação ($\cos^2$): categorias em **vermelho/laranja** estão bem representadas no plano e podem ser interpretadas com confiança; categorias em **azul** têm parte de sua variabilidade em dimensões superiores.

19.2.3 Dimensão 1 (Eixo Horizontal): Local de Compra e Tipo de Produto

Este é o principal eixo de diferenciação, separando consumidores por **onde compram** e **que tipo de chá consomem**:

LADO ESQUERDO (valores negativos):

- `chain store` (supermercados/redes): Bem representado (cor laranja/vermelha)
- `tea bag` (sachês): Posicionado próximo a `chain store`

- **Interpretação:** Consumidores que compram sachês em supermercados - perfil de conveniência e praticidade

LADO DIREITO (valores positivos):

- `tea shop` (lojas especializadas): Bem representado (cor vermelha)
- `unpackaged` (chá a granel): Fortemente associado a `tea shop`
- **Interpretação:** Apreciadores que buscam chás de qualidade em lojas especializadas - perfil de conhecedor

Associação chave: `chain store` + `tea bag` vs. `tea shop` + `unpackaged` são perfis opostos, confirmando dois segmentos distintos de mercado.

19.2.4 Dimensão 2 (Eixo Vertical): Tipo de Chá e Consistência de Perfil

A segunda dimensão diferencia os consumidores quanto ao **tipo de chá preferido** e **consistência do comportamento**:

PARTE SUPERIOR (valores positivos fortes):

- `green` (+0.92): Chá verde
- `tea shop` (+1.29), `unpackaged` (+1.06): Reforçam a associação com chá verde
- **Interpretação:** Perfil consistente de consumidores que preferem chá verde, geralmente comprado a granel em lojas especializadas, consumido puro

PARTE INFERIOR (valores negativos fortes):

- `p_variable` (-0.73): Preço variável/"depende"
- `tea bag+unpackaged` (-0.98): Combinação de categorias opostas
- `chain store+tea shop` (-1.27): Compra em ambos os canais
- `other` (-1.80): Outro tipo de chá (não especificado)
- **Interpretação:** Comportamentos **inconsistentes ou mistos** - consumidores que não têm um perfil bem definido, misturam canais e formatos, ou não se encaixam nas categorias principais

Significado da dimensão: Separa consumidores com **perfil bem definido** (extremos) de consumidores com **comportamento misto/variável** (centro e parte inferior)

19.2.5 Perfis de Consumidor e Implicações Estratégicas:

A ACM revelou **dois segmentos principais** claramente separados pela Dimensão 1:

1. Perfil "Premium/Conhecedor" (lado direito do mapa):

- **Onde compra:** Lojas especializadas (`tea shop`)
- **Tipo de chá:** A granel (`unpackaged`)

- **Características:** Bem representados no mapa (cor vermelha), indicando comportamento consistente
- **Estratégia:** Valoriza qualidade e variedade. Marketing deve enfatizar origem, tipos especiais, experiência de compra

2. Perfil "Prático/Conveniência" (lado esquerdo do mapa):

- **Onde compra:** Supermercados (`chain store`)
- **Tipo de chá:** Sachês (`tea bag`)
- **Características:** Bem representados no mapa (cor laranja), comportamento consistente
- **Estratégia:** Valoriza praticidade e disponibilidade. Marketing deve enfatizar conveniência, custo-benefício

3. Categorias com baixa qualidade de representação (próximas à origem, cor azul):

- Algumas categorias de how (forma de preparo) e `Tea` (tipo de chá específico)
- **Interpretação:** Estas variáveis têm perfis relativamente homogêneos entre os consumidores, não sendo fortes diferenciadores de segmentos
- Sua variabilidade pode estar em dimensões superiores ou simplesmente ser menor

Insights acionáveis:

- A **escolha do canal de venda** (especializado vs. supermercado) e **formato do produto** (granel vs. sachê) definem a segmentação primária
- Estes dois atributos estão fortemente correlacionados: quem compra em tea shops prefere unpackaged, e vice-versa
- Estratégias de distribuição e precificação devem ser diferenciadas para cada segmento
- O segmento premium é menor mas potencialmente mais lucrativo; o segmento prático tem maior volume mas margens menores

19.2.6 Sumário da Análise

A ACM revelou insights claros sobre a estrutura do mercado de chá:

Principais descobertas:

1. **Segmentação em dois perfis distintos:**
    - **Perfil "Premium":** Compra em lojas especializadas (`tea shop`), prefere chá a granel (`unpackaged`), disposto a pagar mais (`p_upscale`)
    - **Perfil "Prático":** Compra em supermercados (`chain store`), usa sachês (`tea bag`), mais sensível a preço
2. **Associações validadas:**

- `tea shop` + `unpackaged` + `p_upscale` formam um cluster coerente (consumidor premium)
- `chain store` + `tea bag` formam o cluster de conveniência
- Estas associações foram identificadas empiricamente nos dados de 300 consumidores

3. **Duas dimensões de diferenciação:**

   - **Dimensão 1 (principal):** Local de compra e formato do produto (`tea shop`+`unpackaged` vs. `chain store`+`tea bag`)
   - **Dimensão 2 (secundária):** Tipo de chá e forma de consumo (chá verde puro vs. chá preto com aditivos)
   - O perfil premium no quadrante superior direito combina: loja especializada + chá a granel + chá verde
   - Marketing deve considerar ambas dimensões: canal de distribuição E tipo de produto oferecido

20 Análise de Correlação Canônica com Pinguins

Neste exemplo, aplicamos a Análise de Correlação Canônica (ACC) para investigar a relação entre medidas do bico e medidas corporais dos pinguins do arquipélago Palmer. Utilizamos Python e o dataset `palmerpenguins`, que contém medidas de três espécies (Adelie, Chinstrap, Gentoo).

Conjuntos de variáveis:

- **Grupo 1 - Bico** ($\mathbf{X}^{(1)}$): `bill_length_mm` (comprimento, na direção do bico) e `bill_depth_mm` (profundidade/altura, medida vertical do bico)
- **Grupo 2 - Corpo** ($\mathbf{X}^{(2)}$): `flipper_length_mm` (nadadeira) e `body_mass_g` (massa)

Objetivo: Quantificar a associação entre a forma do bico e o tamanho corporal, e interpretar essa relação.

20.1 Preparação e Análise Exploratória

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from scipy import linalg
from scipy.stats import chi2 as chi2_dist

sns.set_theme(style="whitegrid")

# Carregar e limpar dados
penguins = sns.load_dataset("penguins")
penguins_clean = penguins.dropna()

# Definir os dois grupos
X = penguins_clean[['bill_length_mm', 'bill_depth_mm']]
Y = penguins_clean[['flipper_length_mm', 'body_mass_g']]
```

```
# Padronizar (trabalharemos com a matriz de correlação)
X_std = (X - X.mean()) / X.std()
Y_std = (Y - Y.mean()) / Y.std()

print(f"Amostra: n = {len(penguins_clean)}, p = {X.shape[1]}, q = {Y.shape[1]}")
```

```
Amostra: n = 333, p = 2, q = 2
```

Antes da ACC, visualizamos a matriz de correlação para verificar as relações entre e dentro dos grupos:

```
combined = pd.concat([X_std, Y_std], axis=1)
corr_matrix = combined.corr()

plt.figure(figsize=(8, 6))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1, fmt=".2f",
            cbar_kws={'label': 'Correlação'})
plt.title("Correlações Intra e Inter-Grupos")
plt.tight_layout()
plt.show()
```

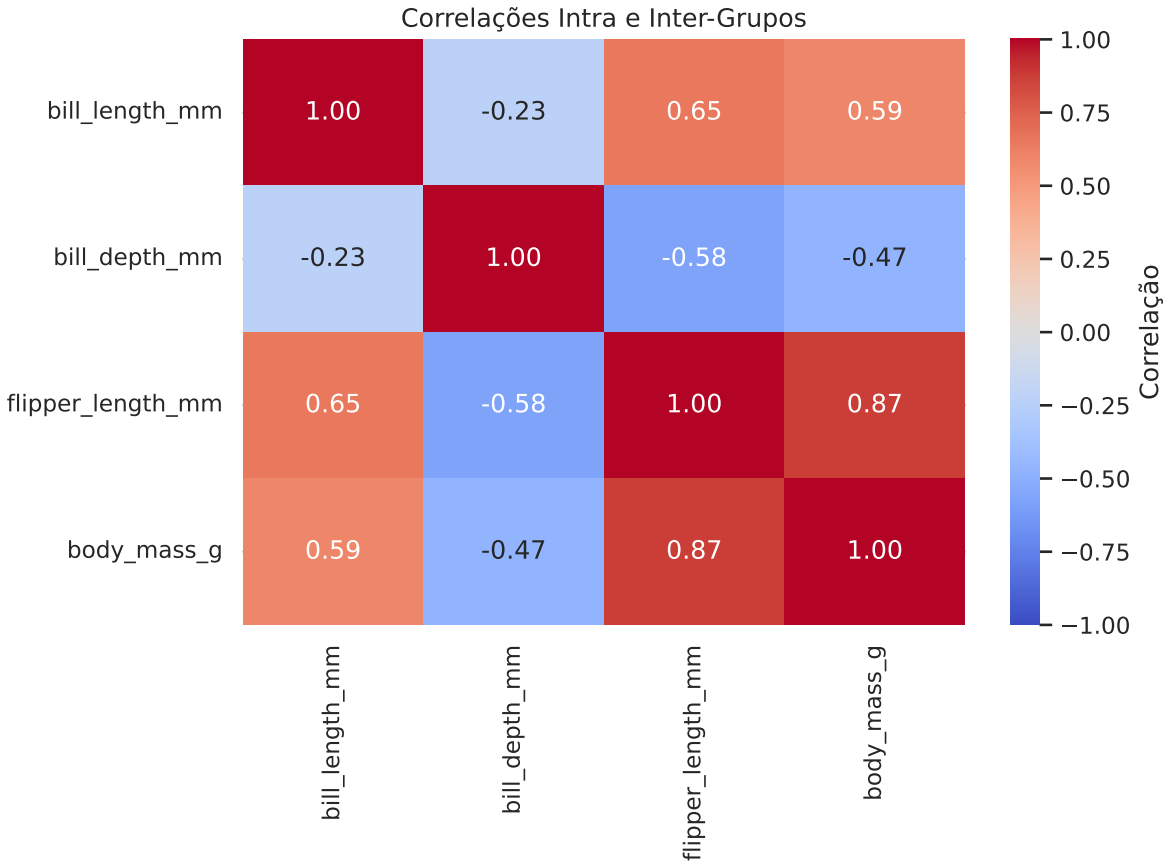

Figura 20.1: Matriz de correlação entre as variáveis dos dois grupos.

Observamos correlações cruzadas substanciais: `flipper_length_mm` correlaciona positivamente com `bill_length_mm` (0.66) e negativamente com `bill_depth_mm` (-0.58). A ACC sintetizará essas relações.

20.2 Cálculo da ACC via SVD

Seguimos a abordagem via SVD apresentada em Capítulo 12. Particionamos a matriz de correlação $\mathbf{R}$ e aplicamos SVD à matriz de coerência $\mathbf{K} = \mathbf{R}_{11}^{-1/2}\mathbf{R}_{12}\mathbf{R}_{22}^{-1/2}$:

```
R = corr_matrix.values
p, q = X.shape[1], Y.shape[1]

# Partições da matriz de correlação
```

```
R11 = R[:p, :p]
R22 = R[p:, p:]
R12 = R[:p, p:]
R21 = R[p:, :p]

# Branqueamento: inversas das raízes quadradas
R11_inv_sqrt = linalg.fractional_matrix_power(R11, -0.5)
R22_inv_sqrt = linalg.fractional_matrix_power(R22, -0.5)

# Matriz de coerência K
K = R11_inv_sqrt @ R12 @ R22_inv_sqrt

# SVD de K
U_k, canonical_corrs, Vh_k = linalg.svd(K)

# Coeficientes canônicos (revertendo o branqueamento)
A = R11_inv_sqrt @ U_k
B = R22_inv_sqrt @ Vh_k.T

# Ajuste de sinal para interpretação intuitiva:
# SVD pode retornar vetores com sinal arbitrário. Para facilitar a interpretação,
# invertemos os sinais de modo que a primeira variável do corpo (flipper_length)
# tenha carga positiva na primeira dimensão canônica.
if (R22 @ B[:, 0])[0] < 0:
    A[:, 0] = -A[:, 0]
    B[:, 0] = -B[:, 0]

print("Correlações Canônicas (\rho*):")
for i, rho in enumerate(canonical_corrs, 1):
    print(f"  rho_{i}* = {rho:.4f}")
```

```
Correlações Canônicas (ho^*):
  rho_1* = 0.7876
  rho_2* = 0.0864
```

As duas correlações canônicas obtidas ($\rho_1^* = 0.7876$ e $\rho_2^* = 0.0864$) revelam que apenas a primeira dimensão é substancial (> 0.3), conforme discutido em Capítulo 12. A segunda é desprezível, sugerindo que a relação entre os dois grupos se concentra essencialmente em uma única dimensão latente.

> **Nota sobre Sinais das Variáveis Canônicas**
>
> Os sinais dos vetores canônicos retornados pela SVD são arbitrários - multiplicar ambos U_k e V_k por -1 resulta na mesma correlação canônica. Para facilitar a interpretação, ajustamos os sinais de modo que valores positivos de V_1 correspondam a pinguins maiores.

20.3 Interpretação via Cargas Canônicas

Conforme a teoria, os **coeficientes** $\mathbf{a}_k$ e $\mathbf{b}_k$ são difíceis de interpretar devido ao branqueamento. Calculamos as **cargas canônicas** (correlações entre variáveis originais e canônicas) e **cargas cruzadas** (correlações entre variáveis de um grupo e canônicas do outro):

```
# Cargas canônicas (l_X,k e l_Y,k)
loadings_X = R11 @ A  # Corr(X(1), U)
loadings_Y = R22 @ B  # Corr(X(2), V)

# Cargas cruzadas (l_X->Y,k e l_Y->X,k)
cross_loadings_X = R12 @ B  # Corr(X(1), V)
cross_loadings_Y = R21 @ A  # Corr(X(2), U)

# Criar DataFrames para visualização
df_loadings_X = pd.DataFrame(loadings_X, index=X.columns, columns=['U1', 'U2'])
df_loadings_Y = pd.DataFrame(loadings_Y, index=Y.columns, columns=['V1', 'V2'])

print("Cargas Canônicas - Primeira Dimensão:")
print("\nGrupo Bico (correlação com U_1):")
print(df_loadings_X['U1'].round(3))
print("\nGrupo Corpo (correlação com V_1):")
print(df_loadings_Y['V1'].round(3))
```

```
Cargas Canônicas - Primeira Dimensão:

Grupo Bico (correlação com U_1):
bill_length_mm    0.828
bill_depth_mm    -0.735
Name: U1, dtype: float64

Grupo Corpo (correlação com V_1):
flipper_length_mm    1.000
body_mass_g          0.866
Name: V1, dtype: float64
```

Visualização das cargas:

```
fig, axes = plt.subplots(1, 2, figsize=(10, 4))

# Grupo Bico
sns.barplot(x=df_loadings_X['U1'], y=df_loadings_X.index, ax=axes[0],
            color='steelblue', edgecolor='black')
axes[0].set_title('Cargas Canônicas: U_1 (Bico)', fontsize=12, fontweight='bold')
axes[0].set_xlabel('Correlação com U_1')
axes[0].set_ylabel('')
axes[0].set_xlim(-1, 1)
axes[0].axvline(0, color='black', linewidth=0.8)

# Grupo Corpo
sns.barplot(x=df_loadings_Y['V1'], y=df_loadings_Y.index, ax=axes[1],
            color='coral', edgecolor='black')
axes[1].set_title('Cargas Canônicas: V_1 (Corpo)', fontsize=12, fontweight='bold')
axes[1].set_xlabel('Correlação com V_1')
axes[1].set_ylabel('')
axes[1].set_xlim(-0.5, 1.1)
axes[1].axvline(0, color='black', linewidth=0.8)

plt.tight_layout()
plt.show()
```

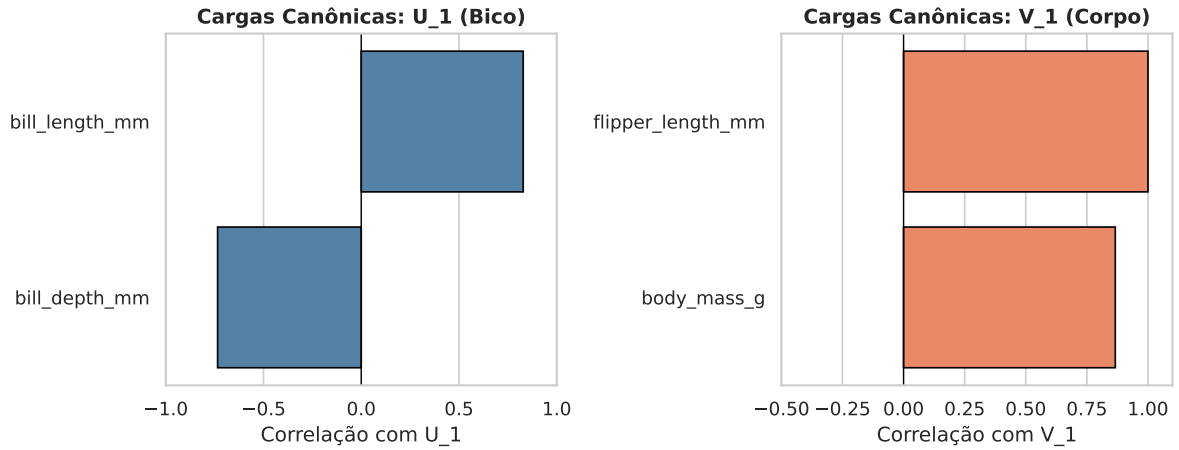

Figura 20.2: Cargas canônicas da primeira dimensão. Valores indicam correlação entre variáveis originais e variáveis canônicas.

Interpretação:

Observando as cargas canônicas:

- **Dimensão Canônica do Bico (U_1):** `bill_length_mm` (comprimento do bico) tem carga positiva elevada (+0.83), enquanto `bill_depth_mm` (altura/espessura do bico) tem carga negativa (-0.74). Isso define um **contraste morfológico**:
  - U_1 **positivo** -> bicos **longos e finos** (alto comprimento, baixa profundidade)
  - U_1 **negativo** -> bicos **curtos e grossos** (baixo comprimento, alta profundidade)
- **Dimensão Canônica do Corpo (V_1):** Ambas as variáveis têm cargas positivas elevadas (`flipper_length_mm`: +1.00, `body_mass_g`: +0.87), representando o **tamanho geral**.
  - V_1 **positivo** -> pinguins **maiores**
  - V_1 **negativo** -> pinguins **menores**

A correlação canônica de 0.79 indica forte associação positiva: pinguins com bicos longos e finos (U_1 alto) tendem a ser maiores (V_1 alto), enquanto pinguins com bicos curtos e grossos (U_1 baixo) tendem a ser menores (V_1 baixo).

Índice de redundância: Conforme discutido em Capítulo 12, o índice de redundância mede quanto da variância de um grupo é explicada pela variável canônica do *outro* grupo:

```
# Variância extraída (média dos quadrados das cargas cruzadas)
var_ext_X = np.mean(cross_loadings_X[:, 0]**2)
var_ext_Y = np.mean(cross_loadings_Y[:, 0]**2)

# Redundância = Variância Extraída × \rho2
red_X = var_ext_X * (canonical_corrs[0]**2)
red_Y = var_ext_Y * (canonical_corrs[0]**2)

print(f"Grupo Bico: Var. Extraída = {var_ext_X:.3f}, Redundância = {red_X:.3f}")
print(f"Grupo Corpo: Var. Extraída = {var_ext_Y:.3f}, Redundância = {red_Y:.3f}")
```

```
Grupo Bico: Var. Extraída = 0.380, Redundância = 0.236
Grupo Corpo: Var. Extraída = 0.543, Redundância = 0.337
```

Interpretação da redundância:

- **Variância Extraída**: Indica quanto da variância de um grupo é capturada pela correlação com a variável canônica do outro grupo. Por exemplo, 54.3% da variância das medidas corporais (V_1) correlaciona com as medidas do bico (U_1).
- **Redundância**: Ajusta a variância extraída pela correlação canônica ao quadrado. Aqui, a dimensão canônica do bico (U_1) explica **33.7% da variância total** das medidas corporais (0.543 × 0.79^2 = 0.337). Reciprocamente, V_1 explica apenas 23.6% da variância das medidas do bico, indicando que o tamanho corporal é mais previsível a partir da forma do bico do que o inverso.

20.4 Visualização no Espaço Canônico

Calculamos os scores das variáveis canônicas para cada observação e visualizamos a distribuição das espécies:

```
# Scores canônicos
U = X_std @ A
V = Y_std @ B

scores_df = pd.DataFrame({
    'U1': U.iloc[:, 0],
    'V1': V.iloc[:, 0],
    'Species': penguins_clean['species']
})

plt.figure(figsize=(8, 5))
sns.scatterplot(data=scores_df, x='U1', y='V1', hue='Species', style='Species',
                s=100, alpha=0.7, palette='Set2')

# Linha representando a correlação canônica
x_vals = np.array([scores_df['U1'].min(), scores_df['U1'].max()])
plt.plot(x_vals, x_vals * canonical_corrs[0], color='gray', linestyle='--',
         linewidth=2, label=f'rho_1* = {canonical_corrs[0]:.2f}')

plt.xlabel('U_1 (Bico: Curto/Grosso <-> Longo/Fino)', fontsize=11)
plt.ylabel('V_1 (Corpo: Menor <-> Maior)', fontsize=11)
plt.title('Espaço Canônico: Forma do Bico vs. Tamanho Corporal', fontsize=12, fontweight=
plt.legend(title='Espécie', fontsize=10)
plt.grid(alpha=0.3)
plt.tight_layout()
plt.show()
```

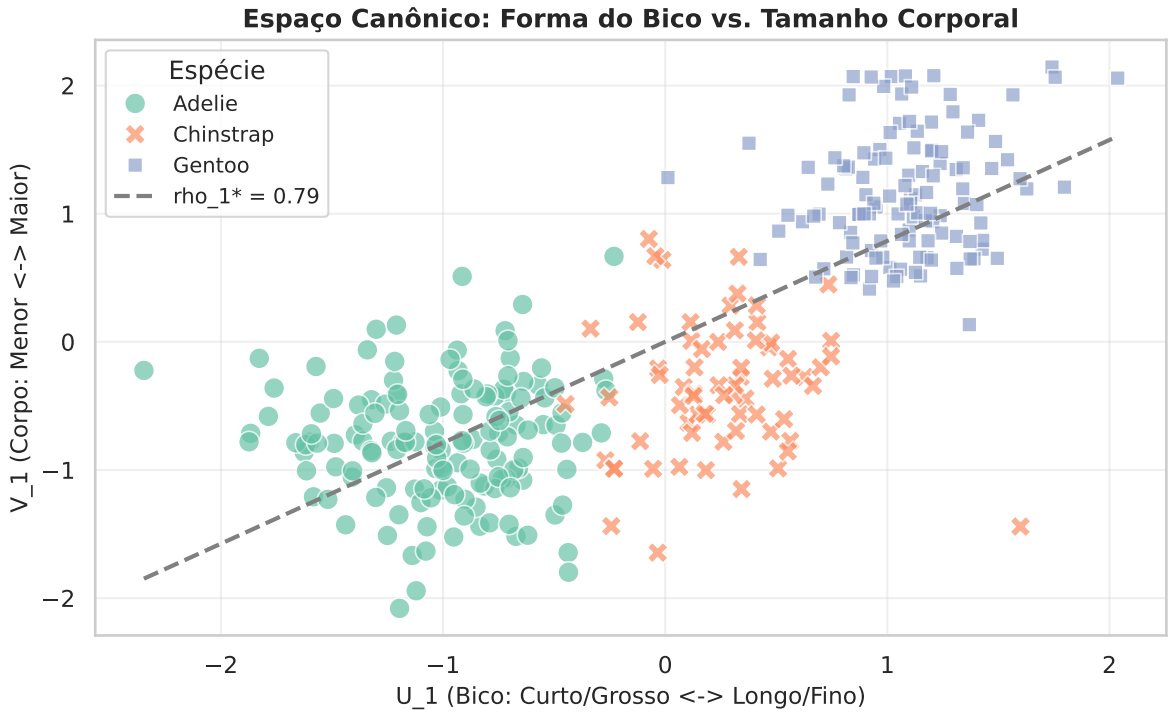

Figura 20.3: Distribuição das espécies no espaço das variáveis canônicas. A correlação canônica de 0.79 é evidente na linearidade, e a separação das espécies revela diferenças morfológicas.

O gráfico confirma:

1. **Correlação linear forte** entre U_1 e V_1 (linha tracejada)
2. **Gentoo** (maiores, bicos longos/finos) no quadrante superior direito
3. **Adelie** (menores, bicos curtos/grossos) no quadrante inferior esquerdo
4. **Chinstrap** em posição intermediária

20.5 Teste de Significância

Conforme Capítulo 12, testamos $H_0 : \boldsymbol{\Sigma}_{12} = \mathbf{0}$ usando a estatística Lambda de Wilks e a aproximação qui-quadrado de Bartlett:

```
n = len(penguins_clean)

# Lambda de Wilks
wilks_lambda = np.prod(1 - canonical_corrs**2)
```

```
# Estatística qui-quadrado de Bartlett
chi2_stat = -(n - 1 - (p + q + 1) / 2) * np.log(wilks_lambda)
df = p * q
p_value = 1 - chi2_dist.cdf(chi2_stat, df)

print(f"Lambda de Wilks (Lambda): {wilks_lambda:.4f}")
print(f"Qui-quadrado: chi2 = {chi2_stat:.2f}, gl = {df}")
print(f"P-valor: {p_value:.2e}")
print(f"\nConclusão: {'Rejeita H_0' if p_value < 0.05 else 'Não rejeita H_0'} (alpha = 0.
```

```
Lambda de Wilks (Lambda): 0.3768
Qui-quadrado: chi^2 = 321.60, gl = 4
P-valor: 0.00e+00

Conclusão: Rejeita H_0 (alpha = 0.05)
```

O p-valor extremamente baixo ($p < 0.001$) rejeita a hipótese de independência. A associação entre forma do bico e tamanho corporal é estatisticamente significativa.

Síntese: A ACC revelou uma dimensão canônica dominante ($\rho_1^* = 0.79$) que relaciona a morfologia do bico (contraste comprimento/profundidade) ao tamanho corporal geral. Essa estrutura reflete diferenças ecológicas entre as espécies de pinguins, com Gentoo exibindo um fenótipo distinto (maior porte, bicos alongados) comparado a Adelie.