

20 Questions to Test Your Skills On Dimensionality Reduction (PCA)

 analyticsvidhya.com/blog/2021/05/20-questions-to-test-your-skills-on-dimensionality-reduction-pca

chirag676


May 28, 2021



Certified AI & ML BlackBelt Plus
50+ Projects | 75+ Mentorship Sessions | 14+ Certifications
[Download Brochure](#)



Home » 20 Questions to Test Your Skills On Dimensionality Reduction (PCA)

chirag676, May 28, 2021 Login to Bookmark this article 

This article was published as a part of the Data Science Blogathon

Introduction

Principal Component Analysis is one of the famous Dimensionality Reduction techniques which helps when we work with datasets having very large dimensions.

Therefore it becomes necessary for every aspiring **Data Scientist** and **Machine Learning Engineer** to have a good knowledge of Dimensionality Reduction.

In this article, we will discuss the most important questions on Dimensionality Reduction which is helpful to get you a clear understanding of the techniques, and also for **Data Science Interviews**, which cover its very fundamental level to complex concepts.

Let's get started,

1. What is Dimensionality Reduction?

In Machine Learning, dimension refers to the number of features in a particular dataset.

In simple words, Dimensionality Reduction refers to reducing dimensions or features so that we can get a more interpretable model, and improves the performance of the model.

2. Explain the significance of Dimensionality Reduction.

There are basically three reasons for Dimensionality reduction:

- **Visualization**
- **Interpretability**
- **Time and Space Complexity**

Let's understand this with an example:

Imagine we have worked on an **MNIST** dataset that contains **28 × 28** images and when we convert images to features we get 784 features.

If we try to think of each feature as one dimension, then how can we think of 784 dimensions in our mind?

We are not able to visualize the scattering of points of 784 dimensions.

That is the first reason why Dimensionality Reduction is Important!

Let's say you are a data scientist and you have to explain your model to clients who do not understand Machine Learning, how will you make them understand the working of 784 features or dimensions.

In simple language, how we interpret the model to the clients.

That is the second reason why Dimensionality Reduction is Important!

Let's say you are working for an internet-based company where the output of something must be in milliseconds or less than that, so "Time complexity" and "Space Complexity" matter a lot. More features need more Time which these types of companies can't afford.

That is the third reason why Dimensionality Reduction is Important!

3. What is PCA? What does a PCA do?

PCA stands for **Principal Component analysis**. It is a dimensionality reduction technique that summarizes a large set of correlated variables (basically high dimensional data) into a smaller number of representative variables, called the Principal Components, that explains most of the variability of the original set i.e, not losing that much of the information.

PCA is a **deterministic algorithm** in which we have not any parameters to initialize and it doesn't have a problem of local minima, like most of the machine learning algorithms has.

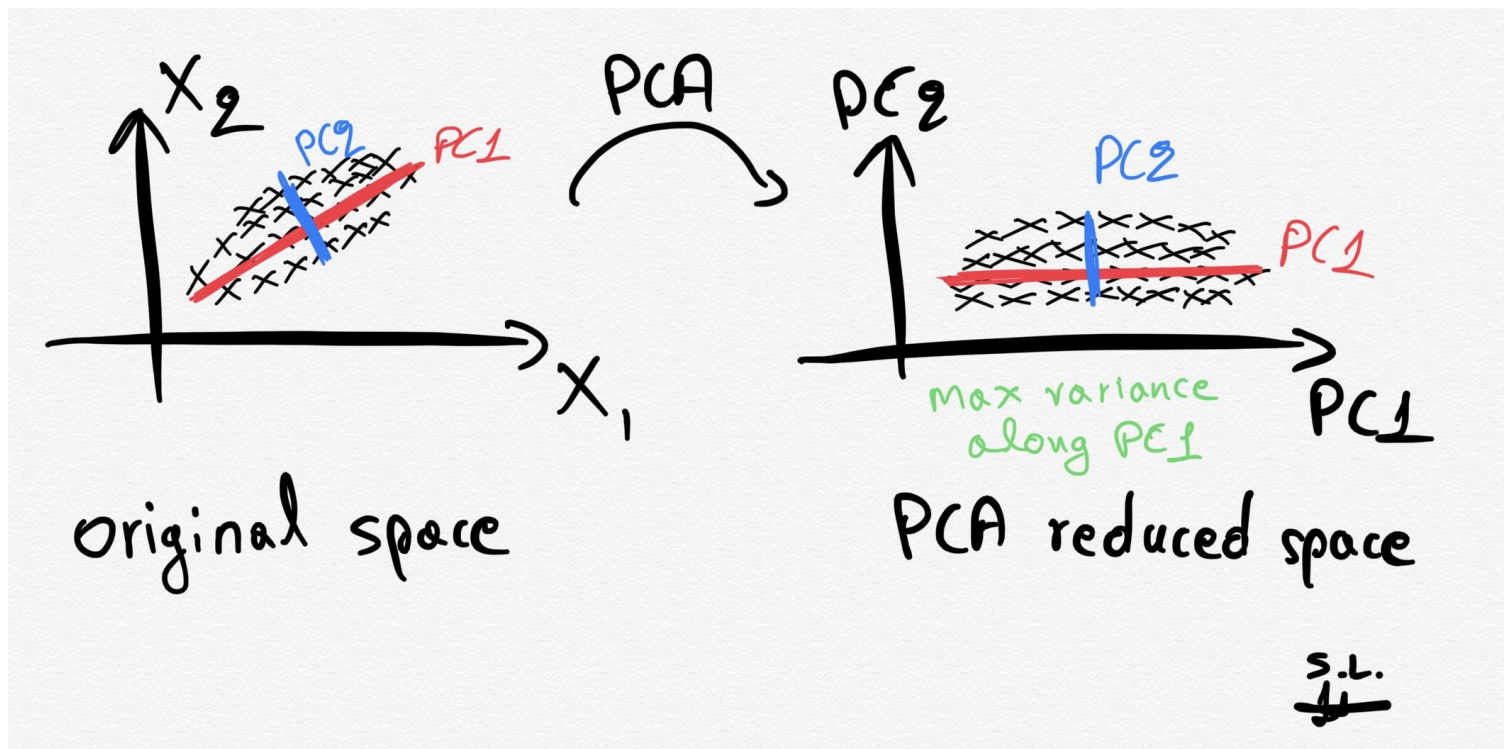


Image Source: Google Images

4. List down the steps of a PCA algorithm.

The major steps which are to be followed while using the PCA algorithm are as follows:

Step-1: Get the dataset.

Step-2: Compute the mean vector (μ).

Step-3: Subtract the means from the given data.

Step-4: Compute the covariance matrix.

Step-5: Determine the eigenvectors and eigenvalues of the covariance matrix.

Step-6: Choosing Principal Components and forming a feature vector.

Step-7: Deriving the new data set by taking the projection on the weight vector.

5. Is it important to standardize the data before applying PCA?

Usually, the aim of standardization is to assign equal weights to all the variables. PCA finds new axes based on the covariance matrix of original variables. As the covariance matrix is sensitive to the standardization of variables therefore if we use features of different scales, we often get misleading directions.

Moreover, if all the variables are on the same scale, then there is no need to standardize the variables.

6. Is rotation necessary in PCA? If yes, Why? Discuss the consequences if we do not rotate the components?

Yes, the idea behind rotation i.e, orthogonal Components is so that we are able to capture the maximum variance of the training set.

If we don't rotate the components, the effect of PCA will diminish and we'll have to select more Principal Components to explain the maximum variance of the training dataset.

7. What are the assumptions taken into consideration while applying PCA?

The assumptions needed for PCA are as follows:

1. PCA is based on **Pearson correlation coefficients**. As a result, there needs to be a linear relationship between the variables for applying the PCA algorithm.
2. For getting reliable results by using the PCA algorithm, we require a large enough sample size i.e, we should have sampling adequacy.
3. Your data should be suitable for data reduction i.e., we need to have adequate correlations between the variables to be reduced to a smaller number of components.
4. No significant **noisy data or outliers** are present in the dataset.

8. What will happen when eigenvalues are roughly equal while applying PCA?

While applying the PCA algorithm, If we get all eigenvectors the same, then the algorithm won't be able to select the Principal Components because in such cases, all the Principal Components are equal.

9. What are the properties of Principal Components in PCA?

The properties of principal components in PCA are as follows:

1. These Principal Components are linear combinations of original variables that result in an axis or a set of axes that explain/s most of the variability in the dataset.
2. All Principal Components are orthogonal to each other.
3. The first Principal Component accounts for most of the possible variability of the original data i.e, maximum possible variance.
4. The number of Principal Components for n-dimensional data should be at utmost equal to n (=dimension). **For Example**, There can be only two Principal Components for a two-dimensional data set.

10. What does a Principal Component in a PCA signify? How can we represent them mathematically?

The Principal Component represents a line or an axis along which the data varies the most and it also is the line that is closest to all of the n observations in the dataset.

In mathematical terms, we can say that the first Principal Component is the eigenvector of the covariance matrix corresponding to the maximum eigenvalue.

Accordingly,

- **Sum of squared distances = Eigenvalue for PC-1**
- **Sqrt of Eigenvalue = Singular value for PC-1**

11. What does the coefficient of Principal Component signify?

If we project all the points on the Principal Component, they tell us that the independent variable 2 is N times as important as of independent variable 1.

12. Can PCA be used for regression-based problem statements? If Yes, then explain the scenario where we can use it.

Yes, we can use Principal Components for regression problem statements.

PCA would perform well in cases when the first few Principal Components are sufficient to capture most of the variation in the independent variables as well as the relationship with the dependent variable.

The only problem with this approach is that the new reduced set of features would be modeled by ignoring the dependent variable Y when applying a PCA and while these features may do a good overall job of explaining the variation in X , the model will perform poorly if these variables don't explain the variation in Y .

13. Can we use PCA for feature selection?

Feature selection refers to choosing a subset of the features from the complete set of features.

No, PCA is not used as a feature selection technique because we know that any Principal Component axis is a linear combination of all the original set of feature variables which defines a new set of axes that explain most of the variations in the data.

Therefore while it performs well in many practical settings, it does not result in the development of a model that relies upon a small set of the original features.

14. Comment whether PCA can be used to reduce the dimensionality of the non-linear dataset.

PCA does not take the nature of the data i.e, linear or non-linear into considerations during its algorithm run but PCA focuses on reducing the dimensionality of most datasets significantly. PCA can at least get rid of useless dimensions.

However, reducing dimensionality with PCA will lose too much information if there are no useless dimensions.

15. How can you evaluate the performance of a dimensionality reduction algorithm on your dataset?

A dimensionality reduction algorithm is said to work well if it eliminates a significant number of dimensions from the dataset without losing too much information. Moreover, the use of dimensionality reduction in preprocessing before training the model allows measuring the performance of the second algorithm.

We can therefore infer if an algorithm performed well if the dimensionality reduction does not lose too much information after applying the algorithm.

Comprehension Type Question: (16 – 18)

Consider a set of 2D points $\{(-3,-3), (-1,-1), (1,1), (3,3)\}$. We want to reduce the dimensionality of these points by 1 using PCA algorithms. Assume $\sqrt{2}=1.414$.

Now, Answer the Following Questions:

16. The eigenvalue of the data matrix XX^T is equal to _____?

17. The weight matrix W will be equal to _____?

18. The reduced dimensionality data will be equal to _____?

SOLUTION:

Here the original data resides in R^2 i.e, two-dimensional space, and our objective is to reduce the dimensionality of the data to 1 i.e, 1-dimensional data $\Rightarrow K=1$

We try to solve these set of problem step by step so that you have a clear understanding of the steps involved in the PCA algorithm:

Step-1: Get the Dataset

Here data matrix X is given by $\begin{bmatrix} -3 & -1 & 1 & 3 \\ -3 & -1 & 1 & 3 \end{bmatrix}$

Step-2: Compute the mean vector (μ)

Mean Vector: $\begin{bmatrix} \{-3+(-1)+1+3\}/4, \{-3+(-1)+1+3\}/4 \end{bmatrix} = \begin{bmatrix} 0, 0 \end{bmatrix}$

Step-3: Subtract the means from the given data

Since here the mean vector is 0, 0 so while subtracting all the points from the mean we get the same data points.

Step-4: Compute the covariance matrix

Therefore, the covariance matrix becomes XX^T since the mean is at the origin.

Therefore, XX^T becomes $\begin{bmatrix} -3 & -1 & 1 & 3 \\ -3 & -1 & 1 & 3 \end{bmatrix} \begin{bmatrix} -3 & -1 & 1 & 3 \\ -3 & -1 & 1 & 3 \end{bmatrix}^T$
 $= \begin{bmatrix} 20 & 20 \\ 20 & 20 \end{bmatrix}$

Step-5: Determine the eigenvectors and eigenvalues of the covariance matrix

$\det(C-\lambda I)=0$ gives the eigenvalues as 0 and 40.

Now, choose the maximum eigenvalue from the calculated and find the eigenvector corresponding to $\lambda = 40$ by using the equations $CX = \lambda X$:

Accordingly, we get the eigenvector as $(1/\sqrt{2}) [1, 1]$

Therefore, the eigenvalues of matrix XX^T are 0 and 40.

Step-6: Choosing Principal Components and forming a weight vector

Here, $U = R^{2 \times 1}$ and equal to the eigenvector of XX^T corresponding to the largest eigenvalue.

Now, the eigenvalue decomposition of $C=XX^T$

And W (weight matrix) is the transpose of the U matrix and given as a row vector.

Therefore, the weight matrix is given by $[1 \ 1]/1.414$

Step-7: Deriving the new data set by taking the projection on the weight vector

Now, reduced dimensionality data is obtained as $x_i = U^T X_i = WX_i$

$$x_1 = WX_1 = (1/\sqrt{2}) [1, 1] [-3, -3]^T = -3\sqrt{2}$$

$$x_2 = WX_2 = (1/\sqrt{2}) [1, 1] [-1, -1]^T = -\sqrt{2}$$

$$x_3 = WX_3 = (1/\sqrt{2}) [1, 1] [1, 1]^T = \sqrt{2}$$

$$x_4 = WX_4 = (1/\sqrt{2}) [1, 1] [3, 3]^T = 3\sqrt{2}$$

Therefore, the reduced dimensionality will be equal to $\{-3*1.414, -1.414, 1.414, 3*1.414\}$.

This completes our example!

19. What are the Advantages of Dimensionality Reduction?

Some of the advantages of Dimensionality reduction are as follows:

1. Less misleading data means model accuracy improves.

2. Fewer dimensions mean less computing. Less data means that algorithms train faster.
3. Less data means **less storage** space required.
4. Removes redundant features and **noise**.
5. Dimensionality Reduction helps us to visualize the data that is present in higher dimensions in 2D or 3D.

20. What are the Disadvantages of Dimensionality Reduction?

Some of the disadvantages of Dimensionality reduction are as follows:

1. While doing dimensionality reduction, we lost some of the information, which can possibly affect the performance of subsequent training algorithms.
2. It can be computationally intensive.
3. Transformed features are often hard to interpret.
4. It makes the independent variables less interpretable.

End Notes

Thanks for reading!

I hope you enjoyed the questions and were able to test your knowledge about Dimensionality Reduction.

If you liked this and want to know more, go visit my other articles on Data Science and Machine Learning by clicking on the **Link**

Please feel free to contact me on **Linkedin, Email**.

Something not mentioned or want to share your thoughts? Feel free to comment below And I'll get back to you.

About the author

Chirag Goyal

Currently, I am pursuing my Bachelor of Technology (B.Tech) in Computer Science and Engineering from the **Indian Institute of Technology Jodhpur(IITJ)**. I am very enthusiastic about Machine learning, Deep Learning, and Artificial Intelligence.

The media shown in this article are not owned by Analytics Vidhya and is used at the Author's discretion.

You can also read this article on our Mobile APP



Leave A Reply

Your email address will not be published.