Data Analytics – Process of Examining raw data with the purpose of drawing conclusion about that information.

Vector Space : also known as linear Space., is a fundamental concept in linear Algebra.
It is a collection of vectors, which are objects that can be added together and multiplied ("Scaled") by numbers, called scalars in this Context.
– A vector space must satisfy a few key properties, related to vector addition & Scalar multiplication.

1. Closure under Addition
   If v_1 & v_2 are vectors in space, then their sum $v_1 + v_2$ is also in space.
   eg. $v_1 = [3, 4]$, $v_2 = [-1, 2]$ ⟹ $v_1 + v_2 = [3+(-1), 4+2] = [2, 6]$

2. Closure under Scalar Multiplication
   – When you multiply any vector (or Matrix) within a set by a scalar (a real no.), the resulting vector (or Matrix) also remain within the set.
   eg. If v is a vector and c is a scalar, then the product $c \times v$ is also in the space.

→ Operation on data structures (such as vectors, Matrices, or datasets)
– Specifically, a data set or a vector space is closed under scalar multiplication, if scaling the data by a constant results in a valid data point or a data structure, i.e. still consistent with the structure of original dataset.
eg. Feature Scaling in ML.
Consider a scenario, where you are working with a dataset of features for a ML model. Let's say each feature is represented by a vector in a vector space. The vector space is closed under scalar multiplication, if scaling any feature (multiplying it by a scalar) still results in a valid feature vector.

Age	Income
25	50000
30	60000
22	45000

You have a dataset with two features Age & Income of a group of individuals. This can be represented as a Matrix:

$$X = \begin{pmatrix} 25 & 50000 \\ 30 & 60000 \\ 22 & 45000 \end{pmatrix}$$

Now let's say you scale the Age feature by a constant scale value $c = 2$. The new feature matrix becomes

$$X_{scaled} = \begin{pmatrix} 50 & 50000 \\ 60 & 60000 \\ 44 & 45000 \end{pmatrix}$$

- In this case, you have scaled the Age Column, but Income remain unchanged.

The new Matrix is still a valid feature matrix for a ML model,
∴ Dataset is closed under scalar multiplication, as multiply by a scalar still results in valid data.

↳ Case where Scalar Multiplication is not valid
eg. Dealing with Binary variable (eg. Gender, where 0 = female and 1 = Male)
Consider dataset:

Gender
0
1

If we multiply the Gender feature by a scalar $c = 2$, we get

$$X_{scaled} = \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix}$$

- The new value of Gender (0, 2, 0) are no longer valid Binary values,

This means the dataset is not closed under scalar multiplication in this case.
Scalar multiplication destroys the integrity of a data, because it introduces invalid value that fall outside the expected range of the variable.

⇒ In Data Analytics - Closure under scalar multiplication ensures that transformation like scaling preserve the validity & consistency of dataset
- It depends on the nature of a data and transformation applied.

3. Commutativity Addition - in Data Analytics

When you add elements (matrices or vectors) does not affect the result. eg. $a+b = b+a$

This means that the sum of two elements is independent of the order in which they are added. This property is crucial, when manipulating data in datasets, as it ensures consistency like aggregating or combining data.

Eg. working with Vectors (Feature Data)

Consider two data points (vectors) representing features of two individuals in a dataset, say age & income.

- vector1 (individual1): $v1 = \begin{pmatrix} 25 \\ 50000 \end{pmatrix}$
- vector2 (individual2): $v2 = \begin{pmatrix} 30 \\ 60000 \end{pmatrix}$

If we add these vectors, we get:

$$v1 + v2 = \begin{pmatrix} 25 \\ 50000 \end{pmatrix} + \begin{pmatrix} 30 \\ 60000 \end{pmatrix} = \begin{pmatrix} 55 \\ 110000 \end{pmatrix}$$

Now, if reverse order of addition

$$v2 + v1 = \begin{pmatrix} 30 \\ 60000 \end{pmatrix} + \begin{pmatrix} 25 \\ 50000 \end{pmatrix} = \begin{pmatrix} 55 \\ 110000 \end{pmatrix}$$

→ As we can see, the order of addition does not affect the result.

∴, the operation of addition is commutative.

⇒ Real life Application: Aggregation of Data.

Imagine you are analyzing Sales data across multiple regions or stores. You may need to aggregate sales data from different stores, the commutative property of addition ensures that it does not matter, which store's data, you add first, total sales remain same.

- Sale 1: $v1 = [500, 400, 600]$
- Sale 2: $v2 = [300, 200, 150]$

Aggregating Sale Data: $v1 + v2 = [800\ 600\ 750]$

And if we reverse the order $v2 + v1 = [800\ 600\ 750]$ ⟹ Again, the order of addition doesn't matter, showing that commutativity holds when aggregating data.

Why is commutativity so important?

- Consistency - When performing operation like summation, commutativity property ensures that the result will be consistent, regardless of order of operation. This is important when aggregating data or combining features.
- Efficiency in Calculation - In large datasets, algorithms can take advantage of commutativity to optimize calculation. eg distributing the computation across multiple processors.
- Data Integrity - Commutativity guarantees that when you are analyzing data (eg. sum, total), the order in which data is processed does not affect the outcome.

1. **Associativity of Addition**:
- States that when you add three or more numbers, the grouping of those numbers does not affect the result.

$(a+b)+c = a+(b+c)$

- When performing operation like data aggregation, feature engineering or summarizing multiple data points.

Eg. Working with vectors (Data Points):

Consider three data points (represents as vectors), each corresponding to a person's age and income.

$v1 = \begin{pmatrix} 25 \\ 10000 \end{pmatrix}$; $v2 = \begin{pmatrix} 30 \\ 60000 \end{pmatrix}$; $v3 = \begin{pmatrix} 22 \\ 45000 \end{pmatrix}$

FIRST GROUPING

$(v1+v2)+v3 = \left(\begin{pmatrix} 25 \\ 10000 \end{pmatrix} + \begin{pmatrix} 30 \\ 60000 \end{pmatrix}\right) + \begin{pmatrix} 22 \\ 45000 \end{pmatrix} = \frac{77}{155000}$

SECOND GROUPING

$v1+(v2+v3) = \begin{pmatrix} 25 \\ 10000 \end{pmatrix} + \left(\begin{pmatrix} 30 \\ 60000 \end{pmatrix} + \begin{pmatrix} 22 \\ 45000 \end{pmatrix}\right) = =$

Thus, Addition is Associative, and the order in which we group the vectors for addition does not change the final results.

- Working with Dataframes – First + Grouping.

$(df1+df2)+df3 = \left(\begin{pmatrix} 200 & 250 & 300 \\ 150 & 200 & 250 \end{pmatrix} + \begin{pmatrix} 300 & 350 & 400 \\ 250 & 250 & 300 \end{pmatrix}\right) + \begin{pmatrix} 100 & 150 & 200 \\ 50 & 100 & 150 \end{pmatrix}$

$= \begin{pmatrix} 500 & 600 & 700 \\ 350 & 450 & 550 \end{pmatrix} + \begin{pmatrix} 100 & 150 & 200 \\ 50 & 100 & 150 \end{pmatrix} = \begin{pmatrix} 600 & 750 & 900 \\ 400 & 550 & 700 \end{pmatrix}$

Second Grouping

$df1+(df2+df3) =$ ✓

⇒ In Data Analytics, the Associativity of Addition allows flexibility and consistency when performing operation such as Aggregating, combining or summing data points, vector or matrices

5. Existence of Additive Inverse :

Additive inverse of a number is the value that, when added to the original number, results in zero.
Every number has an additive inverse, and it is found by simply changing the sign of the original number.

$$a + (-a) = 0$$

Suppose we have 2×2 matrix representing sales data for two product across two shops.

$$M = \begin{pmatrix} 100 & 150 \\ 200 & 250 \end{pmatrix}$$

The additive inverse of Matrix M is simply the matrix with all values negated

$$-M = \begin{pmatrix} -100 & -150 \\ -200 & -250 \end{pmatrix}$$

Now let's add M and $-M$. $= \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$

Subspace :

In a high dimensional dataset, many dimensions might be irrelevant or noisy, masking underlying patterns and clusters.
Subspace Analysis, focuses on finding meaningful clusters within specific subsets (subspace) of the original dimension.

Why use Subspace Analysis ?

- It helps to reduce the complexity of high dimensional data by focusing on the most relevant features.
- Subspace clustering algorithms can identify clusters that might be hidden or obscured in the full dataset.
- Subspace methods can also be used for outlier detection by identifying data points that deviate significantly from the subspace characteristics

eg. Customer segmentation in a E-commerce Platform.
[How subspace are used to simplify complex, high-dimensional data, uncover patterns & improve decision making].

Suppose you are working for an e-commerce company that sells a wide variety of Product online
You have custom data that include feature like
Age, Gender, Time Spent on Site, No. of Purchase, Total Amt Spend, Product Category Purchased, Geographic region (City, State, country)
The dataset has many dimension (features) and it would be hard to manually analyze and understand customer behavior directly from the data.
However, we can use subspace to reduce data dimensionality identify key patterns and create meaningful segments of customers.

PCA

1. Initial Dataset: High dimensional Feature Space.

CustId	Age	Gender	Time Spent	Purchase	Total Amt Spent	Product category	Region
1	25	M	45	5	150	Electronics	Delhi
2.	34	F	30	3	90	Home Goods	Mumbai
3.	45	M	60	10	350	Sports	Delhi
4.	22	F	20	2.	50	Apparel	Chennai
5.	36	M	50	7	220	Sports	Delhi

2. Challenges: High dimensionality.
While the data can tell you some information about individual customers, it is hard to find patterns.
Instead of Analyzing the full 7-features (dimensional space) we can reduce the complexity by finding a subspace that better capture key features of customer behavior.

3. Dimensionality Reduction
- PCA (Principal Component Analysis)
  - a technique used to identify direction (or components) in which the data varies the most and reduce the dimensionality of data.
- To perform PCA, on the given data set, we need to focus on numerical columns that PCA can process.
- PCA works on Numerical data, So we can exclude categorical data such as Gender, Product Category, Purchased and Region.

→ Steps for PCA
1. Prepare the data - organize numerical Column for PCA and standardize the data.
2. Standardize the data - Standardization is important, when features have different scales, So we Normalize the data to have a mean '0' and a standard deviation of 1.
3. Compute Covariance Matrix
4. Find Eigenvalue & Eigenvector - the eigenvectors corresponding
5. Sort Eigenvalue & Eigenvector - the principal components to the largest eigenvalue are
6. Project the Data - Transform the data onto the new set of principal components.

PCA) Step 1 : Standardization – Calculate Mean & std. deviation of each column (Age, Time Spent, Purchase, Total Amt Spent) of Data Set

CustID	Age	Time Spent	Purchase	Total Amt Spent
1	25	45	5	150
2	34	30	3	90
3	45	60	10	350
4	27	90	2	50
5	36	50	7	220

Simplify Problem

- **Mean** – is the sum of value divided by the number of values. It represents central tendency of data

$$\text{Mean} = \frac{\sum x_i}{N}$$

  - x_i – each data Point
  - N – No. of Data Point (In this case, 5)

- **Standard Deviation** – tell how you spread out the values are from the Mean. A high standard deviation mean the data points are spread out, while a low standard deviation means they are close to mean.

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - M)^2}$$

  - x_i – each data point
  - M – Mean of data point
  - N – No of data Point (here, 5)

1. **Age** : Mean $= \frac{25+34+45+27+36}{5} = 33.4$

- **Standard deviation** – First, calculate the squared difference from Mean.

$(25-33.4)^2 = 70.56$; $(34-33.4)^2 = 0.36$; $(45-33.4)^2 = 133.96$

$(27-33.4)^2 = 40.96$; $(36-33.4)^2 = 6.76$

Now divide by N = 5, and take the square root.

Sum of squared difference $= 70.56 + 0.36 + 133.96 + 40.96 + 6.76 = 252.6$

Standard deviation $= \sqrt{\frac{252.6}{5}}$

$= \sqrt{50.52}$

$= 7.12$

2. Time Spent : value : 45, 30, 60, 20, 50

$$\text{Mean} = \frac{45+30+60+20+50}{5} = \frac{205}{5} = 41$$

Standard deviation :

$(45-41)^2 = 16$, $(30-41)^2 = 121$, $(60-41)^2 = 361$

$(20-41)^2 = 441$, $(50-41)^2 = 81$

Sum of squared difference :

$16 + 121 + 361 + 441 + 81 = 1020$

$$\text{Std. Deviation} = \sqrt{\frac{1020}{5}} = \sqrt{204} = 14.28$$

3. Purchase : value : 5, 3, 10, 2, 7.

$$\text{Mean} = \frac{5+3+10+2+7}{5} = \frac{27}{5} = 5.4$$

~~Sum of squared difference~~

standard deviation - First calculate sum of squared differences.

$(5-5.4)^2 = 0.16$, $(3-5.4)^2 = 5.76$, $(10-5.4)^2 = 21.16$

$(2-5.4)^2 = 11.56$, $(7-5.4)^2 = 2.56$

Sum of squared differences

$0.16 + 5.76 + 21.16 + 11.56 + 2.56 = 41.2$

Now divide by $N = 5$ and take the square Root

$$\text{Std. Deviation} = \sqrt{\frac{41.2}{5}} = \sqrt{8.24}$$

$$= 2.87$$

4. Total Amt. Spent: Value: 150, 90, 350, 50, 220

$$\text{mean} = \frac{150 + 90 + 350 + 50 + 220}{5} = \frac{860}{5} = 172$$

Std. deviation - First calculate squared difference from Mean,

$(150 - 172)^2 = 484$, $(90 - 172)^2 = 6724$, $(350 - 172)^2 = 31584$

$(50 - 172)^2 = 14884$, $(220 - 172)^2 = 2304$

Sum of squared differences:

$484 + 6724 + 31584 + 14884 + 2304 = 55176$

Now divide by N = 5, and take the squar Root:

$$\text{std. deviation} = \sqrt{\frac{55176}{5}} = \sqrt{11035.2} = 105.06$$

⇒ Summary

Column	Mean	Standard Deviation
Age	33.4	7.12
Time Spent	41	14.28
Purchase	5.4	2.87
Total Amt Spent	172	105.06

Step 1: Mean and std. deviation

Step 2: Covariance Matrix - After Standardization, calculate the Covariance Matrix to see, how each feature correlate with other

Now, we need to calculate the Covariance between each pair of variables in the data set. Covariance Matrix will have following form:

$$\begin{bmatrix} Cov(X,X) & Cov(X,Y) & Cov(X,Z) & Cov(X,W) \\ Cov(Y,X) & Cov(Y,Y) & Cov(Y,Z) & Cov(Y,W) \\ Cov(Z,X) & Cov(Z,Y) & Cov(Z,Z) & Cov(Z,W) \\ Cov(W,X) & Cov(W,Y) & Cov(W,Z) & Cov(W,W) \end{bmatrix}$$

- X = Age
- Y = Time Spent
- Z = Purchase
- W = Total Amt. Spent

The Covariance between two variables x and y is given by

$$Cov(x,y) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - M_x)(y_i - M_y)$$

- x_i & y_i are individual data points of each variable
- M_x and M_y are the mean of the variables
- n is the no. of data point

Variance of a variable is just the covariance of the variable with itself:

$$Cov(X,X) = Var(X)$$

which is equivalent to the square of the standard deviation of the variable:

$$Var(X) = \sigma^2_x.$$

So, if we assume zero correlation, the covariance matrix would be:

$$\begin{bmatrix} \sigma^2_{age} & 0 & 0 & 0 \\ 0 & \sigma^2_{Time\,Spent} & 0 & 0 \\ 0 & 0 & \sigma^2_{Purchase} & 0 \\ 0 & 0 & 0 & \sigma^2_{Total\,Time\,Spent} \end{bmatrix} \Rightarrow \begin{bmatrix} (7.12)^2 & 0 & 0 & 0 \\ 0 & (14.28)^2 & 0 & 0 \\ 0 & 0 & (2.87)^2 & 0 \\ 0 & 0 & 0 & (105.06)^2 \end{bmatrix}$$

Thus, Covariance Matrix is:

$$\begin{bmatrix} 50.6944 & 0 & 0 & 0 \\ 0 & 204.9184 & 0 & 0 \\ 0 & 0 & 8.2369 & 0 \\ 0 & 0 & 0 & 11043.0036 \end{bmatrix}$$

Interpretation of Covariance Matrix

- The diagonal elements represents the variance of each variable, showing how spread out the data for each variable is.
- The off-diagonal elements (which are all zero) represent the covariance b/w di variables.

Since we assumed zero correlation, the covariance value between different variables is zero.

If Correlation exist — if you had raw data or correlation coefficients between the variable, you could calculate the off-diagonal elements by multiplying std. deviations of the respective variable and the correlation coefficient.

e.g., if the correlation between Age and TimeSpent 0.3, the covariance would be:

$$Cov(Age, TimeSpent) = Correlation(Age, TimeSpent) \times \sigma_{Age} \times \sigma_{TimeSpent}$$

Step 3: Eigenvalues and Eigenvectors

[Compute it, of Covariance Matrix.

$$\begin{bmatrix} 50.6944 & 0 & 0 & 0 \\ 0 & 304.9184 & 0 & 0 \\ 0 & 0 & 8.2369 & 0 \\ 0 & 0 & 0 & 11043.0036 \end{bmatrix}$$

- This is a diagonal matrix, where all off-diagonal elements are zero, indicating no correlation between the variable
- Eigenvalue — each eigenvalue corresponds to the variance along a principal component direction
- Eigenvector — Since Covariance Matrix is diagonal, the eigenvectors will be the standard basis vector

Calculate the Eigenvalue

For the diagonal Matrix, Eigenvalue are simply diagonal value. Independent themselves.

$\lambda_1 = 50.6944$, $\lambda_2 = 204.9184$, $\lambda_3 = 8.269$, $\lambda_4 = 110.43$..

Calculate the Eigenvectors — For a diagonal matrix, the Eigenvectors are straightforward. Each Eigenvector corresponds to one the dimensions in data space (Feature: Age, TimeSpent, Purchase, Total Amt Spent).

$$v_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, v_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, v_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, v_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Each Eigenvector corresponds to the direction of one feature:

v_1 – corresponds to Age; v_2 – corresponds to Time Spent

v_3 – corresponds to Purchase; v_4 = corresponds to Total Amt Spent.

Eigenvalues ✓

- $\lambda_1 = 50.6944$, indicates the variance explained by the first Principal Component (Age)
- $\lambda_2 = 204.9184$, — 2nd, principal Compond (Time Spent)
- $\lambda_3 = 8.2869$, — 3rd Principal Component (Purchase)
- $\lambda_4 = 110.0036$, — 4th, Principal Component (Total Amt Spent)

Eigenvector – being standard basis vectors, indicate that the features (Age, TimeSpent, Purchases, Total Amt Spent) are independent of each other in this case. There is no mixing between them along the principal components.

- $v_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ (Age), $v_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$ (Time Spent), $v_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$ (Purchase), $v_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$ (Total Amt. Spent)

These results shows that each feature in the dataset contributes to a distinct principal component, with no mixing or interaction between them.

Independence of Vector

Vectors do not "depend" on each other for their existence, and none of them can be written as a combination of others.
eg. Let's consider a dataset with two features (eg. Age & Income of individuals) represented as two vector.

- vector 1 (Age) $V1 = \begin{pmatrix} 25 \\ 30 \\ 22 \\ 28 \end{pmatrix}$
- vector 2 (Income) $V2 = \begin{pmatrix} 50000 \\ 60000 \\ 45000 \\ 55000 \end{pmatrix}$

Now, let's test whether these two vectors are linearly independent.
To check if they are linearly independent, we could check if there exists a nontrivial solution to the equation:

$$C_1V_1 + C_2V_2 = 0$$

This gives us the following system of equations:

$$C_1\begin{pmatrix} 25 \\ 30 \\ 22 \\ 28 \end{pmatrix} + C_2\begin{pmatrix} 50000 \\ 60000 \\ 45000 \\ 55000 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

which translate to

$$\begin{pmatrix} 25C_1 + 50000C_2 \\ 30C_1 + 60000C_2 \\ 22C_1 + 45000C_2 \\ 28C_1 + 55000C_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

For these vectors to be linearly independent, the only solution to this system of equations should be $C1=0$ and $C2=0$.
- if we attempt to solve this system, we will find that there is no nontrivial solution where both C_1 and C_2 are nonzero. Thus, these vector are linearly independent, meaning that the age and income features are not directly related to each other, in such a way that one can be written as a multiple of other. This implies that age & income are distinct features in this dataset each contributing unique information to the analysis.

⇒ Basis & dimension allow you to understand the structure of your data, reduce complexity and optimize & modeling efficiency.

- Basis - A minimal set of features that can describe the data.
- Dimension: The no. of features needed to describe your data. (i.e. - the size of the space)

⇒ Techniques like PCA and feature Selection help reduce the dimension by finding a new basis with fewer features.

eg. in context of data Analytics

- Suppose you have a dataset with multiple features (eg Age, Income, education). These features can be thought of as vectors in a higher dimensional space
- A basis would be a set of features that can describe all possible data points in your dataset.

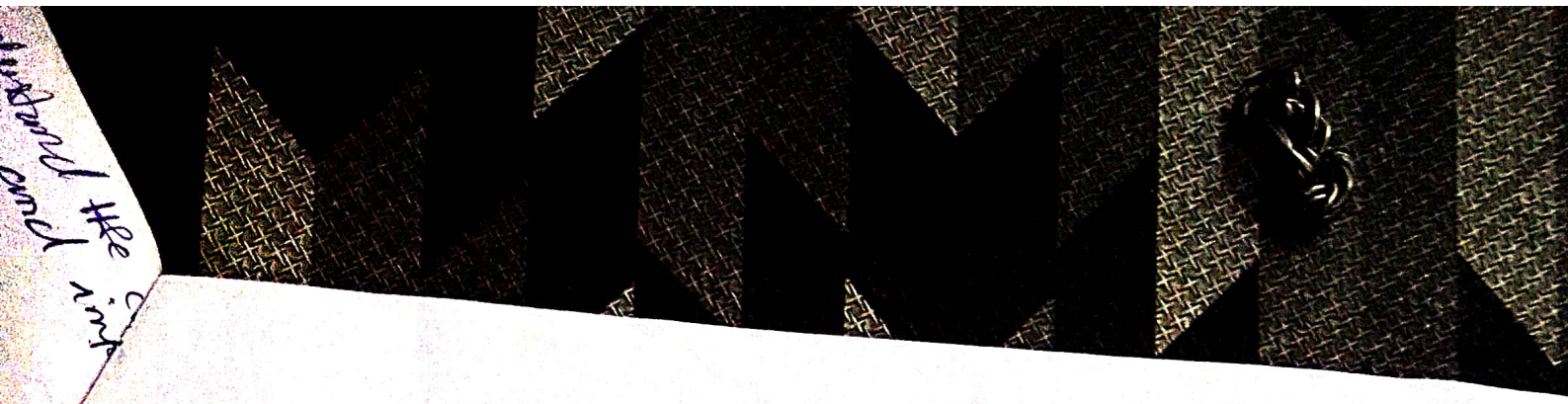
You might not need all features, and some features could be redundant, meaning they are not part of a "minimal" basis.

eg. Dataset with three variables

- x1: Age; x2: Income; x3: Edu. Level

— if you find that x1 & x2 are enough to describe all data points (say, education Level just a function of age & income), then x1 & x2 form a basis for the data space, and x3 would be redundant.

⇒ if your dataset contain 5 features (Age, income, education, years of experience, and city), then your vector space has a dimension of 5.



Data Analytics.

- process of examining raw data with the purpose of drawing conclusions about that information.