Unsupervised Learning

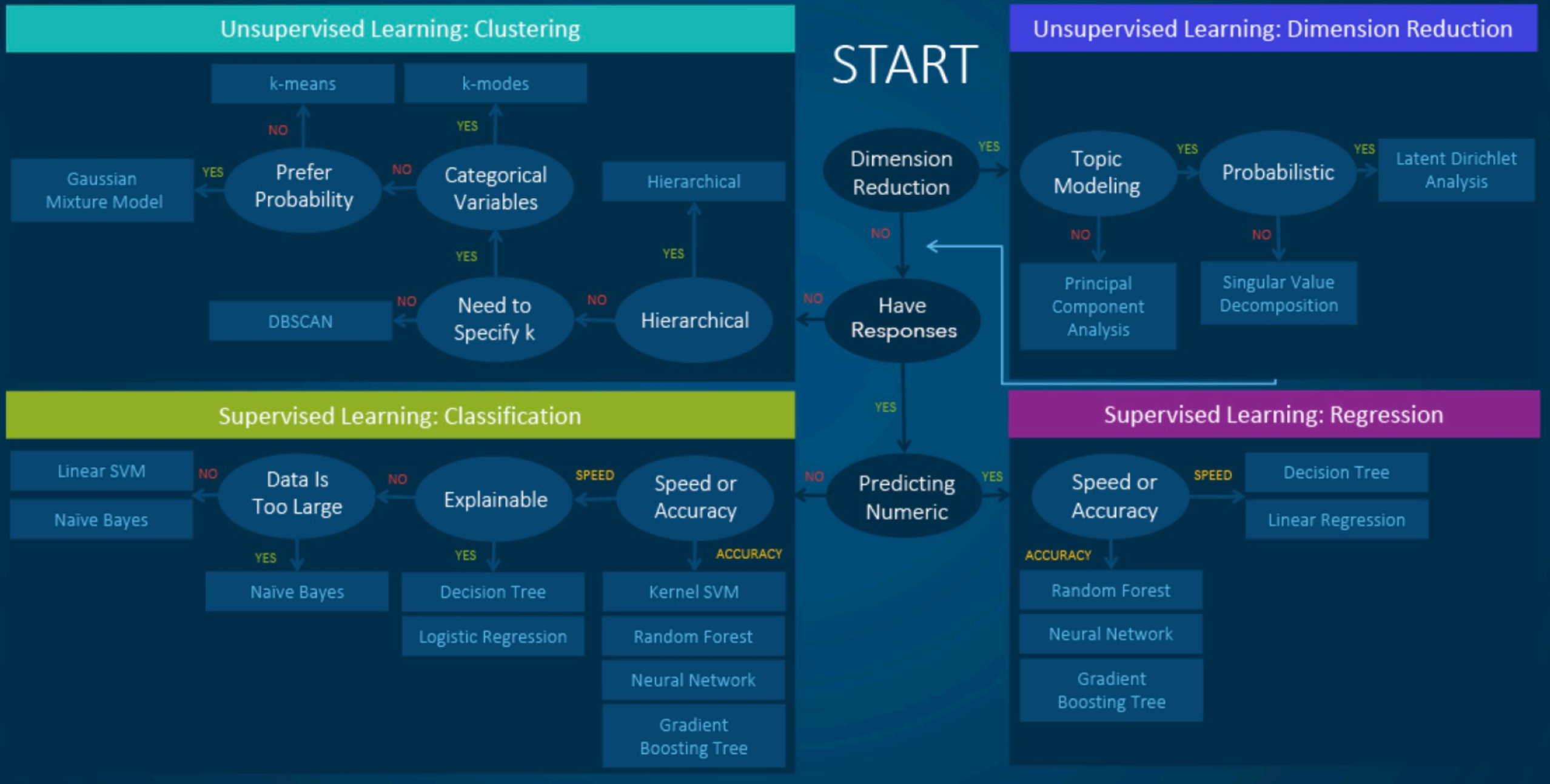
Session-19

SUMMARY of PREVIOUS CLASS

- **Hierarchical Clustering**
  - **Dendrogram**
  - **Agglomerative, Divisive**
- ***Minimal intra cluster distance*** and ***maximal inter cluster distance***.
- **Linkage:** Ward, Complete, Single, Average
- **affinity**: Euclidean, Manhattan, Cosine

- Principal Component Analysis (PCA) Theory
- PCA with Python

Machine Learning Algorithms Cheat Sheet



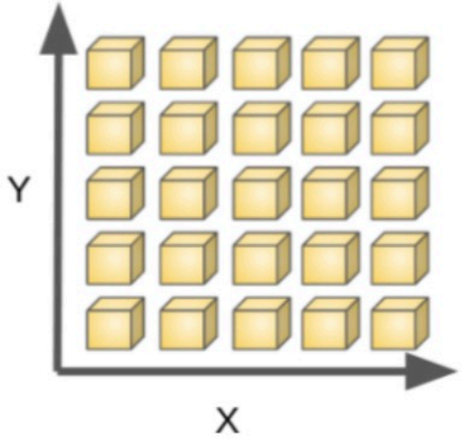
Principal Component Analysis

The Curse of Dimensionality

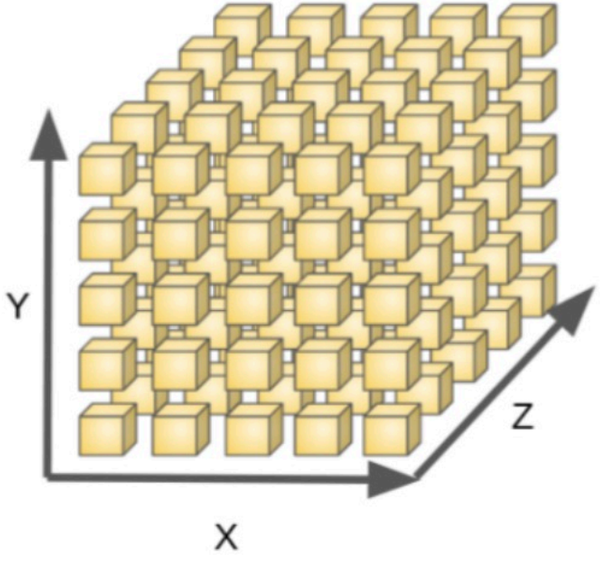
**** Every variable in our data set is a dimension.***

Let's consider running a model that will find an optimum value for each dimension. To simplify even more, suppose we only have 5 different values that we can search for each dimension. So our problem is finding the best of the 5 values for each of the input dimensions. Total number of combinations we need to look for **if we have 20 attributes:**

1D: 5 data points



2D: 25 data points



3D: 125 data points

. . . . **20 features**

20D: 95.367.431.640.625 data points

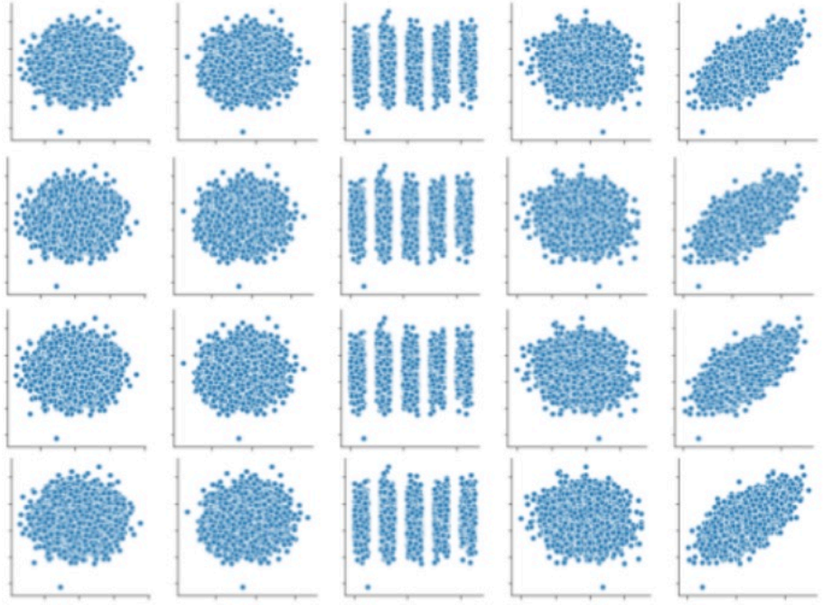
Source: medium.com

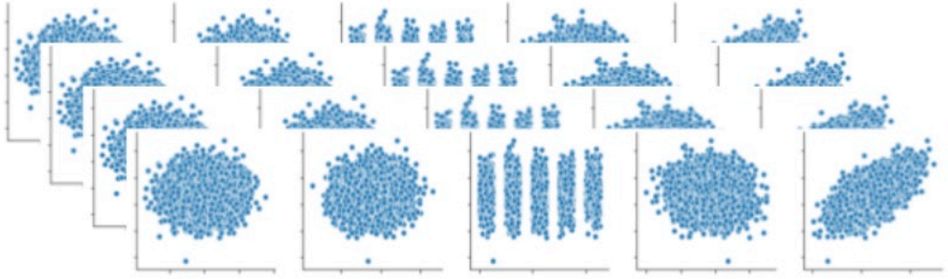
Principal Component Analysis

The Curse of Dimensionality

Let's say we have a data set of dimension **300** (n) × **50** (p). n represents the number of observations and p represents the number of predictors. Since we have a large p = 50, there can be p(p-1)/2 scatter plots, i.e **more than 1000 plots possible to analyze** the variable relationship.

Is it possible to perform good exploratory analysis on this data ?





Source: analyticsvidhya.com

Principal Component Analysis

The Curse of Dimensionality

Ideally, **we might want to include every feature** from our data set in our models. But two serious problems limit us because of the dimensionality curse:

The **calculation time** can increase exponentially each time a feature is added to the model.

(Therefore, running models with many features may become impossible due to the time it takes to run.)

The number of data points required to fit our models **increases exponentially** for each feature added to the model.

Principal Component Analysis

Dimension Reduction

FEATURE SELECTION

Feature importance

Visual Inspection

...

FEATURE TRANSFORM

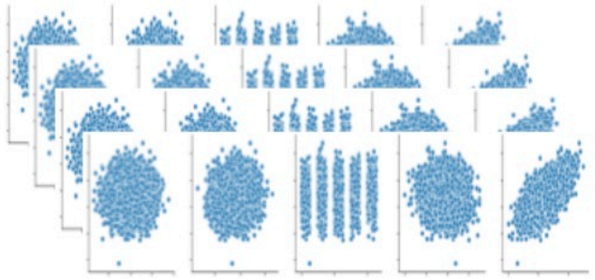
PCA

...

Principal Component Analysis

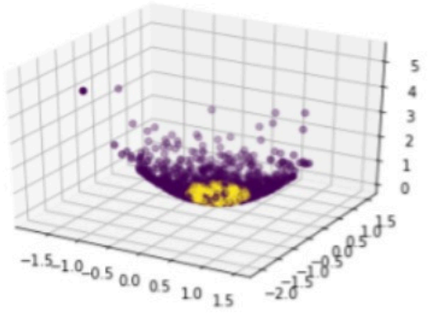
Why Do We Need Dimension Reduction?

EDA Difficulty



Visualization is one of the easiest ways to grasp data.

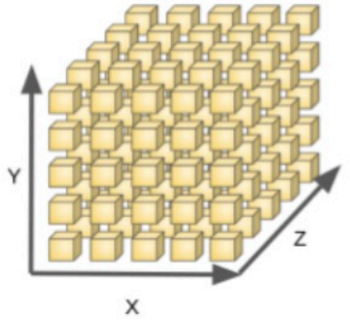
But visualize > 3D?



Some machine learning algorithms are affected by the **multidimensionality curse**

The more features we have, the more data we need.

Additional data collection is expensive and often not possible.

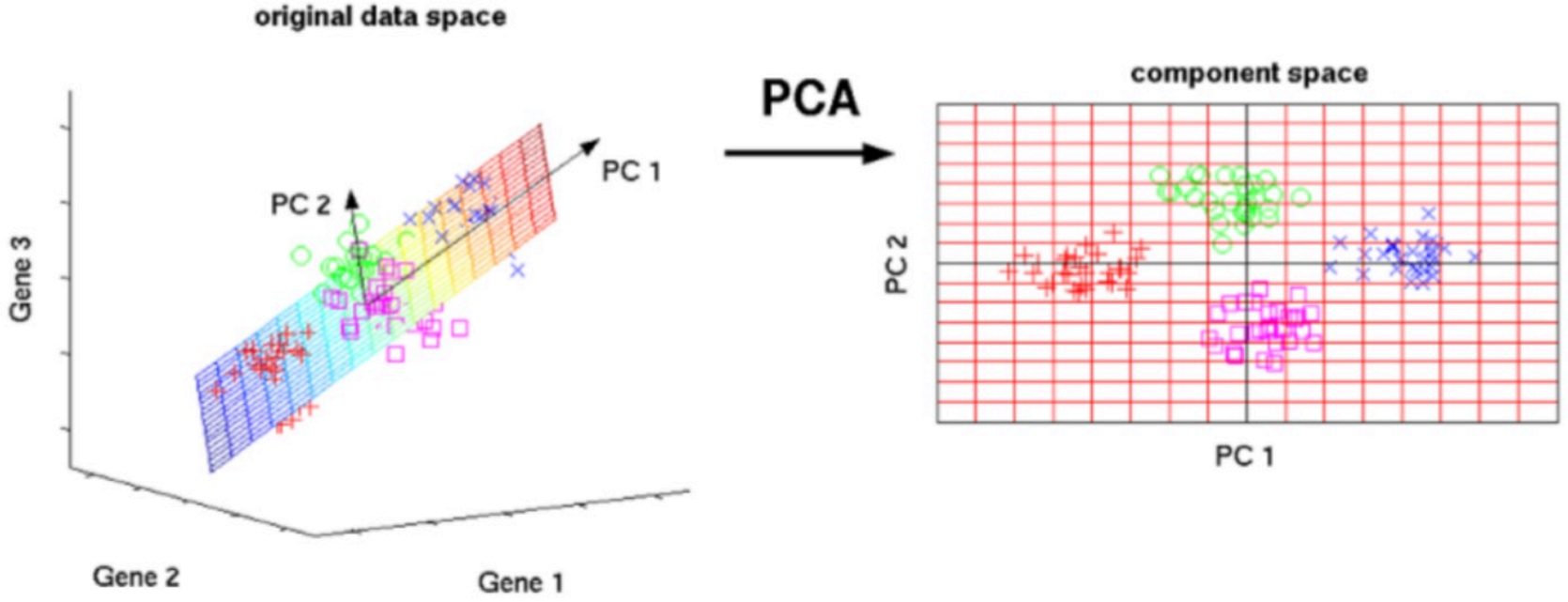


Principal Component Analysis

Dimensionality Reduction Method

Among the various dimension reduction methods, there is one **popular technique** that every data scientist should know.

This technique Linear projection method to reduce the number of parameters is called **Principal Component Analysis** (PCA).



Principal Component Analysis

- 2 human
- children
- long hair
- put their hands together
- height

etc.

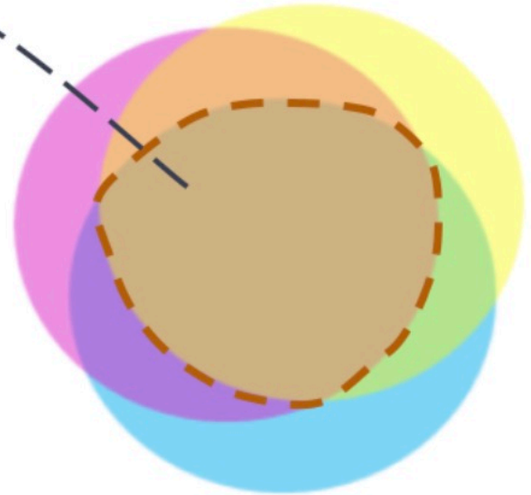
Principal Component Analysis

PCA is a complexity reduction technique that tries to reduce it to a smaller set of **components representing most of the information** in variables.

2D (shadow of the children) *represent most of the information of real picture* (**3D**)

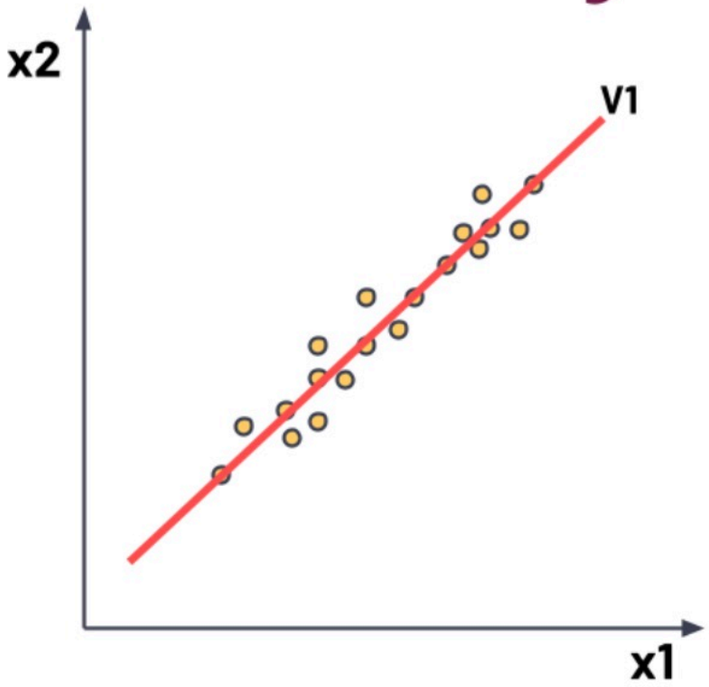
Principal Component Analysis

At the technical level, **PCA identifies sets of variables** that share variance and creates a component to represent this variance.



The area marked in Set A contains approximately ***70% of the information*** contained in the pink, blue and yellow circles.

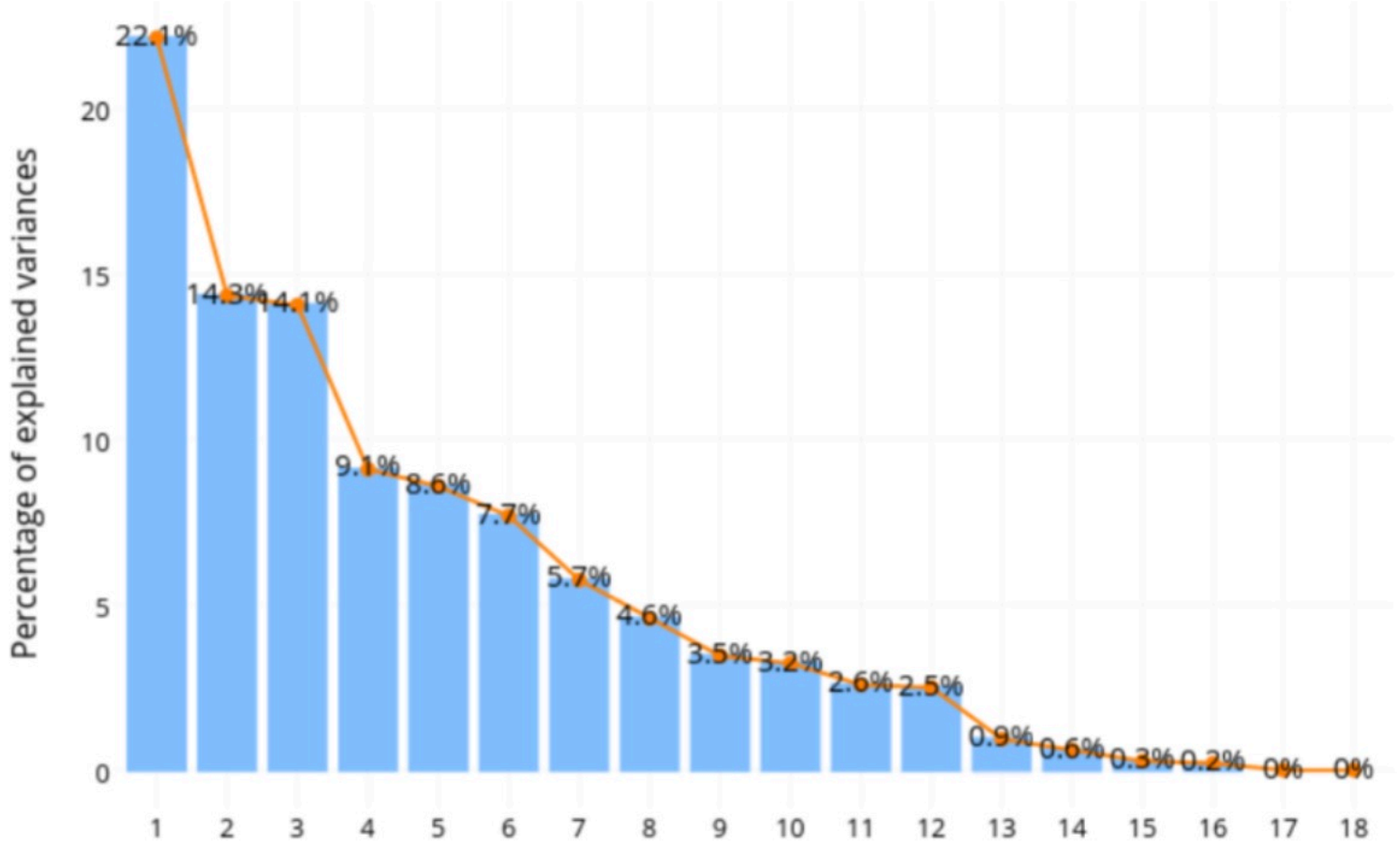
Set A



PCA for on the left or on top will likely result in a component(Set A, V1) that represents the variance shared by all variables, but infers the rest of the information in the data.

Principal Component Analysis

What are Principal Components?



Principal components (PC) basically refer to the new variables constructed as a linear combination of initial features, such that **these new variables are uncorrelated**.

It means the principal components are **independent of each other**.

Principal Component Analysis

Steps of PCA

Standardization:

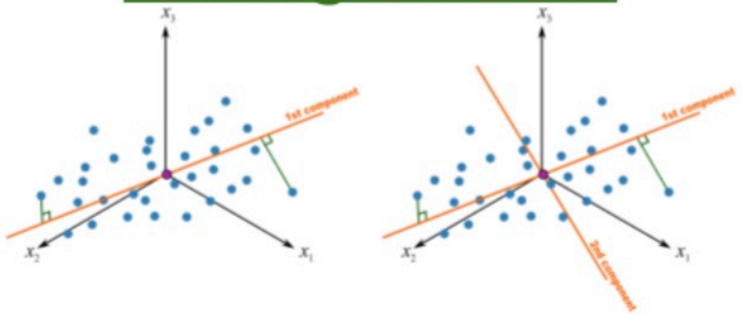
Scale the data so that each variable contributes equally to analysis.

(Remember, PCA can be applied only on numerical data.)

Covariance Matrix Computation

$$\Sigma = \begin{pmatrix} 1 & .5 & .15 & .15 & 0 & 0 \\ .5 & 1 & .15 & .15 & 0 & 0 \\ .15 & .15 & 1 & .25 & 0 & 0 \\ .15 & .15 & .25 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & .10 \\ 0 & 0 & 0 & 0 & .10 & 1 \end{pmatrix},$$

Compute Eigenvectors and Eigenvalues



Choose "k" eigenvectors with the largest eigenvalues

k is the number of dimensions (**principal components**) you wish to have in the new dataset.

Source: https://heartbeat.fritz.ai/Nikita Sharma

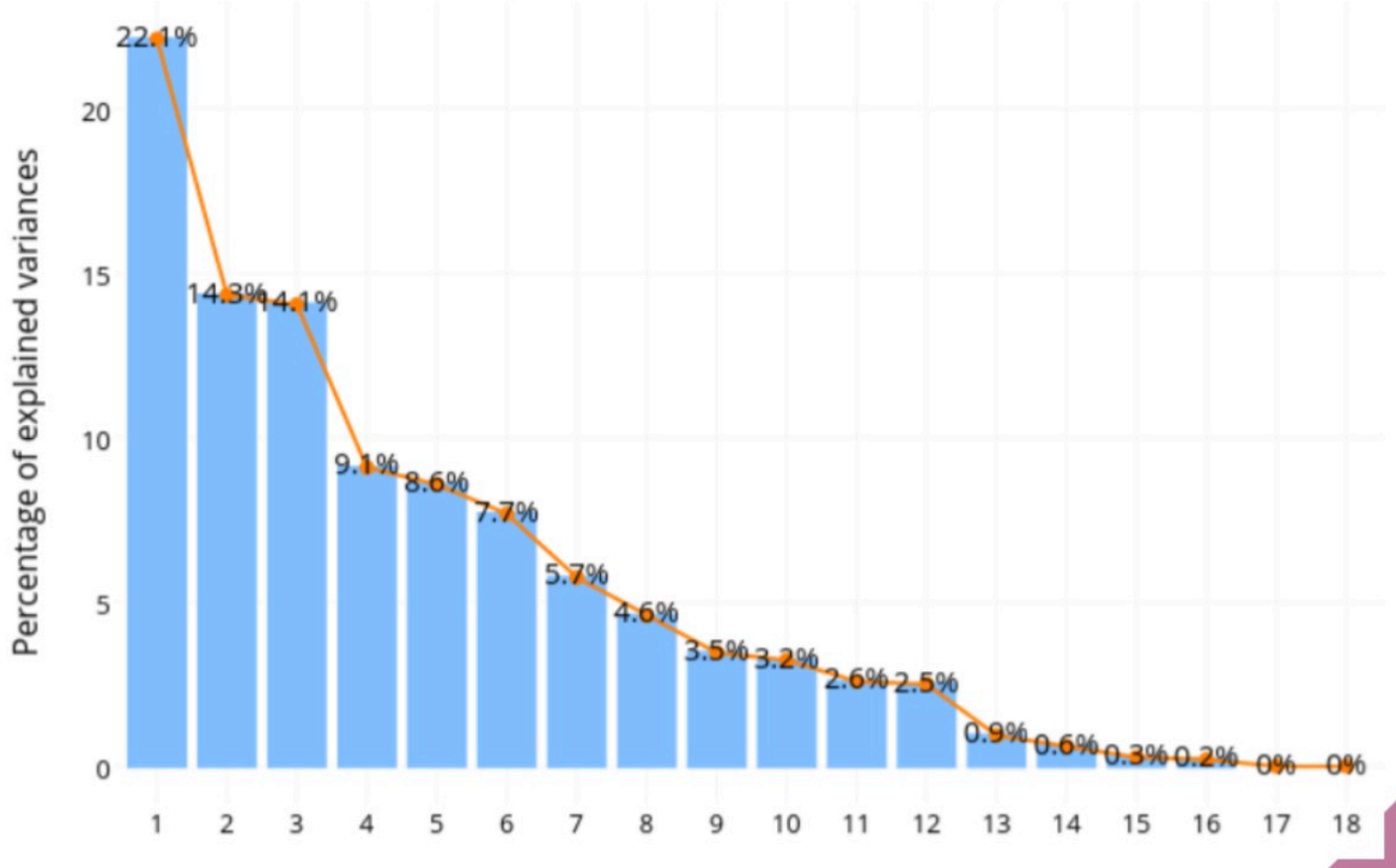
Principal Component Analysis

sklearn.decomposition.PCA

class sklearn.decomposition.**PCA**(*n_components=None, *, copy=True, whiten=False, svd_solver='auto', tol=0.0, iterated_power='auto', random_state=None*) [source]

Choose "k" eigenvectors with the largest eigenvalues

k is the number of dimensions you wish to have in the new dataset.



Principal Component Analysis

Advantages:

– Eradication of correlated features

(All the principal components are independent of one another)

– Improves algorithm performance

(If the input dimensions are too high, then PCA can be used to speed up the algorithm)

– Reduces overfitting

(Overfitting mainly occurs when there are too many variables in the dataset.)

– Improves visualization

(PCA transforms high-dimensional data to low-dimensional data to make the visualization easier.)

Disadvantages:

– Less interpretable

(Principal components are not as readable and interpretable as original features.)

– Data standardization is necessary

(You must standardize your data before implementing PCA; otherwise, PCA will not be able to find the optimal principal components.)

– Loss of Information

(It may miss some information as compared to the original list of features.)

Source: https://heartbeat.fritz.ai/Nikita Sharma

Principal Component Analysis

Be ready for

PCA

Python

Session