Singular Value Decomposition (SVD) and Principal Component Analysis (PCA)

Shusen Wang
Stevens Institute of Technology

September 1, 2019

Abstract

This lecture note covers the following content: (1) singular value decomposition (SVD), (2) power iteration for computing truncated SVD, and (3) principal component analysis (PCA).

1 Orthonormal Basis

A set of vectors $\{\mathbf{v}_1, \cdots, \mathbf{v}_d\} \subset \mathbb{R}^d$ forms an **orthonormal basis** of the vector space $\mathbb{R}^d$ if

- the vectors have unit ℓ_2-norm: $\|\mathbf{v}_i\|_2 = 1$ for all $i = 1$ to d;
- the vectors are orthogonal: $\mathbf{v}_i^T \mathbf{v}_j = 0$ for all $i \neq j$.

The standard basis $\{\mathbf{e}_1, \cdots, \mathbf{e}_d\} \subset \mathbb{R}^d$ is defined by

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \qquad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \cdots, \quad \mathbf{e}_d = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}.$$

The standard basis is an orthonormal basis of $\mathbb{R}^d$. The vector space $\mathbb{R}^d$ has infinitely many orthonormal basis.

Theorem 1.1. *Let $\{\mathbf{v}_1, \cdots, \mathbf{v}_d\}$ be any orthonormal basis of $\mathbb{R}^d$. Then any vector $\mathbf{x} \in \mathbb{R}^n$ can be expressed as a linear combination of $\mathbf{v}_1, \cdots, \mathbf{v}_d$:*

$$\mathbf{x} = \alpha_1 \mathbf{v}_1 + \cdots + \alpha_d \mathbf{v}_d,$$

for some $\alpha_1, \cdots, \alpha_d \in \mathbb{R}$.

Principal component analysis (PCA) is essentially a change of orthonormal basis—from some arbitrary basis to standard basis. Let $\mathbf{x}_1, \cdots, \mathbf{x}_n \in \mathbb{R}^d$ be a set of zero-mean vectors, that is,

$\sum_{i=1}^n \mathbf{x}_i = \mathbf{0}$. Let

$$\begin{aligned}
\mathbf{x}_1 &= \alpha_{1,1}\mathbf{v}_1 + \alpha_{1,2}\mathbf{v}_2 + \cdots + \alpha_{1,d}\mathbf{v}_d, \\
\mathbf{x}_2 &= \alpha_{2,1}\mathbf{v}_1 + \alpha_{2,2}\mathbf{v}_2 + \cdots + \alpha_{2,d}\mathbf{v}_d, \\
&\vdots \\
\mathbf{x}_n &= \alpha_{n,1}\mathbf{v}_1 + \alpha_{n,2}\mathbf{v}_2 + \cdots + \alpha_{n,d}\mathbf{v}_d.
\end{aligned}$$

PCA applies a rotation to every of $\mathbf{x}_1, \cdots, \mathbf{x}_n \in \mathbb{R}^d$ and gets $\mathbf{x}'_1, \cdots, \mathbf{x}'_n \in \mathbb{R}^d$ which can be written as:

$$\begin{aligned}
\mathbf{x}'_1 &= \beta_{1,1}\mathbf{e}_1 + \beta_{1,2}\mathbf{e}_2 + \cdots + \beta_{1,k}\mathbf{e}_k + \beta_{1,k+1}\mathbf{e}_{k+1} + \cdots + \beta_{1,d}\mathbf{e}_d, \\
\mathbf{x}'_2 &= \beta_{2,1}\mathbf{e}_1 + \beta_{2,2}\mathbf{e}_2 + \cdots + \beta_{2,k}\mathbf{e}_k + \beta_{2,k+1}\mathbf{e}_{k+1} + \cdots + \beta_{2,d}\mathbf{e}_d, \\
&\vdots \\
\mathbf{x}'_n &= \beta_{n,1}\mathbf{e}_1 + \beta_{n,2}\mathbf{e}_2 + \cdots + \beta_{n,k}\mathbf{e}_k + \beta_{n,k+1}\mathbf{e}_{k+1} + \cdots + \beta_{n,d}\mathbf{e}_d.
\end{aligned}$$

Here, k is an integer much smaller than d. The objective of the rotation is to make the red terms big and the blue terms small. In this way, discarding the blue terms will not cause much error. Singular value decomposition (SVD) is needed for performing such a rotation that minimizes the blue terms.

2 Singular Value Decomposition (SVD)

Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ be any matrix and $r = \text{rank}(\mathbf{A}) \le m, n$.[1] The SVD of $\mathbf{A}$ can be written as:

$$\mathbf{A} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \cdots + \sigma_r \mathbf{u}_r \mathbf{v}_r^T.$$

Here, we define the singular values and vectors:

- $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_r > 0$ are the singular values in the descending order;
- $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_r \in \mathbb{R}^m$ are the left singular vectors; they form an orthonormal basis of a subspace of $\mathbb{R}^m$;
- $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_r \in \mathbb{R}^n$ are the right singular vectors; they form an orthonormal basis of a subspace of $\mathbb{R}^n$.

Note that $\sigma_i \mathbf{u}_i \mathbf{v}_i^T$ is an $m \times n$ rank-one matrix. SVD can be expressed as the following matrix form:

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T.$$

Here, $\mathbf{U} = [\mathbf{u}_1, \cdots, \mathbf{u}_r] \in \mathbb{R}^{m\times r}$ and $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_r] \in \mathbb{R}^{n\times r}$ have orthonormal columns, and $\mathbf{\Sigma} = \mathsf{diag}(\sigma_1, \cdots, \sigma_r) \in \mathbb{R}^{r\times r}$ is a diagonal matrix. Note that $\mathbf{U}^T\mathbf{U} = \mathbf{I}_r$ and $\mathbf{V}^T\mathbf{V} = \mathbf{I}_r$.[2]

[1] Matrix rank is the maximum number of linearly independent column (or row) vectors.

[2] However, $\mathbf{U}\mathbf{U}^T \in \mathbb{R}^{m\times m}$ and $\mathbf{V}\mathbf{V}^T \in \mathbb{R}^{n\times n}$ are not identity matrices. They are called orthogonal projectors.

Matrix Frobenius norm and spectral norm can be equivalently expressed using singular values:

$$\begin{aligned} \|\mathbf{A}\|_F^2 &= \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2 = \sum_{i=1}^{r}\sigma_i^2, \\ \|\mathbf{A}\|_2 &= \max_{\mathbf{x}\neq 0}\frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sigma_1. \end{aligned}$$

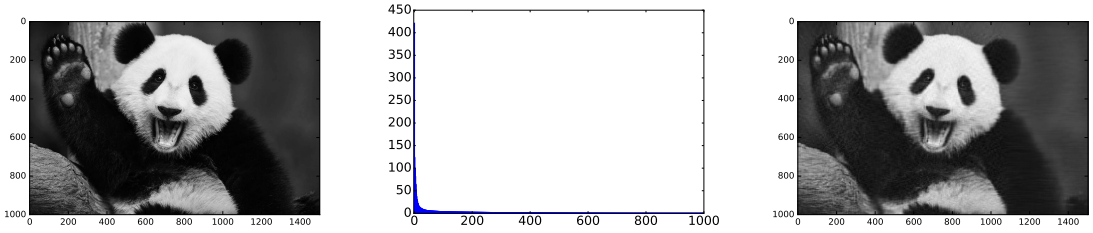
The rank-k ($k < r$) truncated SVD is

$$\mathbf{A}_k = \operatorname*{argmin}_{\mathrm{rank}(\mathbf{X})\leq k} \|\mathbf{A}-\mathbf{X}\|_F^2 = \sum_{i=1}^{k}\sigma_i\mathbf{u}_i\mathbf{v}_i^T.$$

It is the best rank-k approximation to $\mathbf{A}$. The approximation error is

$$\mathbf{A}-\mathbf{A}_k = \sum_{i=k+1}^{r}\sigma_i\mathbf{u}_i\mathbf{v}_i^T, \qquad \|\mathbf{A}-\mathbf{A}_k\|_F^2 = \sum_{i=k+1}^{r}\sigma_i^2, \qquad \|\mathbf{A}-\mathbf{A}_k\|_2 = \sigma_{k+1}.$$

Many types of matrices in the real world have fast decaying singular values. Figure 1(b) plots the singular values of an 1000×1500 matrix. In the case of fast decay, abandoning the bottom singular values and vectors has almost no impact. Figure 1(c) shows the rank-50 approximation, that is, $\mathbf{A}_{50} = \sum_{i=1}^{50}\sigma_i\mathbf{u}_i\mathbf{v}_i^T$. It shows that although 95% of the singular values and vectors are discarded, the matrix remains almost the same to the original.



(a) Original matrix. (b) Singular values. (c) Rank-50 approximation.

Figure 1: Illustration of singular values and truncated SVD.

3 Power Iteration for Truncated SVD

Power iteration. Given $\mathbf{A}\in\mathbb{R}^{m\times n}$ and an integer $k \ll m, n$, how do we compute the top k singular values and vectors of $\mathbf{A}$? A simple (but not very efficient) algorithm is power iteration:

1. Initialization: draw every entry of the vector $\mathbf{x}_0'\in\mathbb{R}^n$ from the Gaussian distribution $\mathcal{N}(0,1)$.
2. Normalization: $\mathbf{x}_0 \longleftarrow \mathbf{x}_0'/\|\mathbf{x}_0'\|_2$.
3. Repeat the two steps: $\mathbf{x}_q' \longleftarrow \mathbf{A}^T\mathbf{A}\mathbf{x}_{q-1}$ and $\mathbf{x}_q \longleftarrow \mathbf{x}_q'/\|\mathbf{x}_q'\|_2$.

If σ_1 is strictly greater than σ_2, then $\mathbf{x}_q$ will converge to $\mathbf{v}_1 \in \mathbb{R}^n$ (the first right singular vector of $\mathbf{A}$.) See Question 3.

With $\mathbf{v}_1$ known, we will immediately have $\sigma_1 \in \mathbb{R}_+$ and $\mathbf{u}_1 \in \mathbb{R}^m$:

$$\tilde{\mathbf{u}} \longleftarrow \mathbf{A}\mathbf{v}_1, \qquad \sigma_1 \longleftarrow \|\tilde{\mathbf{u}}\|_2, \qquad \mathbf{u}_1 \longleftarrow \tilde{\mathbf{u}}/\sigma_1.$$

The above can be proved using the facts that $\mathbf{A}\mathbf{v}_1 = \sigma_1\mathbf{u}_1$ and $\|\mathbf{u}_1\|_2 = 1$.

Deflation. The rest eigenvectors are obtained using deflation. Note that the matrix

$$\mathbf{A}_{-1} = \sum_{i=2}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$$

can be obtained by $\mathbf{A}_{-1} = \mathbf{A} - \sigma_1\mathbf{u}_1\mathbf{v}_1^T$. By applying the power iteration to $\mathbf{A}_{-1}$ (instead of $\mathbf{A}$), we will obtain the second singular value and vectors: σ_2, $\mathbf{u}_2$, and $\mathbf{v}_2$. Then, do the same to get $\mathbf{A}_{-2} = \mathbf{A}_{-1} - \sigma_2\mathbf{u}_2\mathbf{v}_2^T$ and then σ_3, $\mathbf{u}_3$, and $\mathbf{v}_3$. Repeat this process k times so we will get all the top k singular values and vectors.

Deflation has a fundamental problem. Note that power iteration is inaccurate, that is, the output of power iteration is not exactly $\mathbf{v}_1$. Instead, it is an approximation to $\mathbf{v}_1$, and the error decrease as q increases. After the deflation $\mathbf{A}_{-1} = \mathbf{A} - \sigma_1\mathbf{u}_1\mathbf{v}_1^T$, the error in approximating $\mathbf{v}_1$ will propagate to the approximation of $\mathbf{v}_2$, and then all the way to $\mathbf{v}_k$. Since σ_1 can be far greater than σ_k, a tiny error in computing $\mathbf{v}_1$ will propagate to $\mathbf{v}_k$ and ruin everything.

Block power iteration. A more practical algorithm is the block power iteration:

1. Initialization: draw every entry of the matrix $\mathbf{X}_0' \in \mathbb{R}^{n\times k}$ from the Gaussian distribution $\mathcal{N}(0,1)$.
2. Orthogonalize the columns: $\mathbf{X}_0 \longleftarrow \mathsf{orth}(\mathbf{X}_0')$ (so that its columns form an orthonormal basis.)
3. Repeat the two steps: $\mathbf{X}_q' \longleftarrow \mathbf{A}^T\mathbf{A}\mathbf{X}_{q-1}$ and $\mathbf{X}_q \longleftarrow \mathsf{orth}(\mathbf{X}_q')$.

It can be proved that the block power iteration converges to the top k right singular vectors of $\mathbf{A}$. The convergence rate depends on the spectral gap σ_k/σ_{k+1}; a large spectral gap makes convergence faster.

4 Principal Component Analysis (PCA)

Explaining PCA. Let $\mathbf{x}_1, \cdots, \mathbf{x}_n \in \mathbb{R}^d$ be a set of vectors and k be the target dimension. The goal of PCA is to find a linear map that transforms $\mathbf{x}_1, \cdots, \mathbf{x}_n$ to $\mathbf{z}_1, \cdots, \mathbf{z}_n \in \mathbb{R}^k$. PCA has four steps:

- First, make the data points (vectors) centered at the origin. Let $\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i$ be the sample mean. Perform the translation:

  $$\mathbf{x}_i' = \mathbf{x}_i - \hat{\boldsymbol{\mu}}, \qquad \text{for } i = 1, \cdots, n.$$

  See Figure 2 for illustration.

- Second, find the right singular vectors of $\mathbf{X}' = [\mathbf{x}'_1, \cdots, \mathbf{x}'_n]^T \in \mathbb{R}^{n \times d}$. Let $\mathbf{X}' = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ be the SVD of $\mathbf{X}'$. We will use $\mathbf{V} \in \mathbb{R}^{d \times d}$ to perform rotation.[3]

- Third, rotate $\mathbf{x}'_1, \cdots, \mathbf{x}'_n$:

$$\mathbf{x}''_i = \mathbf{V}^T \mathbf{x}'_i \in \mathbb{R}^d, \quad \text{for } i = 1, \cdots, n.$$

  See the illustration in Figure 3. Before the rotation, the maximal variance direction is aligned with $\mathbf{v}_1 \in \mathbb{R}^n$. After the rotation, the maximal variance direction is aligned with the first standard basis $\mathbf{e}_1 = [1, 0, 0, \cdots, 0]^T \in \mathbb{R}^d$.

- Last, dimensionality reduction: let $\mathbf{z}_i \in \mathbb{R}^k$ be the first k entries of $\mathbf{x}''_i \in \mathbb{R}^d$, for $i = 1$ to n. This means we keep only the top k variance directions.

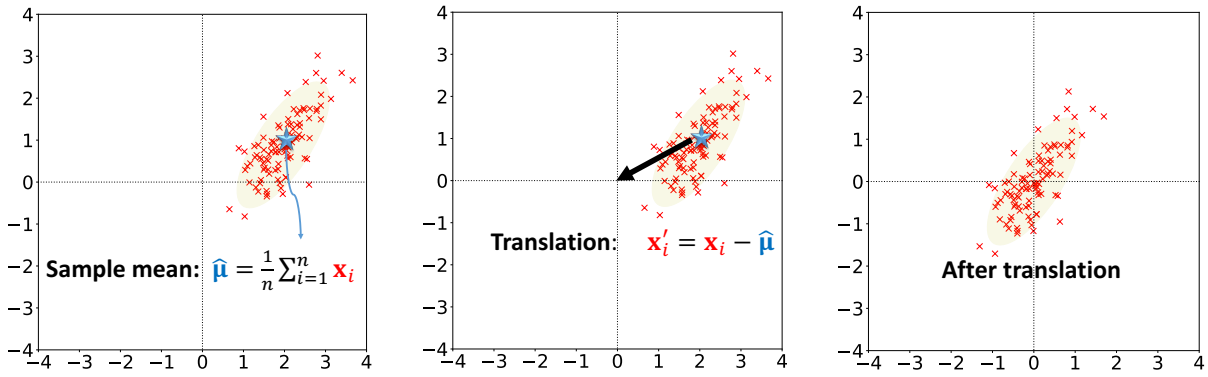


Figure 2: Step 1 of PCA: subtract the sample mean.

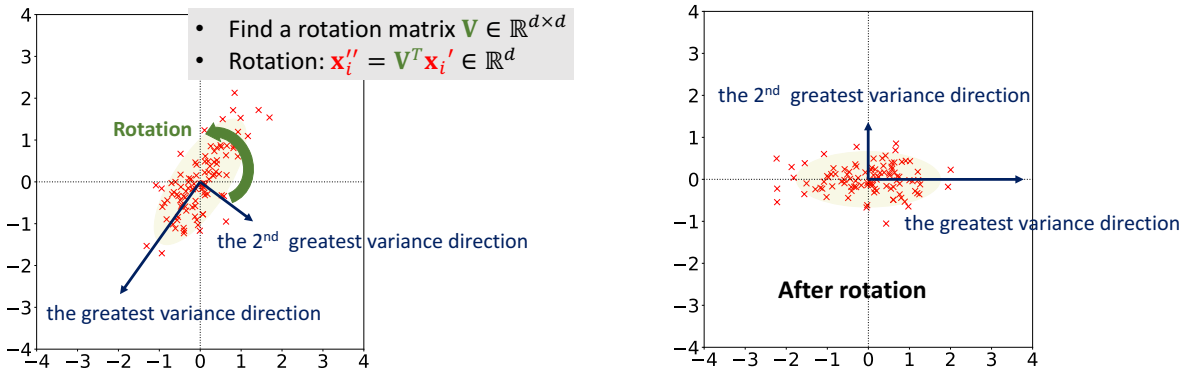


Figure 3: Step 3 of PCA: rotation.

[3]MATLAB's SVD function computes $\mathbf{X}' = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$; Numpy's SVD computes $\mathbf{X}' = \mathbf{U}\mathbf{\Sigma}\mathbf{V}$. Note the difference. Most books and papers (including this lecture note) express SVD in the same way as MATLAB.

Practical implementation. By closely examining the steps of PCA, we can find that we need only the top k right singular vectors of $\mathbf{X}' \in \mathbb{R}^{n\times d}$; the bottom singular vectors are not useful. Thus, in practice, PCA can the performed in the following more efficient way.

- The first step remains the same: make the data points centered at the origin.
- The second step becomes the truncated SVD: $\mathbf{X}' \approx \mathbf{X}'_k = \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^T$. Here, $\mathbf{V}_k \in \mathbb{R}^{d\times k}$ is the concatenation of the top k right singular vectors $\mathbf{v}_1, \cdots, \mathbf{v}_k \in \mathbb{R}^d$.
- The third and fourth steps becomes

$$\mathbf{z}_i \;=\; \mathbf{V}_k^T\mathbf{x}'_i \in \mathbb{R}^k, \qquad \text{for } i = 1, \cdots, n.$$

  In practice, you will use the matrix form: $\mathbf{Z} = \mathbf{X}'\mathbf{V}_k \in \mathbb{R}^{n\times k}$. The rows of $\mathbf{Z}$ are $\mathbf{z}_1, \cdots, \mathbf{z}_n$ computed in the above.

Why do we want to use the truncated SVD instead of the full (or thin) SVD? Computing the truncated SVD by block power iteration has $\mathcal{O}\big(ndk \log \frac{n}{\epsilon}\big)$ time complexity,[4] while computing the (thin) SVD costs $\mathcal{O}(nd^2)$ time. If k is much less than d, then the truncated SVD will be much cheaper to compute.

Why is $\mathbf{v}_1$ the greatest variance direction? In Figure 3(left), the greatest variance direction is $\mathbf{v}_1$. Why? It is because $\mathbf{v}_1$ is the solution to

$$\max_{\mathbf{w}\in\mathbb{R}^d} \mathbf{w}^T\mathbf{C}\mathbf{w}; \qquad \text{s.t. } \|\mathbf{w}\|_2 = 1.$$

See Question 5.

Why after the rotation, the greatest variance direction is the first standard basis $\mathbf{e}_1$? In Figure 3(right), the greatest variance direction is $\mathbf{e}_1 = [1, 0, \cdots, 0]^T \in \mathbb{R}^d$. Why? Let $\mathbf{X}' = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ be the SVD of $\mathbf{X}'$. Let $\mathbf{x}''_i = \mathbf{V}^T\mathbf{x}'_i \in \mathbb{R}^d$ be the result of rotation, for $i = 1$ to n. The sample covariance matrix of $\mathbf{x}''_1, \cdots, \mathbf{x}''_n$ is

$$\mathbf{C}'' \;=\; \frac{1}{n}\sum_{i=1}^n \mathbf{x}''_i\mathbf{x}''^T_i \;=\; \frac{1}{n}\sum_{i=1}^n \mathbf{V}^T\mathbf{x}'_i\mathbf{x}'^T_i\mathbf{V} \;=\; \mathbf{V}^T\mathbf{X}'^T\mathbf{X}'\mathbf{V} \;=\; \mathbf{\Sigma}\mathbf{U}^T\mathbf{U}\mathbf{\Sigma} \;=\; \mathbf{\Sigma}^2,$$

which is a diagonal matrix with descending diagonal entries. Following the answer to Question 5, one can show that $\mathbf{e}_1 \in \mathbb{R}^d$ is the maximizer of

$$\max_{\mathbf{w}\in\mathbb{R}^d} \mathbf{w}^T\mathbf{C}''\mathbf{w}; \qquad \text{s.t. } \|\mathbf{w}\|_2 = 1.$$

[4] Here, ϵ is the error tolerance, and the spectral gap between the k-th and $(k+1)$-th singular values is considered a constant.

5 Questions

Question 1. *Let* $\{\mathbf{v}_1, \cdots, \mathbf{v}_n\} \subset \mathbb{R}^n$ *be an orthonormal basis of* $\mathbb{R}^n$*. Let* $\mathbf{x} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i \in \mathbb{R}^n$ *and* $\mathbf{A} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$ *(*$r \leq n$*). Express* $\mathbf{A}\mathbf{x} \in \mathbb{R}^m$ *using the singular values and vectors of* $\mathbf{A}$*.*

Question 2. *Express* $\mathbf{A}^T\mathbf{A} \in \mathbb{R}^{n \times n}$ *using the singular values and vectors of* $\mathbf{A}$*.*

Question 3. *Prove that if* $\sigma_1 > \sigma_2$*, then as* q *grows,* $\mathbf{x}_q$ *will converge to* $\mathbf{v}_1$*. To be specific, if* $q = \Omega\big(\frac{\log(n/\epsilon)}{\log(\sigma_1/\sigma_2)}\big)$*, then* $\left|\mathbf{v}_1^T \mathbf{x}_q\right| \geq 1 - \epsilon$ *with high probability.*[5]

Question 4. *Use Python or MATLAB to implement PCA. Hint: you will need the MATLAB function*

[U,S,V] = svds(X, k)

or the Python function

U, S, VT = scipy.sparse.linalg.svds(X, k)

Keep in mind that what Python returns is $\mathbf{V}_k^T \in \mathbb{R}^{k \times d}$*, not* $\mathbf{V} \in \mathbb{R}^{d \times k}$*.*

Question 5. *Proof* $\mathbf{v}_1$ *is the greatest variance direction:* $\mathbf{v}_1 = \text{argmax}_{\mathbf{w} \in \mathbb{R}^d, \|\mathbf{w}\|_2 = 1} \mathbf{w}^T \mathbf{C} \mathbf{w}$*. Hint: there exist some* $\alpha_1, \cdots, \alpha_d \in \mathbb{R}$ *such that* $\mathbf{w} = \alpha_1 \mathbf{v}_1 + \cdots + \alpha_d \mathbf{v}_d$*.*

[5] Since $\mathbf{v}_1$ and $\mathbf{x}_q$ are both unit vectors, $\epsilon \to 0$ implies $\mathbf{x}_q \to \mathbf{v}_1$.

A Answers

Answer to Question 1. Recall that $\mathbf{A} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$ is the SVD of $\mathbf{A}$ and $\mathbf{x} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i$. We can define $\sigma_{r+1} = \cdots = \sigma_n = 0$ and equivalently write $\mathbf{A}$ as $\sum_{i=1}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$. Then

$$\begin{aligned}\mathbf{A}\mathbf{x} &= \left(\sum_{i=1}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T\right)\left(\sum_{j=1}^{n} \alpha_j \mathbf{v}_j\right) = \sum_{i=1}^{n}\sum_{j=1}^{n} \left(\sigma_i \mathbf{u}_i \mathbf{v}_i^T\right)\left(\alpha_j \mathbf{v}_j\right) = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_j \sigma_i \mathbf{u}_i \mathbf{v}_i^T \mathbf{v}_j \\ &= \sum_{l=1}^{n} \alpha_l \sigma_l \mathbf{u}_l \mathbf{v}_l^T \mathbf{v}_l + \sum_{i \neq j} \alpha_j \sigma_i \mathbf{u}_i \mathbf{v}_i^T \mathbf{v}_j = \sum_{l=1}^{n} \alpha_l \sigma_l \mathbf{u}_l + 0 = \sum_{l=1}^{r} \alpha_l \sigma_l \mathbf{u}_l.\end{aligned}$$

Here, the second to last identity is obtained using the two properties of orthonormal basis: $\mathbf{v}_l^T \mathbf{v}_l = 1$ and $\mathbf{v}_i^T \mathbf{v}_j = 0$ for $i \neq j$; the last identity follows from that $\sigma_{r+1} = \cdots = \sigma_n = 0$.

Answer to Question 2. Let $\mathbf{A} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$ be the SVD of $\mathbf{A}$. Then

$$\begin{aligned}\mathbf{A}^T\mathbf{A} &= \left(\sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T\right)^T \left(\sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^T\right) = \left(\sum_{i=1}^{r} \sigma_i \mathbf{v}_i \mathbf{u}_i^T\right)\left(\sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^T\right) \\ &= \sum_{i=1}^{r}\sum_{j=1}^{r} \left(\sigma_i \mathbf{v}_i \mathbf{u}_i^T\right)\left(\sigma_j \mathbf{u}_j \mathbf{v}_j^T\right) = \sum_{i=1}^{r}\sum_{j=1}^{r} \sigma_i \sigma_j \mathbf{v}_i \left(\mathbf{u}_i^T \mathbf{u}_j\right) \mathbf{v}_j^T \\ &= \sum_{l=1}^{r} \sigma_l^2 \mathbf{v}_l \left(\mathbf{u}_l^T \mathbf{u}_l\right) \mathbf{v}_l^T + \sum_{i \neq j} \sigma_i \sigma_j \mathbf{v}_i \left(\mathbf{u}_i^T \mathbf{u}_j\right) \mathbf{v}_j^T = \sum_{l=1}^{r} \sigma_l^2 \mathbf{v}_l \mathbf{v}_l^T + 0.\end{aligned}$$

The last identity is obtained using the two properties of orthonormal basis: $\mathbf{u}_l^T \mathbf{u}_l = 1$ and $\mathbf{u}_i^T \mathbf{u}_j = 0$ for $i \neq j$. Hence, $\mathbf{A}^T\mathbf{A} = \sum_{l=1}^{r} \sigma_l^2 \mathbf{v}_l \mathbf{v}_l^T$.

Answer to Question 3. It is easy to show that

$$\mathbf{x}_q = (\mathbf{A}^T\mathbf{A})^q \mathbf{x}_0 \,/\, \left\|(\mathbf{A}^T\mathbf{A})^q \mathbf{x}_0\right\|_2 \propto (\mathbf{A}^T\mathbf{A})^q \mathbf{x}_0.$$

Here, "proportional to" means that the two vectors have the same directions; they are equal up to a positive or negative scaling. The answer to Question 2 shows that $\mathbf{A}^T\mathbf{A} = \sum_{i=1}^{r} \sigma_i^2 \mathbf{v}_i \mathbf{v}_i^T$. It can be proved using induction that

$$\left(\mathbf{A}^T\mathbf{A}\right)^q = \sum_{i=1}^{r} \sigma_i^{2q} \mathbf{v}_i \mathbf{v}_i^T.$$

It follows that

$$\mathbf{x}_q \propto \left(\mathbf{A}^T\mathbf{A}\right)^q \mathbf{x}_0 = \sum_{i=1}^{r} \sigma_i^{2q} \mathbf{v}_i \mathbf{v}_i^T \mathbf{x}_0.$$

Whatever $\mathbf{x}_0$ is, it can be written as the linear combination $\mathbf{x}_0 = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i$ for some $\alpha_1, \cdots, \alpha_n$. Thus

$$\begin{aligned}\mathbf{x}_q &\propto \sum_{i=1}^{r} \sigma_i^{2q} \mathbf{v}_i \mathbf{v}_i^T \mathbf{x}_0 = \sum_{i=1}^{r} \sigma_i^{2q} \mathbf{v}_i \mathbf{v}_i^T \left(\sum_{j=1}^{n} \alpha_j \mathbf{v}_j\right) = \sum_{i=1}^{r}\sum_{j=1}^{r} \sigma_i^{2q} \alpha_j \mathbf{v}_i \mathbf{v}_i^T \mathbf{v}_j \\ &= \sum_{l=1}^{r} \sigma_l^{2q} \alpha_l \mathbf{v}_l \mathbf{v}_l^T \mathbf{v}_l + \sum_{i \neq j} \sigma_i^{2q} \alpha_j \mathbf{v}_i \mathbf{v}_i^T \mathbf{v}_j = \sum_{l=1}^{r} \sigma_l^{2q} \alpha_l \mathbf{v}_l + 0.\end{aligned}$$

The last identity is obtained using the two properties of orthonormal basis: $\mathbf{u}_l^T\mathbf{u}_l = 1$ and $\mathbf{u}_i^T\mathbf{u}_j = 0$ for $i \neq j$. It follows that

$$\mathbf{x}_q \;\propto\; \sum_{i=1}^{r}\left(\frac{\sigma_i}{\sigma_1}\right)^{2q}\alpha_i\mathbf{v}_i \;=\; \alpha_1\mathbf{v}_1 + \sum_{i=2}^{r}\left(\frac{\sigma_i}{\sigma_1}\right)^{2q}\alpha_i\mathbf{v}_i \;\propto\; \mathbf{v}_1 + \sum_{i=2}^{r}\left(\frac{\sigma_i}{\sigma_1}\right)^{2q}\frac{\alpha_i}{\alpha_l}\mathbf{v}_i \tag{A.1}$$

Since $\mathbf{x}_0$ is randomly drawn from the unit sphere in the $\mathbb{R}^n$ vector space and $\mathbf{v}_i$ is a fixed unit vector, probability theories guarantees that

$$\left|\alpha_i\right| \;=\; \left|\mathbf{x}_0^T\mathbf{v}_i\right| \;=\; \Omega\big(\tfrac{1}{n}\big)$$

with high probability. Obviously, $\alpha_i \leq 1$ for all i. Thus $\frac{\alpha_i}{\alpha_1} = \mathcal{O}(n)$ for all i. The righthand side of (A.1) can be bounded by:

$$\begin{aligned}\gamma \;&\triangleq\; \left\|\mathbf{v}_1 + \sum_{i=2}^{r}\left(\frac{\sigma_i}{\sigma_1}\right)^{2q}\frac{\alpha_i}{\alpha_1}\mathbf{v}_i\right\|_2 \;\leq\; \left\|\mathbf{v}_1\right\|_2 + \left\|\sum_{i=2}^{r}\left(\frac{\sigma_i}{\sigma_1}\right)^{2q}\frac{\alpha_i}{\alpha_1}\mathbf{v}_i\right\|_2 \;\leq\; 1 + \sum_{i=2}^{r}\left(\frac{\sigma_i}{\sigma_1}\right)^{2q}\frac{\alpha_i}{\alpha_1}\\ &\leq\; 1 + (r-1)\left(\frac{\sigma_2}{\sigma_1}\right)^{2q}\frac{\alpha_i}{\alpha_1} \;\leq\; 1 + n\cdot\left(\frac{\sigma_2}{\sigma_1}\right)^{2q}\frac{\alpha_i}{\alpha_1} \;=\; 1 + \mathcal{O}\big(n^2\big)\cdot\left(\frac{\sigma_2}{\sigma_1}\right)^{2q}.\end{aligned}$$

It follows from (A.1) and $\|\mathbf{x}_q\|_2 = 1$ that

$$\mathbf{x}_q \;=\; \pm\frac{1}{\gamma}\left(\mathbf{v}_1 + \sum_{i=2}^{r}\left(\frac{\sigma_i}{\sigma_1}\right)^{2q}\frac{\alpha_i}{\alpha_1}\mathbf{v}_i\right),$$

and thus

$$\left|\mathbf{x}_q^T\mathbf{v}_1\right| \;=\; \frac{1}{\gamma}\left(\mathbf{v}_1^T\mathbf{v}_1 + \sum_{i=2}^{r}\left(\frac{\sigma_i}{\sigma_1}\right)^{2q}\frac{\alpha_i}{\alpha_1}\mathbf{v}_i^T\mathbf{v}_1\right) \;=\; \frac{1}{\gamma}.$$

We conclude that for $q = \Omega\big(\frac{\log(n/\epsilon)}{\log(\sigma_1/\sigma_2)}\big)$,

$$1 - \left|\mathbf{x}_q^T\mathbf{v}_1\right| \;=\; 1 - \frac{1}{\gamma} \;=\; \mathcal{O}\Big(n^2\frac{\sigma_2^{2q}}{\sigma_1^{2q}}\Big) \;\leq\; \epsilon.$$

Answer to Question 5. The sample covariance matrix of $\mathbf{x}_1, \cdots, \mathbf{x}_n \in \mathbb{R}^d$ is

$$\mathbf{C} \;=\; \frac{1}{n}\sum_{i=1}^{n}\big(\mathbf{x}_i - \hat{\boldsymbol{\mu}}\big)\big(\mathbf{x}_i - \hat{\boldsymbol{\mu}}\big)^T \;=\; \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i'\mathbf{x}_i'^T.$$

Let $\mathbf{x}_i' = \mathbf{x}_i - \hat{\boldsymbol{\mu}}$, $\mathbf{X}' = [\mathbf{x}_1', \cdots, \mathbf{x}_n']^T \in \mathbb{R}^{n\times d}$, and $\mathbf{X}' = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = \sum_{i=1}^{d}\sigma_i\mathbf{u}_i\mathbf{v}_i^T$ be the SVD of $\mathbf{X}'$. Then

$$\mathbf{C} \;=\; \mathbf{X}'^T\mathbf{X}' \;=\; \sum_{i=1}^{n}\sigma_i^2\mathbf{v}_i\mathbf{v}_i^T.$$

The above equation follows from the answer to Question 2. There exist some $\alpha_1, \cdots, \alpha_d \in \mathbb{R}$ such that $\mathbf{w} = \alpha_1 \mathbf{v}_1 + \cdots + \alpha_d \mathbf{v}_d$. Then

$$\begin{aligned}\mathbf{C}\mathbf{w} &= \left(\sum_{i=1}^{d} \sigma_i^2 \mathbf{v}_i \mathbf{v}_i^T\right)\left(\sum_{j=1}^{d} \alpha_j \mathbf{v}_j\right) = \sum_{i=1}^{d}\sum_{j=1}^{d} \sigma_i^2 \alpha_j \mathbf{v}_i \mathbf{v}_i^T \mathbf{v}_j \\ &= \sum_{l=1}^{d} \sigma_l^2 \alpha_l \mathbf{v}_l \mathbf{v}_l^T \mathbf{v}_l + \sum_{i \neq j} \sigma_i^2 \alpha_j \mathbf{v}_i \mathbf{v}_i^T \mathbf{v}_i = \sum_{l=1}^{d} \sigma_l^2 \alpha_l \mathbf{v}_l.\end{aligned}$$

We can similarly show that

$$\mathbf{w}^T\mathbf{C}\mathbf{w} = \left(\sum_{i=1}^{d} \alpha_i \mathbf{v}_i^T\right)\left(\sum_{l=1}^{d} \sigma_l^2 \alpha_l \mathbf{v}_l\right) = \sum_{i=1}^{d}\sum_{l=1}^{d} \sigma_l^2 \alpha_l \alpha_i \mathbf{v}_i^T \mathbf{v}_l = \sum_{i=1}^{d} \sigma_i^2 \alpha_i^2.$$

Since $\|\mathbf{w}\|_2^2 = \alpha_1^2 + \cdots + \alpha_d^2$,

$$\underset{\mathbf{w}}{\text{argmax}}\ \mathbf{w}^T\mathbf{C}\mathbf{w}; \qquad \text{s.t. } \|\mathbf{w}\|_2 = 1 \tag{A.2}$$

can be equivalently written as

$$\underset{\alpha_1,\cdots,\alpha_d}{\text{argmax}} \sum_{i=1}^{d} \sigma_i^2 \alpha_i^2; \qquad \text{s.t. } \sum_{j=1}^{d} \alpha_j^2 = 1.$$

Since $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d$, we have that

$$\left[\alpha_1^\star, \alpha_2^\star, \cdots, \alpha_d^\star\right] = \left[1, 0, 0, \cdots, 0\right]$$

is a maximizer. Thus,

$$\mathbf{w}^\star = \alpha_1^\star \mathbf{v}_1 + \cdots \alpha_d^\star \mathbf{v}_d = \mathbf{v}_1$$

is the solution to (A.2).