**Revista Eletrônica
Paulista de Matemática**

ISSN 2316-9664
Volume 20, jul. 2021
Iniciação Científica

Rafael Froner Prando

Instituto de Geociências
e Ciências Exatas
UNESP-Universidade Estadual
Paulista “Júlio de Mesquita Filho”
rafael.froner@unesp.br

Marta Cilene Gadotti

Instituto de Geociências
e Ciências Exatas
UNESP-Universidade Estadual
Paulista “Júlio de Mesquita Filho”
mc.gadotti@unesp.br

Uma aplicação da Álgebra Linear na representação de imagens

An application of Linear Algebra on image representation

Resumo

Com o advento da computação moderna e o crescimento exponencial da capacidade de processamento de novas máquinas, a Álgebra Linear tem se mostrado uma ferramenta de extrema utilidade quando é necessário lidar com grandes volumes de informação, cenário esse que tem se tornado cada vez mais comum. A Álgebra Linear possui uma miríade de ferramentas e técnicas capazes de otimizar uma variedade de processos computacionais. Nesse trabalho exploraremos uma dessas técnicas aplicada no contexto de representação de imagens, apesar de poder ser usada para qualquer coleção de informações que podem ser expressas na forma matricial. Para isso, será necessário apresentar generalizações de conceitos da Álgebra Linear clássica como o Teorema de Mudança de Base e a Decomposição Espectral para o caso de matrizes não quadradas. As imagens exemplo apresentadas no texto foram confeccionadas com a ajuda de um *script* na linguagem Python 3.7.3 que implementa a técnica discutida.

Palavras-chave: Aplicação. Autovalores e Autovetores. Cálculos Matriciais.

Abstract

With the advent of modern computation and the exponential growth of the processing power of new machines, Linear Algebra has proven itself to be an extremely useful tool when it comes to dealing with large sets of information. Linear Algebra disposes of a myriad of tools and techniques capable of optimizing a variety of computational processes. In this text we explore one of these techniques applied on the context of image representation, despite it being applicable on any information set in matrix form. For this method to work on any image, we will need to present generalizations for some classic concepts of Linear Algebra such as the Change of Basis Theorem and Spectral Decomposition for non-square matrices. The example images presented were made using a Python 3.7.3 script that implements the discussed technique.

Keywords: Application. Eigenvalues and Eigenvectors. Matrix Computations.

1 Introdução

Em muitos casos quando lidamos com grandes volumes de informação, é benéfico que a simplifiquemos um pouco antes de processá-la e transmiti-la a fim de sustentar uma boa velocidade de processamento.

Evidentemente, nem sempre conseguiremos manter toda a informação intacta depois desse processo de simplificação. No entanto, esse é um preço que muitas vezes vale a pena ser pago. Seja pela priorização da velocidade de processamento ou a não necessidade de preservar absolutamente todos os detalhes de um conjunto de informações.

Sendo assim, construiremos com ferramentas de Álgebra Linear tais como o Teorema de Mudança de Base, a Decomposição em Valores Singulares e o Teorema Espectral um procedimento para representar qualquer conjunto de dados em formato matricial com apenas uma fração das informações iniciais segundo a referência [1]. Em particular, esse procedimento será aplicado em imagens no formato .png.

2 As bases da álgebra linear

Sabemos que em um espaço vetorial de dimensão n , todos os vetores podem ser expressos como combinação linear dos n vetores de uma base. Entretanto, alguns vetores específicos não necessitam de todos os vetores da base para serem expressos se escolhermos bem tal base. Essa será a observação fundamental que nos permitirá simplificar informações em formato matricial.

2.1 O teorema de mudança de base

O Teorema de Mudança de Base é a ferramenta que nos permitirá reconstruir (pelo menos parcialmente) a matriz original usando apenas uma parte da informação nela contida.

Cabe destacar que este resultado, assim como diversos outros que serão apresentados, são generalizações de resultados clássicos da Álgebra Linear para matrizes não quadradas.

Teorema 1 (Teorema de Mudança de Base) *Seja U uma matriz $m \times n$ com entradas reais, $m \geq n$ e que possui n vetores coluna que formam uma base ortonormal de um subespaço H de $\mathbb{R}^m$. Então, para $x \in H$, sua representação n -dimensional é o vetor coordenada $c \in \mathbb{R}^n$ dado por $c = U^t x$. A reconstrução de x dadas essas coordenadas é $x = Uc$. Portanto, se $x \in H$, $x = UU^t x$.*

Vejamos um exemplo de aplicação do Teorema de Mudança de Base:

Exemplo 2 Sejam $x_1 = \begin{bmatrix} 7 \\ -1 \\ 4 \\ -1 \end{bmatrix}$ e $U = \begin{bmatrix} \frac{1}{\sqrt{11}} & \frac{3}{\sqrt{14}} \\ -\frac{1}{\sqrt{11}} & 0 \\ 0 & \frac{2}{\sqrt{14}} \\ -\frac{3}{\sqrt{11}} & \frac{1}{\sqrt{14}} \end{bmatrix}$. Temos $m = 4$, $n = 2$, logo $m > n$. Podemos separar U em $u_1 = \begin{bmatrix} \frac{1}{\sqrt{11}} \\ -\frac{1}{\sqrt{11}} \\ 0 \\ -\frac{3}{\sqrt{11}} \end{bmatrix}$ e $u_2 = \begin{bmatrix} \frac{3}{\sqrt{14}} \\ 0 \\ \frac{2}{\sqrt{14}} \\ \frac{1}{\sqrt{14}} \end{bmatrix}$. Evidentemente, $\|u_1\| = \|u_2\| = 1$ e $\langle u_1, u_2 \rangle = 0$, ou seja,

$\{u_1, u_2\}$ é base ortonormal de um subespaço H de $\mathbb{R}^4$. Além disso, veja que $x_1 \in H$, pois podemos escrever $x_1 = \sqrt{11} \cdot u_1 + 2\sqrt{14} \cdot u_2$.

Assim, aplicamos o Teorema de Mudança de Base para encontrar uma representação de x_1 em menor dimensão:

$$c = U^t x_1 = \begin{bmatrix} \frac{1}{\sqrt{11}} & -\frac{1}{\sqrt{11}} & 0 & -\frac{3}{\sqrt{11}} \\ \frac{3}{\sqrt{14}} & 0 & \frac{2}{\sqrt{14}} & \frac{1}{\sqrt{14}} \end{bmatrix} \begin{bmatrix} 7 \\ -1 \\ 4 \\ -1 \end{bmatrix} = \begin{bmatrix} \frac{11}{\sqrt{11}} & \frac{28}{\sqrt{14}} \end{bmatrix} = [\sqrt{11} \quad 2\sqrt{14}].$$

Logo, as coordenadas de x_1 na base $\{u_1, u_2\}$ são $(\sqrt{11}, 2\sqrt{14})$ como observado anteriormente. Para reconstruir x_1 , calculamos:

$$Uc = \begin{bmatrix} \frac{1}{\sqrt{11}} & \frac{3}{\sqrt{14}} \\ -\frac{1}{\sqrt{11}} & 0 \\ 0 & \frac{2}{\sqrt{14}} \\ -\frac{3}{\sqrt{11}} & \frac{1}{\sqrt{14}} \end{bmatrix} [\sqrt{11} \quad 2\sqrt{14}] = \begin{bmatrix} 7 \\ -1 \\ 4 \\ -1 \end{bmatrix} = x_1.$$

Note que a reconstrução é idêntica a x_1 somente porque $x_1 \in H$.

Exemplo 3 Considere agora $x_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$. Temos que $x_2 \notin H$ já que os zeros na segunda e terceira

coordenadas poderiam apenas ser obtidos se os escalares associados aos vetores da base fossem ambos nulos. Dessa forma, quando calculamos sua reconstrução, obtemos um vetor diferente do original. De fato:

$$\hat{x}_2 = UU^t x_2 = \begin{bmatrix} \frac{1}{\sqrt{11}} & \frac{3}{\sqrt{14}} \\ -\frac{1}{\sqrt{11}} & 0 \\ 0 & \frac{2}{\sqrt{14}} \\ -\frac{3}{\sqrt{11}} & \frac{1}{\sqrt{14}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{11}} & -\frac{1}{\sqrt{11}} & 0 & -\frac{3}{\sqrt{11}} \\ \frac{3}{\sqrt{14}} & 0 & \frac{2}{\sqrt{14}} & \frac{1}{\sqrt{14}} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.88961039 \\ 0.18181818 \\ 0.71428571 \\ 1.25974026 \end{bmatrix} \neq x_2.$$

A princípio pode não parecer tão útil escrever x em função de si mesmo. Porém, veja que se $x \notin H$, o vetor obtido é uma projeção de x sobre o espaço gerado pelas colunas de U . Chamemos essa projeção de $\hat{x}$, portanto, $\hat{x} = UU^t x$. Essa será a fórmula de projeção que utilizaremos para reconstruir a imagem original.

Sendo assim, a questão agora se torna: como encontrar a base que forneça a melhor aproximação de um vetor fora do subespaço H com a quantidade mínima de vetores?

2.2 Decomposição em valores singulares

Para encontrarmos a base ideal, ou seja, a base que minimiza o erro no processo de projeção, o teorema da decomposição em valores singulares é de extrema importância. Contudo, antes de enunciá-lo e mostrar sua utilidade, vejamos alguns conceitos e resultados preliminares como apresentados nas referências [1] e [2].

Definição 4 Uma matriz A de dimensões $n \times n$ é dita diagonalizável se pode ser escrita como $A = PDP^{-1}$ onde P é a matriz quadrada dos autovetores de A e D é a matriz diagonal dos autovalores associados. Uma matriz A é dita ortogonalmente diagonalizável se podemos realizar tal decomposição como $A = PDP^t$.

Teorema 5 (Teorema Espectral) Para uma matriz simétrica com entradas reais A e dimensões $n \times n$, valem as seguintes propriedades:

- (a) A possui n autovalores reais.
- (b) A dimensão do espaço gerado por cada autovetor é igual à sua multiplicidade algébrica no polinômio característico de A .
- (c) Os autovetores de A são ortogonais dois a dois.
- (d) A é ortogonalmente diagonalizável.

Do item (d), podemos reescrever A como $A = PDP^t$. Como existem n autovalores reais, tomemos eles como sendo $\{\lambda_1, \dots, \lambda_n\}$ e os autovetores correspondentes $\{u_1, \dots, u_n\}$. Portanto,

$$A = U\Lambda U^t = \begin{bmatrix} u_1 & \cdots & u_n \end{bmatrix} \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} \begin{bmatrix} u_1^t \\ \vdots \\ u_n^t \end{bmatrix} = \lambda_1 u_1 u_1^t + \cdots + \lambda_n u_n u_n^t$$

é a decomposição espectral da matriz A .

Decomposição em Valores Singulares (DVS) é um processo semelhante à diagonalização que pode ser usado em qualquer matriz, seja ela quadrada ou não.

Sendo assim, tome A uma matriz $m \times n$ com $m \neq n$. Veja que a matriz $A^t A$ é quadrada de dimensões $n \times n$ e simétrica, já que $(A^t A)^t = A^t (A^t)^t = A^t A$. A matriz AA^t também é quadrada e simétrica, porém com dimensões $m \times m$.

Usando o Teorema Espectral, concluímos que $A^t A$ possui autovalores $\{\lambda_1, \dots, \lambda_n\}$ e autovetores $\{v_1, \dots, v_n\}$. Também inferimos que AA^t possui autovalores $\{\delta_1, \dots, \delta_m\}$ e autovetores $\{u_1, \dots, u_m\}$. Sendo ambas ortogonalmente diagonalizáveis, podemos escrever $A^t A = V\Lambda V^t$ e $AA^t = U\Delta U^t$.

Veja que, por definição, $A^t A v_i = \lambda_i v_i$. Multiplicando ambos os lados por A na esquerda, temos:

$$(AA^t)Av_i = A\lambda_i v_i = \lambda_i Av_i.$$

Portanto, Av_i é autovetor de AA^t com o mesmo autovalor λ_i . Assim, concluímos que $Av_i = u_i$. Se assumirmos v_i e u_i unitários, devemos dividir por $\|Av_i\|$ para manter a norma 1.

Veja que $\|Av_i\|^2 = (Av_i)^t Av_i = v_i^t (A^t Av_i) = v_i^t \lambda_i v_i = \lambda_i v_i^t v_i$. Como $\|v_i\|^2 = v_i^t v_i$, temos que

$$\|Av_i\|^2 = \lambda_i \|v_i\|^2 = \lambda_i \implies \|Av_i\| = \sqrt{\lambda_i}.$$

Denotamos esses valores de $\|Av_i\|$ por σ_i e os chamamos de valores singulares. Assim, $u_i = \frac{Av_i}{\sigma_i}$ é unitário como queríamos.

Convencionaremos a partir desse ponto que os valores singulares serão sempre apresentados em ordem decrescente.

Temos então:

$$\begin{aligned} V &= \begin{bmatrix} v_1 & \cdots & v_n \end{bmatrix} \\ U &= \begin{bmatrix} u_1 & \cdots & u_m \end{bmatrix} \\ AV &= \begin{bmatrix} Av_1 & \cdots & Av_m \end{bmatrix} \\ &= \begin{bmatrix} \sigma_1 u_1 & \cdots & \sigma_m u_m \end{bmatrix}. \end{aligned}$$

Se tomarmos Σ como sendo a matriz $m \times n$ com valores singulares na diagonal e zeros em todas as outras entradas, podemos escrever $AV = U\Sigma$. Por fim, como V é ortogonal, temos que $VV^t = I$ e, portanto, $A = U\Sigma V^t$. Sendo assim, chegamos ao resultado desejado.

Teorema 6 (Decomposição em Valores Singulares) *Uma matriz arbitrária A pode ser decomposta como $A = U\Sigma V^t$ com*

$$\begin{aligned} V &= \begin{bmatrix} v_1 & \cdots & v_n \end{bmatrix} \\ U &= \begin{bmatrix} u_1 & \cdots & u_m \end{bmatrix} \\ D &= \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_m \end{bmatrix} \\ \Sigma &= \begin{bmatrix} D & & 0 \\ & \ddots & \\ 0 & & 0 \end{bmatrix} \end{aligned}$$

Vejamos um exemplo de utilização da DVS:

Exemplo 7 Tome $A = \begin{bmatrix} 1 & 2 & 3 \\ -2 & 1 & 1 \end{bmatrix}$. Temos

$$\det(A^t A - \lambda I) = \begin{vmatrix} 5 - \lambda & 0 & 1 \\ 0 & 5 - \lambda & 7 \\ 1 & 7 & 10 - \lambda \end{vmatrix} = -\lambda^3 + 20\lambda^2 - 75\lambda = -\lambda(15 - \lambda)(5 - \lambda).$$

Portanto, $A^t A$ tem autovalores $\lambda_1 = 15$, $\lambda_2 = 5$, $\lambda_3 = 0$ com autovetores normalizados:

$$v_1 = \begin{bmatrix} \frac{1}{\sqrt{150}} \\ \frac{7}{\sqrt{150}} \\ \frac{10}{\sqrt{150}} \end{bmatrix} \quad v_2 = \begin{bmatrix} -\frac{7}{\sqrt{50}} \\ \frac{1}{\sqrt{50}} \\ 0 \end{bmatrix} \quad v_3 = \begin{bmatrix} \frac{1}{\sqrt{75}} \\ \frac{7}{\sqrt{75}} \\ -\frac{5}{\sqrt{75}} \end{bmatrix}.$$

Veja que $\{v_1, v_2, v_3\}$ é um conjunto ortonormal e que

$$V^t = \begin{bmatrix} \frac{1}{\sqrt{150}} & \frac{7}{\sqrt{150}} & \frac{10}{\sqrt{150}} \\ -\frac{7}{\sqrt{50}} & \frac{1}{\sqrt{50}} & 0 \\ \frac{1}{\sqrt{75}} & \frac{7}{\sqrt{75}} & -\frac{5}{\sqrt{75}} \end{bmatrix}.$$

Os valores singulares de $A^t A$ são $\sigma_1 = \sqrt{15}$, $\sigma_2 = \sqrt{5}$, $\sigma_3 = 0$. Construimos então

$$D = \begin{bmatrix} \sqrt{15} & 0 \\ 0 & \sqrt{5} \end{bmatrix} \text{ e } \Sigma = \begin{bmatrix} \sqrt{15} & 0 & 0 \\ 0 & \sqrt{5} & 0 \end{bmatrix}.$$

Por fim, calculamos u_1 e u_2

$$\begin{aligned} u_1 &= \frac{Av_1}{\sigma_1} = \frac{1}{\sqrt{15}} \begin{bmatrix} \frac{45}{\sqrt{150}} \\ \frac{15}{\sqrt{150}} \\ \frac{15}{\sqrt{150}} \end{bmatrix} = \begin{bmatrix} \frac{3}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} \end{bmatrix} \\ u_2 &= \frac{Av_2}{\sigma_2} = \frac{1}{\sqrt{5}} \begin{bmatrix} -\frac{5}{\sqrt{50}} \\ \frac{15}{\sqrt{50}} \\ \frac{15}{\sqrt{50}} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\sqrt{10}} \\ \frac{3}{\sqrt{10}} \\ \frac{3}{\sqrt{10}} \end{bmatrix} \end{aligned}$$

e construímos U :

$$U = \begin{bmatrix} \frac{3}{\sqrt{10}} & -\frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{3}{\sqrt{10}} \end{bmatrix}.$$

Portanto, a decomposição de A é:

$$A = U\Sigma V^t = \begin{bmatrix} \frac{3}{\sqrt{10}} & -\frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{3}{\sqrt{10}} \end{bmatrix} \begin{bmatrix} \sqrt{15} & 0 & 0 \\ 0 & \sqrt{5} & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{150}} & \frac{7}{\sqrt{150}} & \frac{10}{\sqrt{150}} \\ -\frac{7}{\sqrt{50}} & \frac{1}{\sqrt{50}} & 0 \\ \frac{1}{\sqrt{75}} & \frac{7}{\sqrt{75}} & -\frac{5}{\sqrt{75}} \end{bmatrix}.$$

Veja que quando não há m autovalores não nulos, D pode ter dimensões menores. Σ , no entanto, nunca é alterada. Isso nos leva a pensar que valores queremos incluir na Decomposição por Valores Singulares. Chamamos a decomposição de completa quando consideramos todos os autovalores. Nesse caso, Σ é $m \times n$ e U, V são quadradas com todos os seus vetores.

Porém, algumas vezes os autovalores nulos não são tão interessantes. Chamamos uma Decomposição em Valores Singulares de reduzida se existirem apenas r autovalores não nulos. Teríamos U com dimensões $m \times r$, V com dimensões $n \times r$ e D com dimensões $r \times r$. Veja, no entanto, que $r \leq \min\{m, n\}$, pois U e V têm o mesmo número de autovalores não nulos.

A Decomposição em Valores Singulares será uma ferramenta essencial na busca pela melhor base. Sendo aliado a alguns conceitos que apresentaremos a seguir, esse resultado é o que permite descrever a melhor base explicitamente.

2.3 A matriz de covariância

Definição 8 Para A uma matriz $m \times n$ com entradas reais, definimos a matriz de covariância C como

$$C = \frac{1}{n-1} AA^t.$$

Essa definição assume que a soma dos elementos de cada linha é 0.

Veja que podemos facilmente conseguir uma matriz equivalente à A cuja soma dos elementos de cada linha é 0 calculando a média $\bar{a}_i = \frac{1}{n} \sum_{k=1}^n a_{ik}$ de cada linha e a subtraindo dessa linha.

A matriz de covariância mede o quanto duas colunas de A estão relacionadas. Como A possui m vetores linha, existem m^2 possibilidades de comparação (incluindo testar um vetor contra si mesmo). c_{ij} testa a correlação do vetor na i -ésima linha contra o vetor da j -ésima coluna. Essa é uma razão pela qual C deve ser uma matriz simétrica $m \times m$.

Note que também podemos definir a matriz de covariância como

$$C = \frac{1}{n-1} A^t A$$

para testar a correlação entre os vetores coluna. Usaremos, no entanto apenas a primeira definição da matriz de covariância no desenvolvimento subsequente.

Substituindo a Decomposição por Valores Singulares de A na primeira definição para a matriz de covariância resulta em

$$\begin{aligned} C &= \frac{1}{n-1} (U \Sigma V^t) (V \Sigma^t U^t) \\ &= \frac{1}{n-1} U \Sigma (V^t V) \Sigma^t U^t \\ &= U \left(\frac{1}{n-1} \Sigma^2 \right) U^t. \end{aligned}$$

Cabe ressaltar que como C é simétrica, ela pode ser escrita como PDP^t onde P é ortonormal e D é diagonal com os autovalores de C . Sabemos por construção que U é ortonormal e que Σ^2 é diagonal. Assim, $\frac{1}{n-1} \Sigma^2$ possui em sua diagonal os autovalores de C .

Como Σ é originária da Decomposição em Valores Singulares de A , notamos que é possível calcular os autovalores e autovetores da matriz de covariância usando a DVS de A . Se γ_i é autovalor da matriz de covariância, tem-se que:

$$\gamma_i = \frac{1}{n-1} \sigma_i^2.$$

Da natureza dos σ_i , vemos que os autovalores da matriz de covariância são os mesmos que os da matriz original divididos por $n-1$.

2.4 A melhor base

Suponha A uma matriz formada por n vetores coluna de $\mathbb{R}^m$ cuja representação matricial tenha a soma de cada linha igual a zero como pede a matriz de covariância. Projetando essas informações sobre um vetor unitário u , a variância S_u^2 é dada por

$$S_u^2 = u^t C u,$$

onde C é a matriz de covariância.

Quando u é o primeiro autovetor de C : $v_1^t C v_1 = v_1^t \lambda_1 v_1 = \lambda_1 \|v_1\|^2$. Como v_1 é unitário: $v_1^t C v_1 = \lambda_1$. Sendo assim, a variância projetada num autovetor é correspondente ao autovalor associado.

Agora queremos estimar um vetor x usando partes de uma base ortonormal $\{u_1, \dots, u_n\}$. Temos a combinação linear

$$x = c_1 u_1 + \dots + c_n u_n.$$

O vetor coordenada $c = (c_1, \dots, c_n)$ pode ser reescrito como $U^t x$:

$$x = u_1^t x u_1 + \dots + u_n^t x u_n.$$

Calculemos agora o produto escalar de ambos os lados por x . Como $\langle x, x \rangle = \|x\|^2$ e sabendo que U é ortonormal, qualquer termo com dois vetores diferentes de U se anula, ou seja:

$$\|x\|^2 = (u_1^t x)(x^t u_1) + \dots + (u_n^t x)(x^t u_n).$$

Nos foquemos primeiro em representar x com apenas um vetor da base. Sendo assim, vamos ignorar todos os termos dessa soma exceto um e introduzir um erro. Temos:

$$\|x\|^2 = u_i^t x x^t u_i + \|x_{err}\|^2.$$

Dividindo ambos os lados por $n - 1$ e lembrando a fórmula da matriz de covariância, vemos que

$$\frac{1}{n-1} \|x\|^2 = u_i^t C u_i + \frac{1}{n-1} \|x_{err}\|^2.$$

Note que $u_i^t C u_i$ nada mais é que a variância de x projetada sobre o vetor u_i da base ortonormal. Como nessa última expressão o termo da esquerda é constante e os dois termos da direita dependem da escolha do u_i , minimizar o erro é equivalente a maximizar a variância.

Como C é simétrica, seus autovetores formam uma base ortonormal de $\mathbb{R}^n$. Pelo Teorema de Mudança de Base, $u = Vc$ onde c é vetor constante e V é a matriz com os autovetores de C nas colunas. Também $C = VDV^t$ pelo Teorema Espectral. Assim:

$$u^t C u = (Vc)^t (VDV^t) (Vc) = c^t (V^t V) D (V^t V c) = c^t D c.$$

Como D é a matriz diagonal dos autovalores de C , temos

$$c^t D c = c_1^2 \lambda_1 + \dots + c_n^2 \lambda_n.$$

Como $\|c^t D c\|^2 = \sum c_i^2 = 1$, para maximizar a variância usando apenas um vetor, basta encontrar o maior autovalor λ_j da matriz de covariância e tomar $c_j^2 = 1$ e todos os outros $c_i = 0$.

Portanto, a melhor base de dimensão 1 é formada pelo autovetor associado ao maior autovalor da matriz de covariância. E em geral, a melhor base k -dimensional é formada pelos k autovetores associados aos k maiores autovalores.

3 Implementação do método

Tendo em mãos uma caracterização explícita para a melhor base, é relativamente simples implementar um método de compressão que faz uso das propriedades dessa base no contexto de imagens, por exemplo. Isso porque podemos estabelecer uma simples sequência de manipulações matriciais que nos fornecerá o resultado desejado.

Primeiramente, separamos os canais de cor (vermelho, verde e azul) da imagem original e os convertemos para matrizes. Temos então 3 matrizes $m \times n$ com entradas reais no intervalo $[0, 255]$, uma vez que o sistema RGB se utiliza de 3 valores nesse intervalo para definir todas as cores.

Calculamos a média de cada linha, salvamos essa matriz e a subtraímos da matriz original. Veja que é necessário salvar a matriz com as médias de cada linha, pois o processo aproxima apenas matrizes cuja soma das linhas é 0. Se quisermos obter uma aproximação da matriz original, é necessário somar as médias de volta ao final do processo.

Selecionamos então a quantidade k de autovetores que serão utilizados na reconstrução. Depois disso computamos a matriz de covariância $C = \frac{1}{n-1} A A^t$ e encontramos seus autovalores e autovetores. Esses autovetores são então normalizados e postos em ordem decrescente de acordo com seus autovalores associados.

Depois disso, juntamos os k primeiros autovetores para formar uma matriz M de dimensões $m \times k$. Sobre o espaço gerado pelas colunas de M projetamos cada coluna x da matriz pela equação descrita no Teorema de Mudança de Base:

$$M_{m \times k} M_{k \times m}^t x_{m \times 1} = \hat{x}_{m \times 1}.$$

Com as n colunas reconstruídas, as juntamos para formar uma matriz $m \times n$ que é a reconstrução da matriz com soma das linhas igual a zero. Para terminar o processo, basta somar a matriz com as médias de cada linha novamente.

Esse processo é repetido para as três matrizes representando as camadas de cor. Por fim, as três matrizes são reunidas e convertidas de volta para uma imagem.

Em termos computacionais, esse passo a passo é implementado com um *script* na linguagem Python 3.7.3 com assistência da biblioteca NumPy para manipulações matriciais e da *Python Imaging Library* para conversão e processamento das imagens.

Cabe notar que é possível ajustar o número de autovetores da reconstrução separadamente para cada uma das camadas. Sendo assim, uma cor menos presente em certa imagem pode ser reconstruída com menos autovetores sem perdas notáveis de qualidade.

Vejamos exemplos da técnica aplicada em algumas imagens para podermos avaliar sua eficácia assim como suas limitações. Considere a imagem de uma ponte estaiada na cidade de São José dos Campos - SP (veja Figura 1):

Figura 1: Ponte Estaiada (Edher de Souza, 2020)

Essa imagem tem dimensões 1260×650 . Assim, são necessários 650 vetores para representar a imagem perfeitamente.

Apresentamos as reconstruções usando, respectivamente, $k = 3, 10, 30, 50, 100, 300$ autovetores na reconstrução de cada camada de cor (veja Figura 2):

(a) $k = 3$

(b) $k = 10$

(c) $k = 30$

(d) $k = 50$

(e) $k = 100$

(f) $k = 300$

Figura 2: Reconstruções (elaborada pelos autores)

Podemos ver que em $k = 100$, a imagem já é facilmente reconhecível apesar de termos utilizado apenas uma fração da informação da imagem original. As formas e contornos da imagem são rapidamente definidas conforme aumentamos o número de autovetores.

Quanto às cores, os defeitos de compressão são facilmente observáveis e demoram um pouco mais para se tornarem menos notáveis. Portanto, imagens em escala de cinza, por exemplo, são bastante propícias para a aplicação desse método, visto que a gama de cores desse tipo de imagem é severamente limitada.

A fim de examinar um pouco melhor como o método lida com cores, tomemos agora uma imagem com contornos simples e um fundo branco: o logo da Revista Eletrônica Paulista de Matemática.

As dimensões dessa figura são 760×760 . Apresentamos as reconstruções com $k = 3, 6, 10, 30, 50, 100, 300, 400$ e 500 autovetores:

Figura 3: Logo CQD (Revista Eletrônica Paulista de Matemática)

(a) $k = 3$

(b) $k = 6$

(c) $k = 10$

Figura 4: Reconstruções (elaborada pelos autores)

Nesse exemplo observamos que todos os contornos e cores exceto o branco estão bem definidos por volta de $k = 50$. No entanto, o fundo branco só começa a se parecer com o original por volta de $k = 400$.

Note também que adicionamos 20 autovetores quando variamos k de 10 para 30 e de 30 para 50. No entanto, a melhoria na qualidade da reconstrução é mais notável quando fizemos a mudança de 10 para 30 o que exemplifica o fato dos autovetores associados aos maiores autovalores serem os que carregam mais informação da imagem original.

4 Conclusões

Esses exemplos ilustram que é possível preservar grande parte das informações iniciais contidas na matriz usando apenas uma parcela dos dados originais. Se escolhermos adequadamente a base para realizar a projeção dos vetores, conseguimos reconstruir muito mais do que inicialmente pode parecer possível.

Os exemplos apresentados evidenciam que o método produz alguns artefatos de compressão na forma de manchas de cor em alguns locais. Posto em um contexto de cores muito escuras (asfalto na Figura 2) ou muito claras (fundo branco na Figura 4), esses artefatos aparecem mais perceptíveis nas figuras, dadas as cores que o cercam.

Na parte computacional, o programa poderia receber uma série de otimizações para que rodasse mais rapidamente e utilizasse menos recursos. Além disso, o *script* poderia ser portado para uma linguagem compilada, o que também aumentaria significativamente a velocidade de processamento. Na parte matemática, o estudo poderia prosseguir no sentido de apresentar outros tipos de

decomposição matricial como a decomposição de Cholesky e suas aplicações.

5 Referências bibliográficas

1. SCHAFER, Casey. **The neural network, its techniques and applications**. [S.l.:s.n.], 2016.
2. HOFFMAN, K.; KUNZE, R. **Álgebra linear**, 2a. edição. Livros Técnicos e Científicos Editora, Rio de Janeiro, 1979.