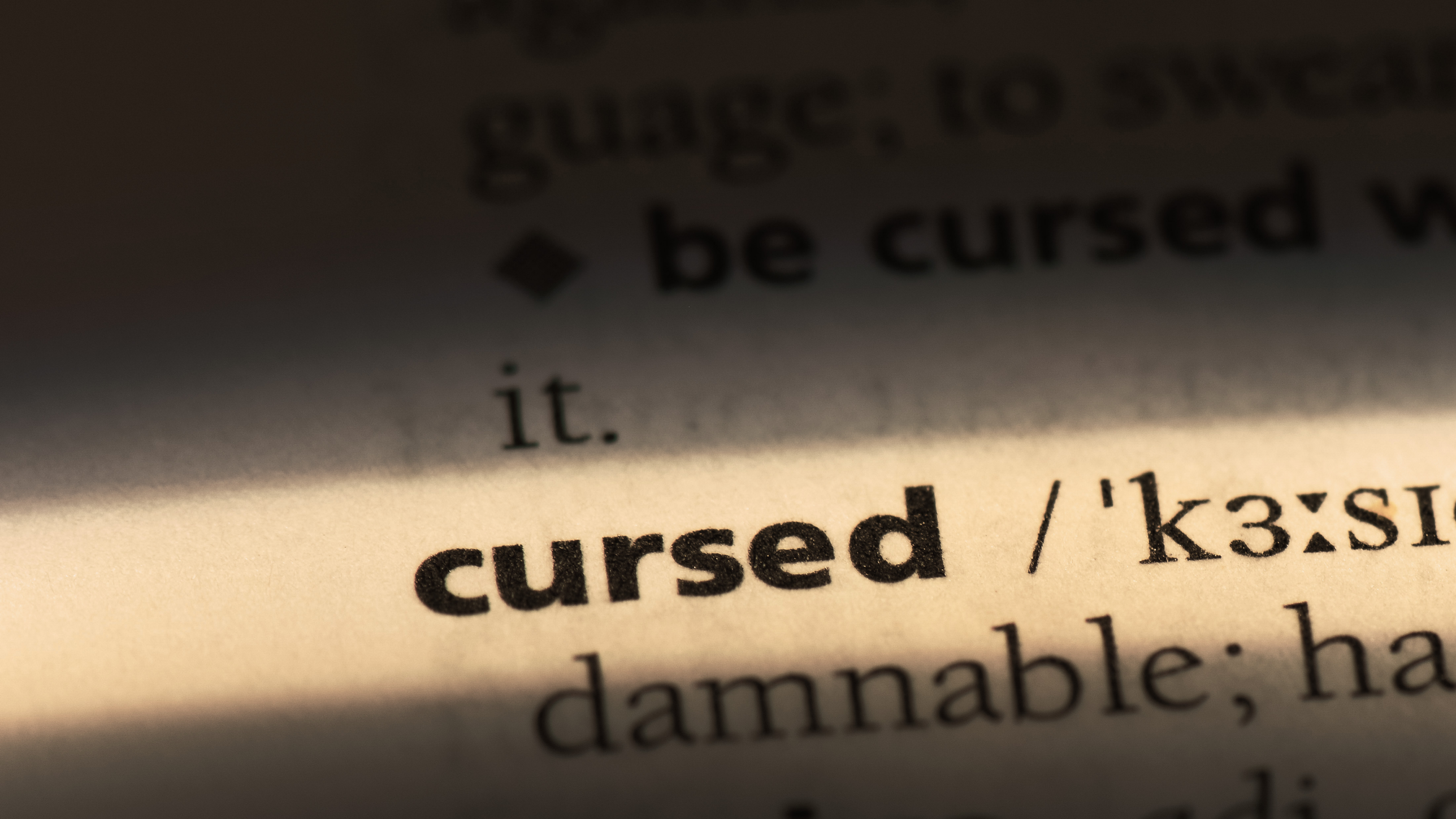
Dimensionality Reduction

Xavier Morera

Helping developers understand and work with data

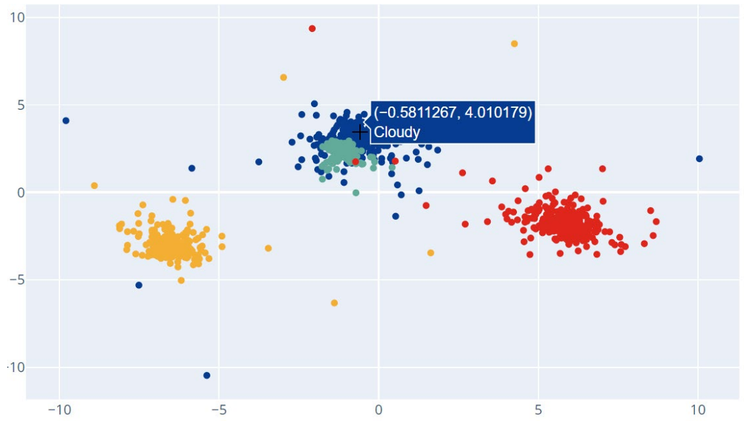
@xmorera www.xaviermorera.com / www.bigdatainc.org

Curse of Dimensionality


be cursed
it.
cursed
damnable; ha

Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA)



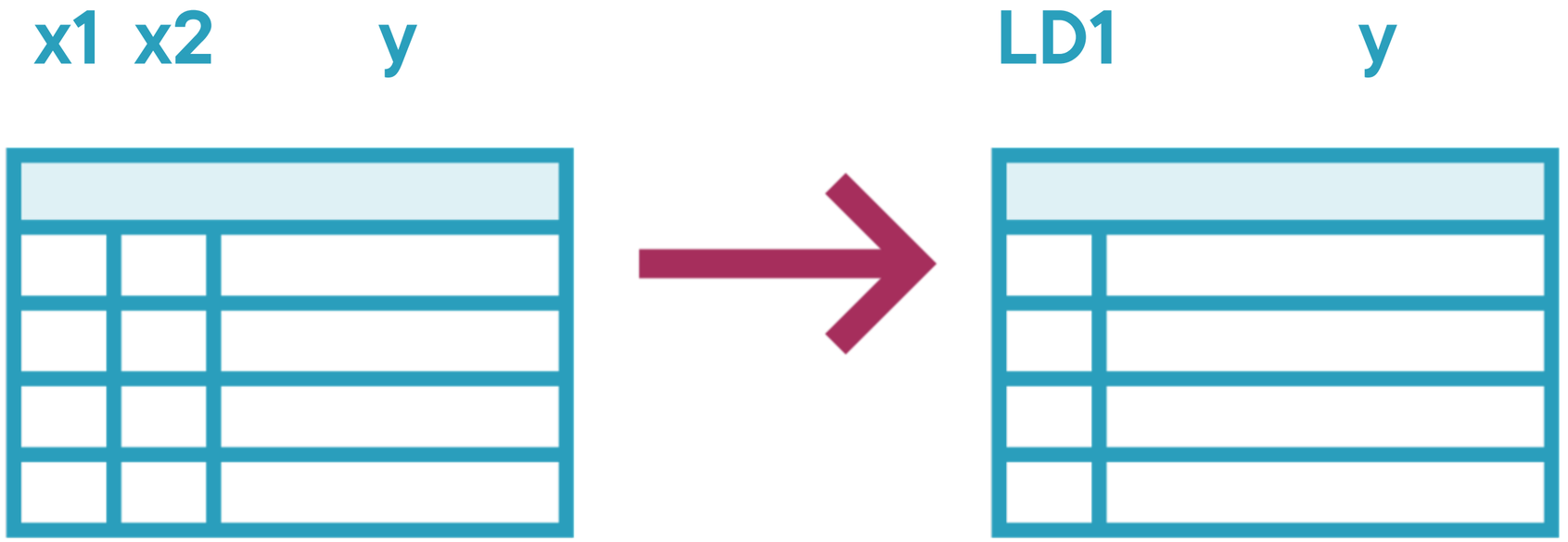
Supervised dimensional reduction algorithm

Serves two purposes

- Can work as a linear classifier
- Reduces dimensions of a dataset by searching for linear relationships between variables

Assumes that each class is normally-distributed

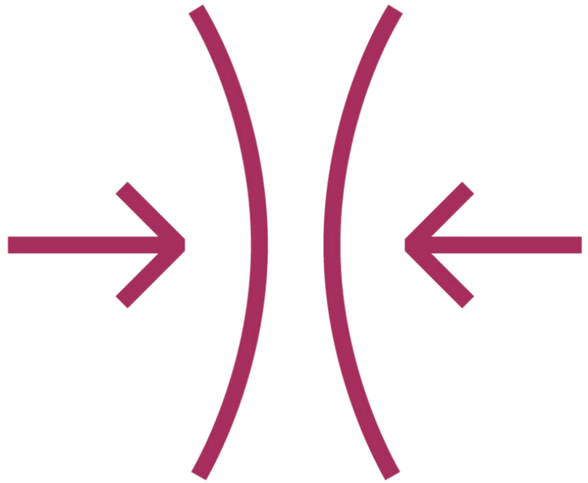
Linear Discriminant Analysis (LDA)



Once the data is projected, the newly created feature has no particular meaning

Makes explainability harder

Using Dimensionality Reduction

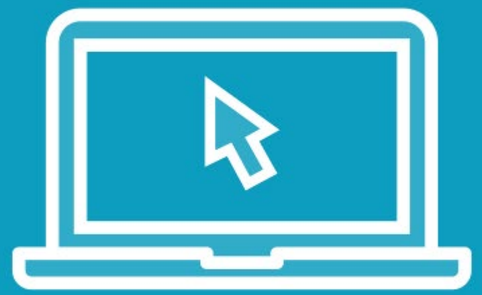


Confirm data is outlier-free

Data must be normally distributed

Use data transformation techniques

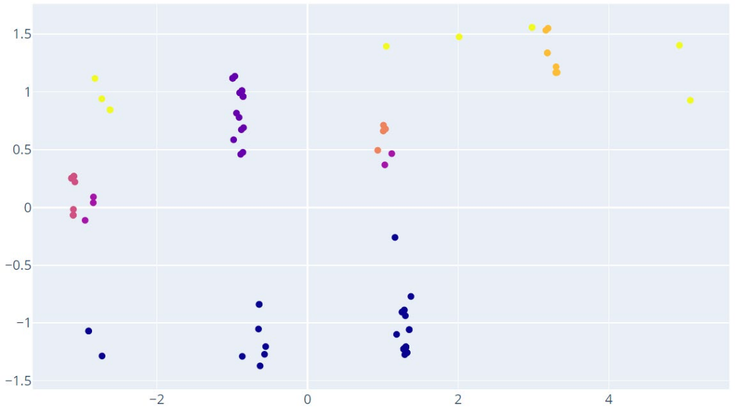
Demo



Linear Discriminant Analysis (LDA)

Principal Component Analysis (PCA)

Principal Component Analysis (PCA)



Similar to LDA in many respects

- LDA uses supervised learning while PCA can use both supervised and unsupervised learning

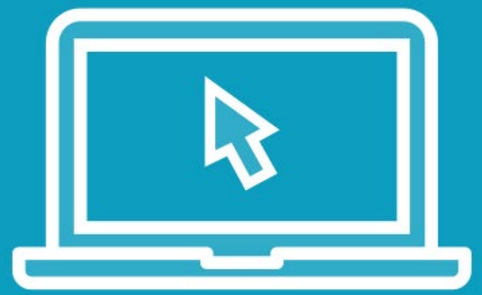
Similar but not the same

- Test both algorithms and compare results

PCA performs a transformation that merges n-number of dimensions into a number of k-components

Once you reduce dimensionality, the new component deals with collinearity and keeps the variance from the old features, making the new components very relevant for predicting tasks

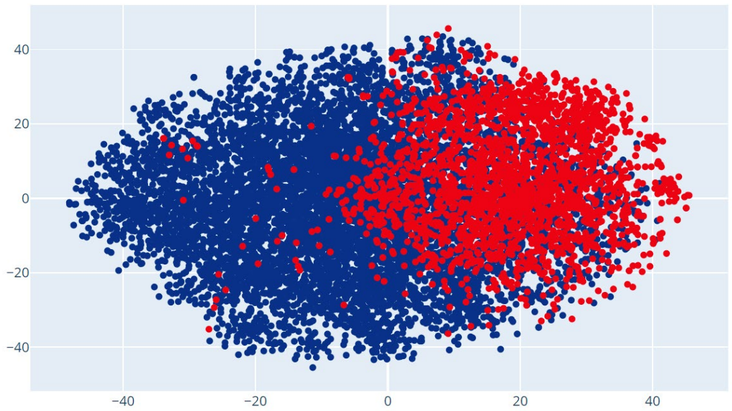
Demo



Principal Component Analysis (PCA)

t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-Distributed Stochastic Neighbor Embedding (t-SNE)



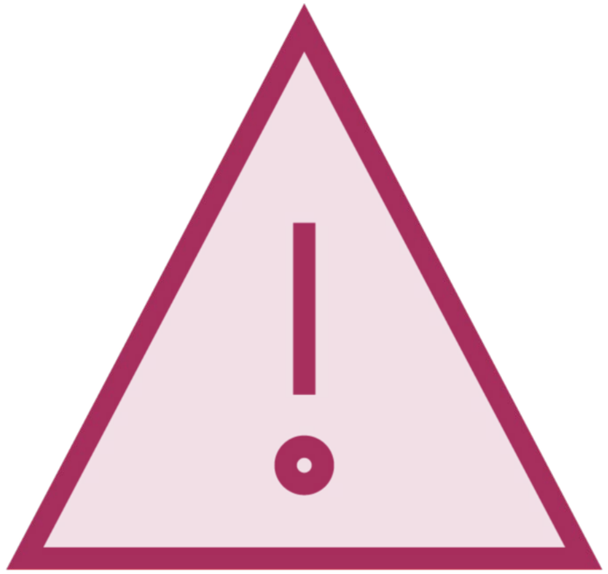
Non-linear dimensionality reduction technique

Can reduce dimensions in speech, image, NLP, and genomic data because it is non-linear

Fully probabilistic and calculates similarity of points in different dimensional spaces

Recommended for small and mid-sized datasets as it is computationally expensive

Take this into account

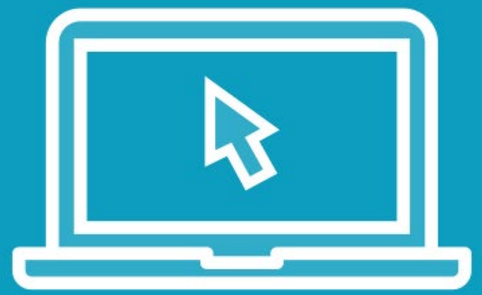


Might produce different results over multiple runs due to its probabilistic nature

t-SNE can handle polynomial data, but it might be a slow process

Hyperparameters can change drastically

Demo



t-Distributed Stochastic Neighbor Embedding (t-SNE)

Takeaway

Curse of dimensionality

- How to deal with too many features

Reducing dimensions can help algorithms converge faster

- And create visualizations

Other dimensionality reduction techniques available

- TruncateSVD, Isomap Embedding, Linear Embedding, Multidimensional Scaling, and Factor Analysis