Kernel PCA

Advanced Institute for Artificial Intelligence – AI2

https://advancedinstitute.ai

Introdução

Kernel PCA (KPCA) é uma técnica que busca generalizar PCA visando realizar **redução não linear de dimensionalidade**. Lembrando que PCA assume que os dados estão em um espaço Euclidiano, ou seja, linear. Na hipótese que os dados estão em um espaço não linear, PCA não consegue garantir a sua otimalidade.

Seja, então, um conjunto de dados $\mathcal{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$ tal que $\boldsymbol{x}_i \in \mathbb{R}^n$ representa uma amostra no espaço de características original. A ideia do KPCA é fazer uso de uma função de mapeamento não linear, também conhecida por *kernel*, $\phi(\boldsymbol{x}) \in \mathbb{R}^{n'}$, tal que $n' > n$, visando mapear os dados para um espaço de maior dimensão. Geralmente, esses mapeamentos são custosos pois envolvem a projeção das amostras nestes espaços com maior dimensão. No entanto, podemos contornar isso com o *kernel trick*, em que podemos **calcular o produto interno** no espaço de alta dimensão **sem a necessidade** de projeção dos dados.

Desta forma, KPCA realiza um aumento temporário da dimensionalidade do espaço para depois realizar a sua redução. **A ideia é incorporar não linearidades durante o processo de redução de dimensionalidade.**

A primeira hipótese do KPCA é assumir que a média amostral após o mapeamento para o espaço de maior dimensão vale zero, ou seja:

$$\boldsymbol{\mu}' = \frac{1}{m} \sum_{i=1}^{m} \phi(\boldsymbol{x}_i) = 0, \tag{1}$$

em que $\boldsymbol{\mu}' \in \mathbb{R}^{n'}$. Neste caso, KPCA assume que a média amostral está situada na origem do espaço de dados.

Já a matriz de covariância $\boldsymbol{\Sigma}' \in \mathbb{R}^{n' \times n'}$ é dada por:

$$\boldsymbol{\Sigma}' = \frac{1}{m} \sum_{i=1}^{m} (\phi(\boldsymbol{x}_i) - \overset{0}{\cancel{\boldsymbol{\mu}'}})(\phi(\boldsymbol{x}_i) - \overset{0}{\cancel{\boldsymbol{\mu}'}})^T = \frac{1}{m} \sum_{i=1}^{m} \phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^T. \tag{2}$$

Temos que os autovetores $\boldsymbol{v}_k$ da matriz $\boldsymbol{\Sigma}$ são dados por:

$$\boldsymbol{\Sigma}' \boldsymbol{v}_k = \lambda \boldsymbol{v}_k, \quad \forall k = 1, 2, \ldots, n'. \tag{3}$$

Existe um teorema bastante importante que nos diz que os autovetores de $\boldsymbol{\Sigma}$ podem ser escritos como uma combinação linear das características, ou seja:

$$\boldsymbol{v}_k = \sum_{i=1}^{m} \alpha_{ki} \phi(\boldsymbol{x}_i). \tag{4}$$

Assim sendo, para calcularmos os autovetores da matriz de covariância $\boldsymbol{\Sigma}'$, basta sabermos os vetores $\boldsymbol{\alpha}_k$. Substituindo-se a Equação 2 na Equação 3, temos que:

$$\boldsymbol{\Sigma}' \boldsymbol{v}_k = \underbrace{\frac{1}{m} \sum_{i=1}^{m} \phi(\boldsymbol{x}_i) \phi(\boldsymbol{x}_i)^T}_{\boldsymbol{\Sigma}'} \boldsymbol{v}_k = \lambda \boldsymbol{v}_k. \tag{5}$$

A Equação 5 implica em:

$$\boldsymbol{v}_k = \frac{1}{m\lambda}\sum_{i=1}^{m}\phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^T\boldsymbol{v}_k = \frac{1}{m\lambda}\sum_{i=1}^{m}[\phi(\boldsymbol{x}_i)^T\boldsymbol{v}_k]\phi(\boldsymbol{x}_i) = \sum_{i=1}^{m}\alpha_{ki}\phi(\boldsymbol{x}_i), \tag{6}$$

em que $\alpha_{ki} = \frac{1}{m\lambda}\phi(\boldsymbol{x}_i)^T\boldsymbol{v}_k$. Agora, substituindo-se a Equação 6 na Equação 5, temos que:

$$\frac{1}{m}\sum_{i=1}^{m}\phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^T\left(\sum_{j=1}^{m}\alpha_{ki}\phi(\boldsymbol{x}_i)\right) = \lambda\sum_{j=1}^{m}\alpha_{kj}\phi(\boldsymbol{x}_j). \tag{7}$$

Reescrevendo a Equação 7, temos que:

$$\frac{1}{m}\sum_{i=1}^{m}\phi(\boldsymbol{x}_i)\left(\sum_{j=1}^{m}\alpha_{kj}\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)\right)=\lambda\sum_{j=1}^{m}\alpha_{kj}\phi(\boldsymbol{x}_j). \tag{8}$$

Fazendo uso do *kernel trick*, ou seja, $K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$ temos que:

$$\frac{1}{m}\sum_{i=1}^{m}\phi(\boldsymbol{x}_i)\left(\sum_{j=1}^{m}\alpha_{kj}K(\boldsymbol{x}_i,\boldsymbol{x}_j)\right)=\lambda\sum_{j=1}^{m}\alpha_{kj}\phi(\boldsymbol{x}_j). \tag{9}$$

Multiplicando ambos lados da Equação 9 por $\phi(\boldsymbol{x}_l)^T$, temos que:

$$\frac{1}{m}\sum_{i=1}^{m}\phi(\boldsymbol{x}_l)^T\phi(\boldsymbol{x}_i)\left(\sum_{j=1}^{m}\alpha_{kj}K(\boldsymbol{x}_i,\boldsymbol{x}_j)\right)=\lambda\sum_{j=1}^{m}\alpha_{kj}\phi(\boldsymbol{x}_l)^T\phi(\boldsymbol{x}_j). \tag{10}$$

Usando novamente o *kernel trick*, temos que:

$$\frac{1}{m}\sum_{i=1}^{m}K(\boldsymbol{x}_l,\boldsymbol{x}_i)\left(\sum_{j=1}^{m}\alpha_{kj}K(\boldsymbol{x}_i,\boldsymbol{x}_j)\right)=\lambda\sum_{j=1}^{m}\alpha_{kj}K(\boldsymbol{x}_l,\boldsymbol{x}_i). \tag{11}$$

Temos que nossa formulação está, agora, escrita em função da matriz de *kernel*.

Podemos expressar a Equação 11 em sua notação matriz-vetor, como segue:

$$\boldsymbol{K}^2\boldsymbol{\alpha}_k = (\lambda m)\boldsymbol{K}\boldsymbol{\alpha}_k. \tag{12}$$

Multiplicando-se ambos lados por $\boldsymbol{K}^{-1}$, temos que:

$$\boldsymbol{K}\boldsymbol{\alpha}_k = (\lambda m)\boldsymbol{\alpha}_k. \tag{13}$$

Quem são os $\boldsymbol{\alpha}_k$? São os **autovetores** da matriz de *kernel*!

Para uma dada amostra $\boldsymbol{x}$, calculamos a sua projeção na k-ésima componente da seguinte forma:

$$\hat{\boldsymbol{x}} = \phi(\boldsymbol{x})^T \boldsymbol{v}_k = \underbrace{\sum_{i=1}^{n} \alpha_{ki} \phi(\boldsymbol{x}_i)}_{\boldsymbol{v}_k} \phi(\boldsymbol{x})^T = \sum_{i=1}^{n} \alpha_{ki} K(\boldsymbol{x}, \boldsymbol{x}_i). \tag{14}$$

Basicamente, a projeção de uma amostra $\boldsymbol{x}$ na k-ésima componente resume-se a realizar o seu mapeamento para um espaço de maior dimensão com $\phi(\boldsymbol{x})$ e depois projetar em $\boldsymbol{v}_k$. **A vantagem do *kernel trick* é que não precisamos calcular explicitamente $\phi(\boldsymbol{x}_i)$, $\forall i$, basta apenas computar a matriz de *kernel* diretamente do conjunto de dados.**

Quais tipos de *kernel* podemos utilizar? Os mais empregados são:

- Polinomial: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^T \boldsymbol{x}_j + c)^d$, em que $c \geq 0$ é uma constante e d o grau do polinômio.
- Gaussiano (RBF): $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left\{-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2}\right\}$.

Note, também, que se o dado projetado não tiver média nula, precisamos centralizá-lo da seguinte maneira:

$$\tilde{\phi}(\boldsymbol{x}_i) = \phi(\boldsymbol{x}_i) - \frac{1}{m}\sum_{j=1}^{m}\phi(\boldsymbol{x}_j). \tag{15}$$

Já a matriz *kernel* correspondente (centralizada) é dada por:

$$\begin{aligned}
\tilde{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \tilde{\phi}(\boldsymbol{x}_i)^T \tilde{\phi}(\boldsymbol{x}_j) = \left(\phi(\boldsymbol{x}_i) - \frac{1}{m} \sum_{k=1}^{m} \phi(\boldsymbol{x}_k) \right)^T \left(\phi(\boldsymbol{x}_j) - \frac{1}{m} \sum_{k=1}^{m} \phi(\boldsymbol{x}_k) \right) \\
&= \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) - \frac{1}{m} \sum_{k=1}^{m} \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_k) - \frac{1}{m} \sum_{k=1}^{m} \phi(\boldsymbol{x}_k)^T \phi(\boldsymbol{x}_j) \\
&\quad + \frac{1}{m^2} \sum_{k=1}^{m} \sum_{l=1}^{m} \phi(\boldsymbol{x}_k)^T \phi(\boldsymbol{x}_l) \\
&= K(\boldsymbol{x}_i, \boldsymbol{x}_j) - \frac{1}{m} \sum_{k=1}^{m} K(\boldsymbol{x}_i, \boldsymbol{x}_j) - \frac{1}{m} \sum_{k=1}^{m} K(\boldsymbol{x}_k, \boldsymbol{x}_j) \\
&\quad + \frac{1}{n^2} \sum_{k=1}^{m} \sum_{l=1}^{m} K(\boldsymbol{x}_k, \boldsymbol{x}_l).
\end{aligned} \tag{16}$$

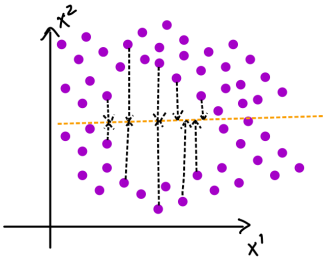
Em sua forma matricial, podemos escrever a Equação 16 como segue (matriz Gram):

$$\tilde{\boldsymbol{K}} = \boldsymbol{K} - 1_m \boldsymbol{K} - \boldsymbol{K} 1_m + 1_m \boldsymbol{K} 1_m, \tag{17}$$

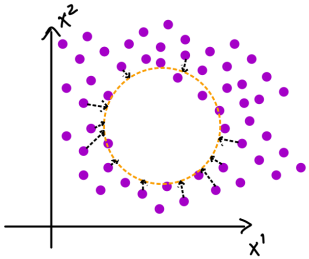
em que $1_n \in \mathbb{R}^{m \times m}$ é uma matriz com todos elementos iguais à $1/m$. Vejamos, agora, o algoritmo do KPCA:

1. Construir matriz *kernel* $\boldsymbol{K}$ a partir dos dados, em que $K_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$.
2. Construir matriz Gram $\tilde{\boldsymbol{K}}$ utilizando a Equação 17.
3. Utilize Equação 13 para encontrar os autovetores $\boldsymbol{\alpha}_k$ (use $\tilde{\boldsymbol{K}}$).
4. Calcule os d componentes principais utilizando a Equação 14.

Exemplos de projeções utilizando PCA e KPCA.



PCA



KPCA

Um agradecimento especial ao **Prof. Alexandre Levada** do Centro de Ciências Exatas e de Tecnologia, Departamento de Computação, Universidade Federal de São Carlos, pelas notas de aula.