PCA

Unsupervised.

The most common way of reducing the dimension of multivariate data.

High dimensionality is one of the challenging problems machine learning engineers face when dealing with a dataset with a huge number of features and samples. PCA is an unsupervised method that focuses on capturing the most variance in the data, which can be useful when labels are not available or when seeking to remove noise and redundancy.

PCA is a LINEAR dimensionality reduction technique which **converts a set of correlated features in the VERY HIGH dimensional space into** a low-dimensional space while capturing most of information.
PCA works by **simply reducing the number of features (columns) while retaining maximum information**.

Application - Noise filtering, feature extractions, stock market predictions, and gene data analysis.

Dimensionality reduction refers **to techniques for reducing the number of input variables in TRAINING data**.

If preserving the original meaning of features is crucial, feature selection should be your go-to option since it retains original variables. On the contrary, feature extraction transforms the original (correlated) variables into a new set of features, which might be difficult to interpret.

The curse of dimensionality states that as the number of dimensions or features in a dataset increases, the volume of the data space expands exponentially.

Dimensionality reduction reduces the complexity of data, which reduces irrelevant data and improves performance. Increase in visualization. High dimensional data is more difficult to visualize when compared to lower/simplified dimensional data. Prevents overfitting.

Variance - how much variation in a dataset can be attributed to each of the principal component.

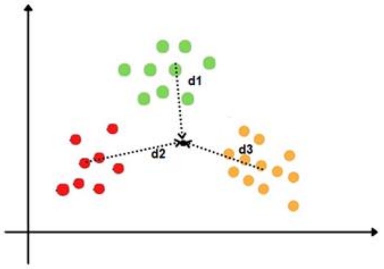
We find the directions in which we can **capture maximum variance** using Eigenvalues and Eigenvectors.

Disadvantages of PCA:

- Independent variables become less interpretable
- Data standardization is a must before PCA
- Information Loss

There are many methods for Dimensionality Reduction like PCA, **ICA, t-SNE**. Sample of clustering Iris dataset.

Both LDA and **PCA are linear transformation techniques but LDA is supervised and has class labels. Can be used for vizualizations to simply create charts for high dimensional dataset.**

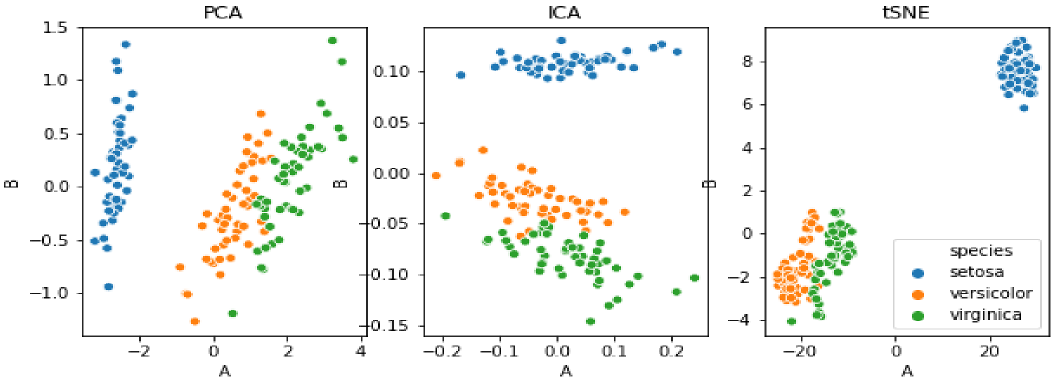


LDA focuses on finding a feature subspace that **maximizes the separability** between the groups.

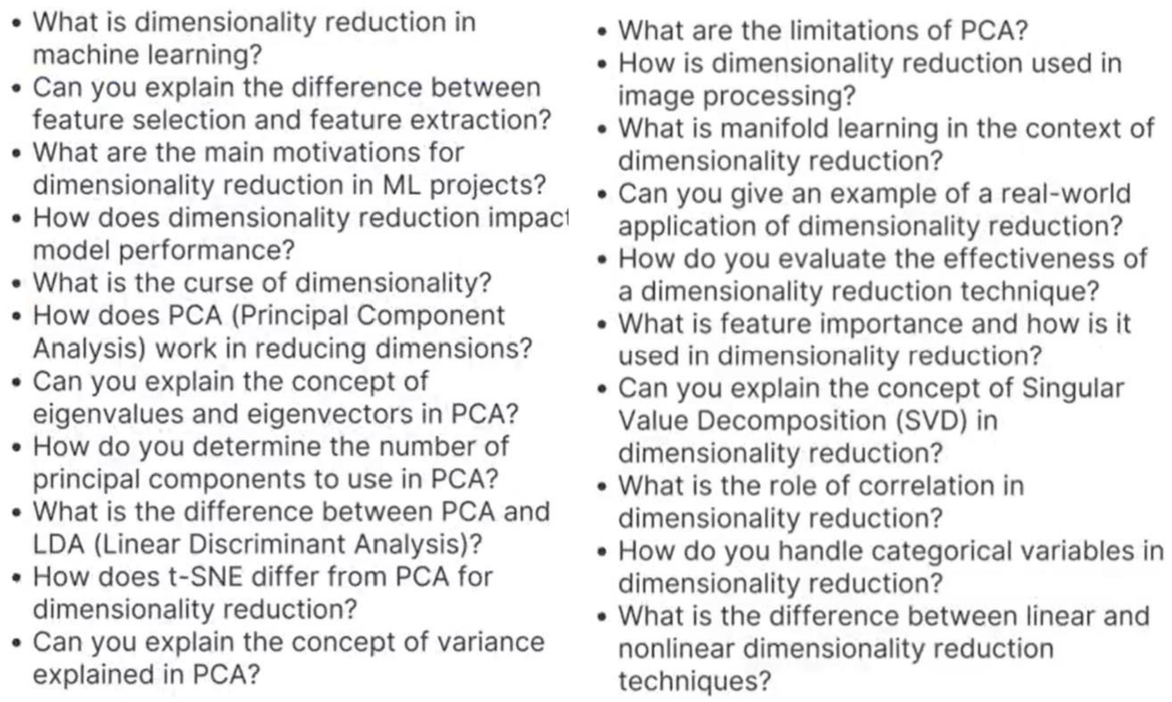
LDA calculates the central point of all the categories and the distance between the central points of each category to that point.

It then projects the data onto the new axes in such a way that there is a maximum separation between the groups and minimum variation within the groups.

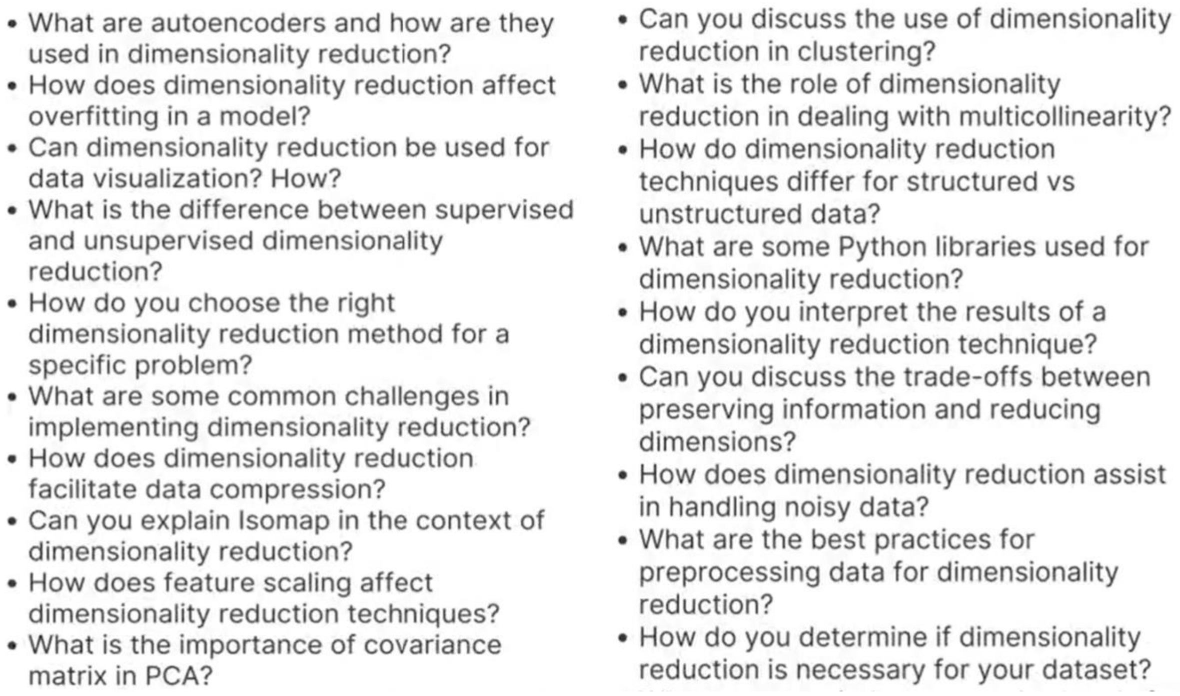
LDA and PCA both form a new set of components. PCA (max variations in data, LDA – max variation in groups).



.NO.	**PCA**	**t-SNE**
1.	It is a linear Dimensionality reduction technique.	It is a non-linear Dimensionality reduction technique.
2.	It tries to preserve the global structure of the data.	It tries to preserve the local structure(cluster) of data.
3.	It does not work well as compared to t-SNE.	It is one of the best dimensionality reduction technique.
4.	It does not involve Hyperparameters.	It involves Hyperparameters such as perplexity, learning rate and number of steps.
5.	It gets highly affected by outliers.	It can handle outliers.
6.	PCA is a deterministic algorithm.	It is a non-deterministic or randomised algorithm.
7.	It works by rotating the vectors for preserving variance.	It works by minimising the distance between the point in a gaussian.
8.	We can find decide on how much variance to preserve using eigen values.	We cannot preserve variance instead we can preserve distance using hyperparameters.
9.	PCA is computationally less expensive than t-SNE, especially for large datasets.	t-SNE can be computationally expensive, especially for high-dimensional datasets with a large number of data points.
10.	It can be used for visualization of high-dimensional data in a low-dimensional space.	It is specifically designed for visualization and is known to perform better in this regard.
11.	It is suitable for linearly separable datasets.	It is more suitable for non-linearly separable datasets.
12.	It can be used for feature extraction	It is mainly used for visualization and exploratory data analysis.
13.	PCA can be sensitive to the ordering of the data points	t-SNE is less sensitive to the ordering of the data points.



- What is dimensionality reduction in machine learning?
- Can you explain the difference between feature selection and feature extraction?
- What are the main motivations for dimensionality reduction in ML projects?
- How does dimensionality reduction impact model performance?
- What is the curse of dimensionality?
- How does PCA (Principal Component Analysis) work in reducing dimensions?
- Can you explain the concept of eigenvalues and eigenvectors in PCA?
- How do you determine the number of principal components to use in PCA?
- What is the difference between PCA and LDA (Linear Discriminant Analysis)?
- How does t-SNE differ from PCA for dimensionality reduction?
- Can you explain the concept of variance explained in PCA?
- What are the limitations of PCA?
- How is dimensionality reduction used in image processing?
- What is manifold learning in the context of dimensionality reduction?
- Can you give an example of a real-world application of dimensionality reduction?
- How do you evaluate the effectiveness of a dimensionality reduction technique?
- What is feature importance and how is it used in dimensionality reduction?
- Can you explain the concept of Singular Value Decomposition (SVD) in dimensionality reduction?
- What is the role of correlation in dimensionality reduction?
- How do you handle categorical variables in dimensionality reduction?
- What is the difference between linear and nonlinear dimensionality reduction techniques?



- What are autoencoders and how are they used in dimensionality reduction?
- How does dimensionality reduction affect overfitting in a model?
- Can dimensionality reduction be used for data visualization? How?
- What is the difference between supervised and unsupervised dimensionality reduction?
- How do you choose the right dimensionality reduction method for a specific problem?
- What are some common challenges in implementing dimensionality reduction?
- How does dimensionality reduction facilitate data compression?
- Can you explain Isomap in the context of dimensionality reduction?
- How does feature scaling affect dimensionality reduction techniques?
- What is the importance of covariance matrix in PCA?
- Can you discuss the use of dimensionality reduction in clustering?
- What is the role of dimensionality reduction in dealing with multicollinearity?
- How do dimensionality reduction techniques differ for structured vs unstructured data?
- What are some Python libraries used for dimensionality reduction?
- How do you interpret the results of a dimensionality reduction technique?
- Can you discuss the trade-offs between preserving information and reducing dimensions?
- How does dimensionality reduction assist in handling noisy data?
- What are the best practices for preprocessing data for dimensionality reduction?
- How do you determine if dimensionality reduction is necessary for your dataset?

- How do you validate the results of dimensionality reduction?
- Can dimensionality reduction be automated?
- What is the impact of missing data on dimensionality reduction techniques?
- How does dimensionality reduction affect computational efficiency?
- Can you compare and contrast different dimensionality reduction algorithms?
- What is the concept of distance metrics in dimensionality reduction?
- How can dimensionality reduction be used in feature engineering?
- Can you explain the concept of local vs global dimensionality reduction?
- How does the choice of distance metric affect the outcome of t-SNE?
- What are some recent advancements in dimensionality reduction techniques?
- How do dimensionality reduction techniques differ in supervised learning vs

- How does dimensionality reduction interact with other preprocessing steps?
- What are the common misconceptions about dimensionality reduction?
- Can you predict future trends in dimensionality reduction techniques in machine learning?
- Can dimensionality reduction be applied to time-series data?
- What are the ethical considerations in applying dimensionality reduction?
- How do you assess the stability of dimensionality reduction methods?
- What is the role of dimensionality reduction in Big Data?
- How can dimensionality reduction influence the interpretability of a model?
- What are some domain-specific challenges in dimensionality reduction?
- Can you discuss the integration of dimensionality reduction in deep learning models?
- How do you handle outliers in dimensionality reduction?
- What is the significance of batch effects in dimensionality reduction of biological data?