1.What are the key reasons for reducing the dimensionality of a dataset? What are the major disadvantages?

Ans.

Reducing the dimensionality of a dataset can simplify the data and make it easier to analyze, visualize, and understand. It can also help to reduce computational complexity and save storage space. However, reducing dimensionality can also result in loss of information and may make it harder to interpret the data.

2.What is the dimensionality curse?

Ans.

The dimensionality curse refers to the problem of increased computational complexity and reduced performance in machine learning models as the number of features or dimensions in the dataset increases. This can lead to overfitting, poor generalization, and increased error rates.

3.Tell if it's possible to reverse the process of reducing the dimensionality of a dataset? If so, how can you go about doing it? If not, what is the reason?

Ans.

In most cases, it is not possible to completely reverse the process of reducing the dimensionality of a dataset as some information is lost in the process. However, it may be possible to use techniques such as reconstruction or inverse transformations to approximate the original data from the reduced-dimensional representation.

4.Can PCA be utilized to reduce the dimensionality of a nonlinear dataset with a lot of variables?

Ans.

PCA is designed for linear data transformations and may not work well on nonlinear datasets. In such cases, nonlinear dimensionality reduction techniques such as kernel PCA may be more appropriate.

5.Assume you're running PCA on a 1,000-dimensional dataset with a 95 percent explained variance ratio. What is the number of dimensions that the resulting dataset would have?

Ans.

The number of dimensions in the resulting dataset would depend on the amount of variance that needs to be explained. If the 95 percent explained variance ratio is achieved using fewer dimensions, then the resulting dataset will have fewer dimensions.

6.Will you use vanilla PCA, incremental PCA, randomized PCA, or kernel PCA in which situations?

Ans.

Vanilla PCA can be used when the dataset is not too large and can fit in memory. Incremental PCA can be used when the dataset is too large to fit in memory and needs to be processed in batches. Randomized PCA can be used when faster computations are required, and kernel PCA can be used when dealing with nonlinear datasets.

7.How do you assess a dimensionality reduction algorithm's success on your dataset?

Ans.

The success of a dimensionality reduction algorithm can be assessed by measuring the amount of variance retained, comparing the performance of the algorithm on the original and reduced datasets, and assessing the impact of dimensionality reduction on downstream tasks.

8.Is it logical to use two different dimensionality reduction algorithms in a chain?

Ans.

Yes, it can be logical to use two different dimensionality reduction algorithms in a chain if the first algorithm cannot handle the dataset's nonlinearities, and the second algorithm is better suited for the task. However, care should be taken to avoid overfitting and loss of information in the process.