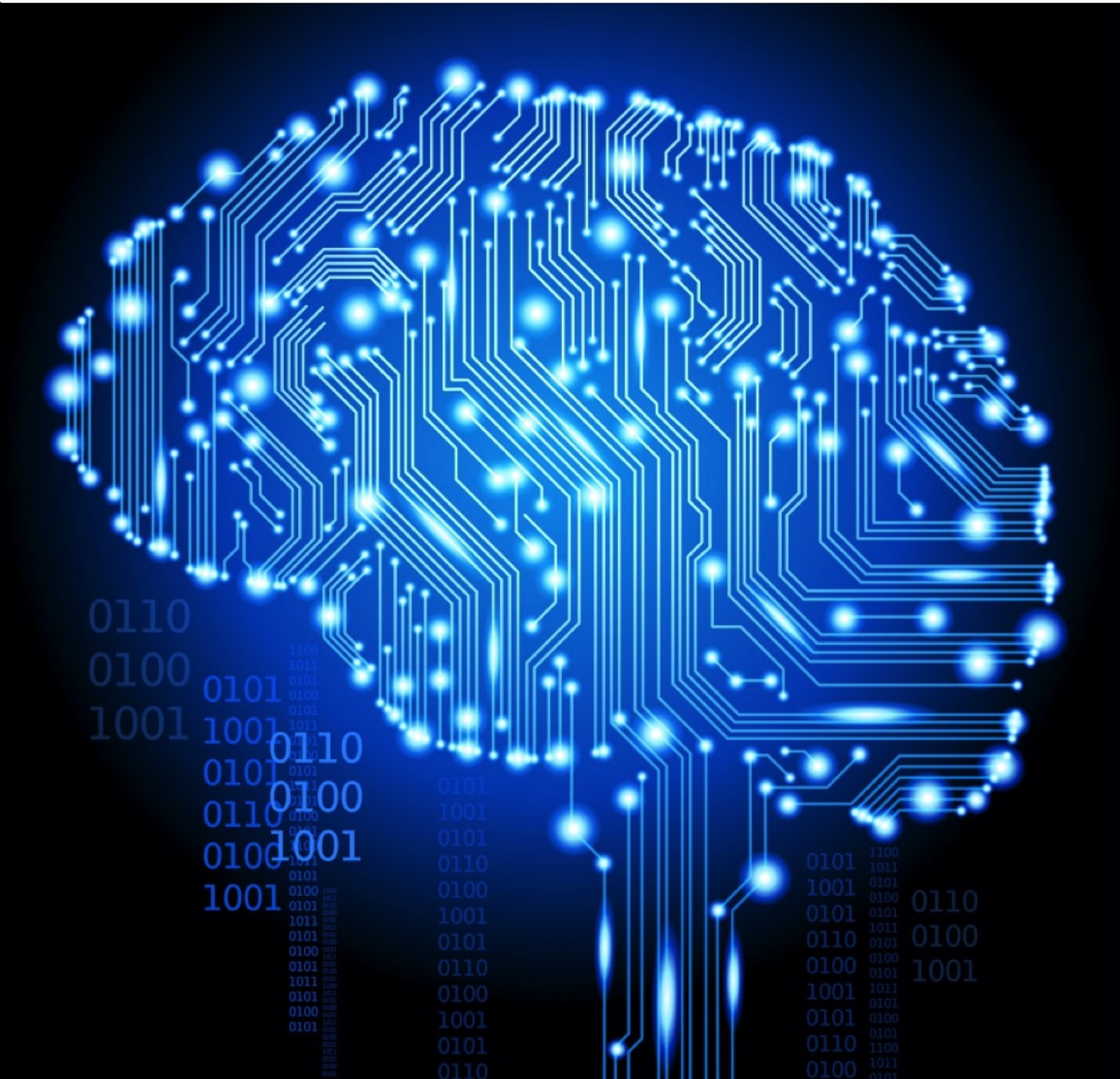
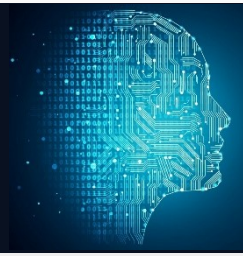
Data Science and Machine Learning Practical tools and programming Part I

Leeor Langer

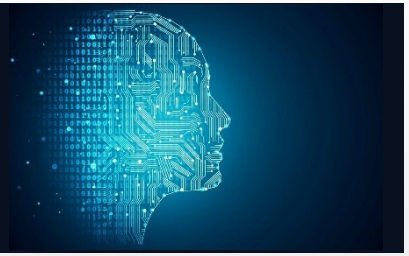


AGENDA



1. Introduction to Data Science
2. Data Preparation using various tools
3. Running Machine Learning Algorithms
4. Mini-Project Part A: Recommendation Systems

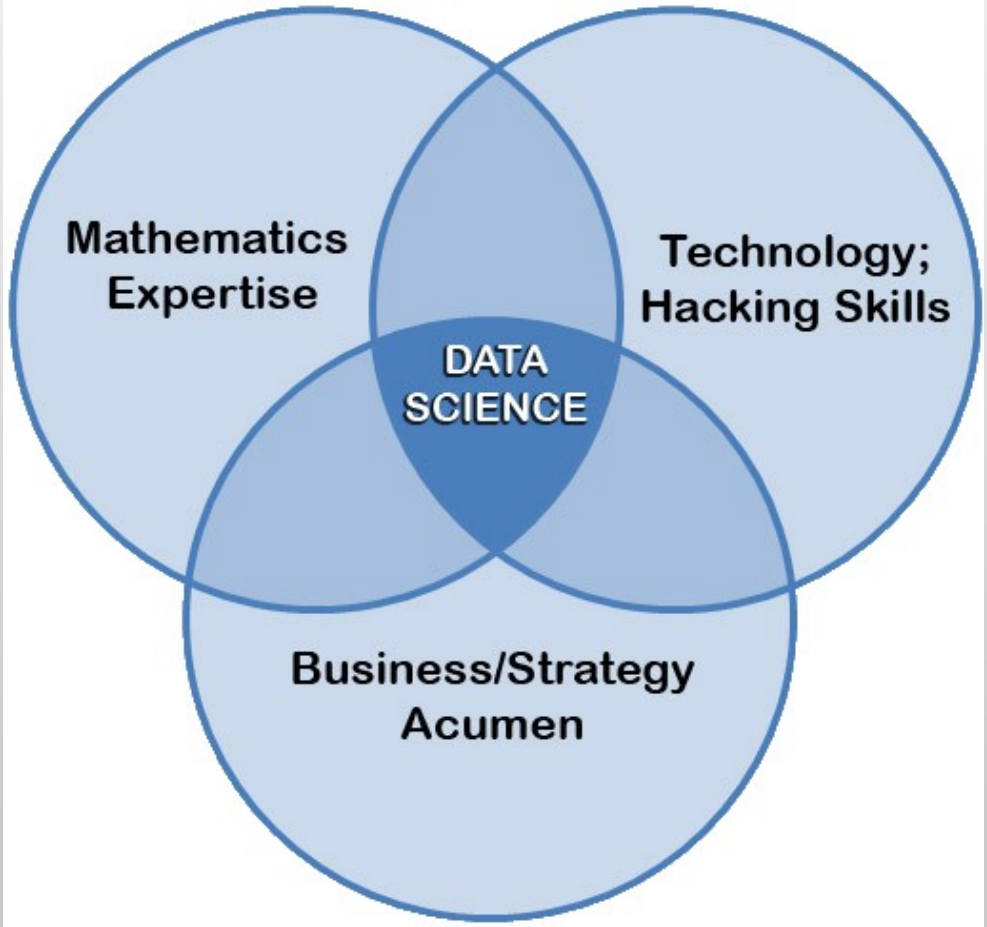
WHAT IS DATA SCIENCE?



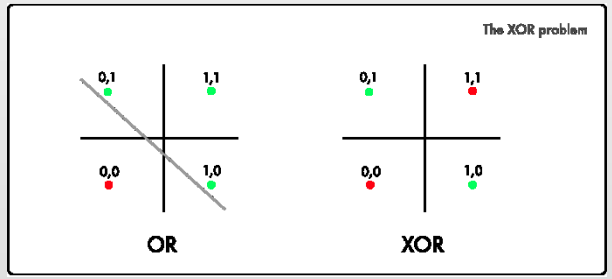
“Big Data is like teenage sex: everyone talks about it, nobody really knows how to do it, everyone thinks everyone else is doing it, so everyone claims they are doing it.”

-Dan Ariely

WHAT IS DATA SCIENCE?



AI TIMELINE



1930s
- Turing Machine (Turing)
- Incompleteness Theorem (Gödel)

1950s
- Perceptron (Rosenblatt)

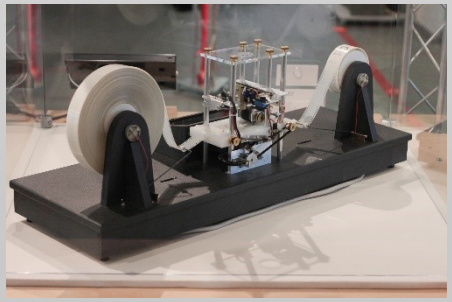
1960s
- Bayesian Methods (Solomonoff)

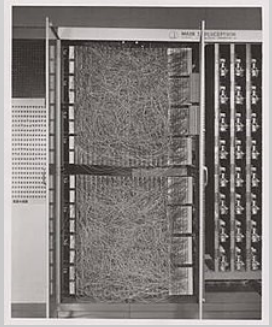
1970s
- XOR problem

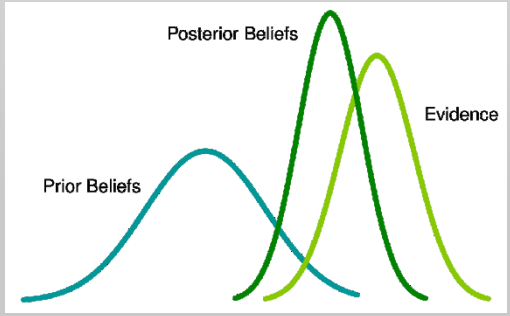
1980s
- Multi Layer Perceptron
- Backpropagation (Hinton)

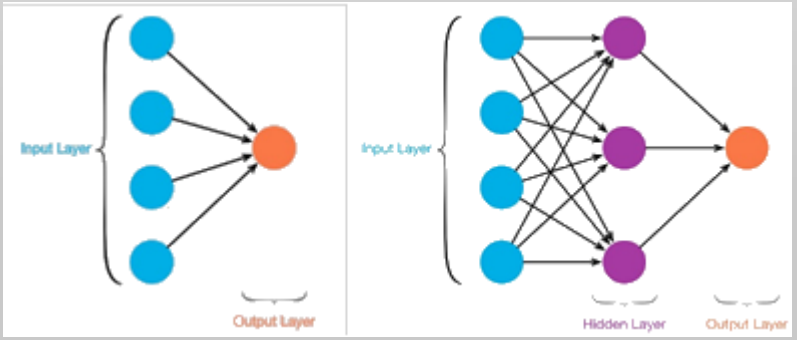
1990s
- SVM
- Random Forest
- (Vapnik, Breiman)

2000s
- Deep Neural Networks
- (Lecun, Benjio, Hinton)

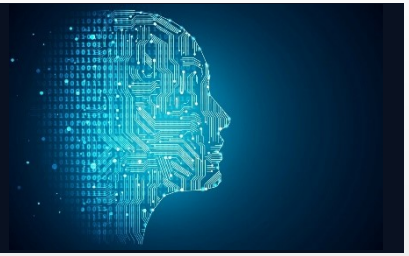








WHAT IS DATA SCIENCE?

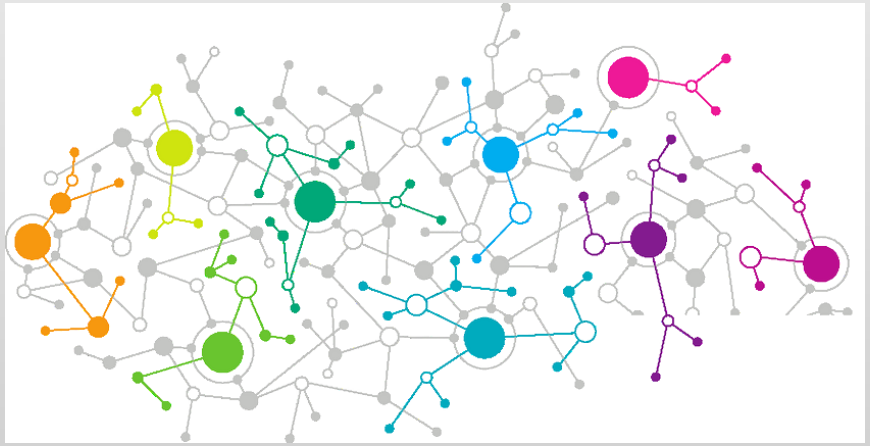


“There are two cultures in the use of statistical modeling to reach conclusions from data. One assumes that the data are generated by a given stochastic data model. The other uses algorithmic models and treats the data mechanism as unknown. The statistical community has been committed to the almost exclusive use of data models… If our goal as a field is to use data to solve problems, then we need to move away from exclusive dependence on data models and adopt a more diverse set of tools.“

- The Two Cultures, Leo Breiman

THE BIG PICTURE

Complex models

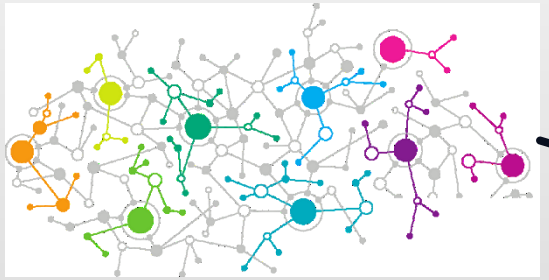


Massive Compute Power

Big Data

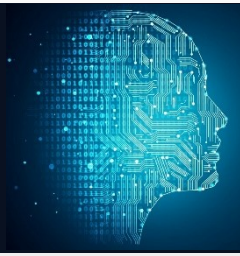
Machine Learning = Complex models + Massive Compute Power + Big Data

THE BIG PICTURE: ROCKET ANALOGY



Rocket Engine = Complex Model
Big Data = Fuel
-Andrew NG

THE BIG PICTURE



Sorting Lego Blocks

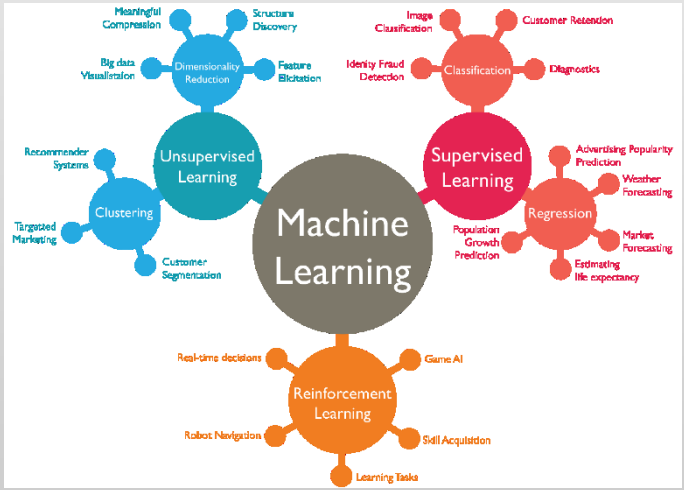
Simple example of a child sorting Lego blocks illustrates the differences between the three machine learning styles

Unsupervised Learning
Child has to sort blocks by color, shape or both with no instructions

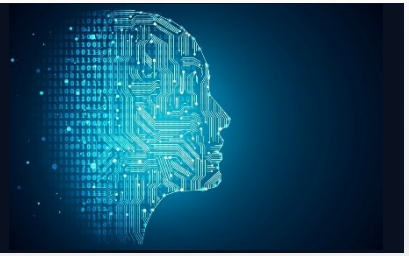
Supervised Learning
Child has to sort the colored blocks by matching the colors of the block with the colors of the bag

Reinforcement Learning
Child gets feedback from Mom when he does something right or something wrong

source pwc via @mikequindazzi



THE BIG PICTURE

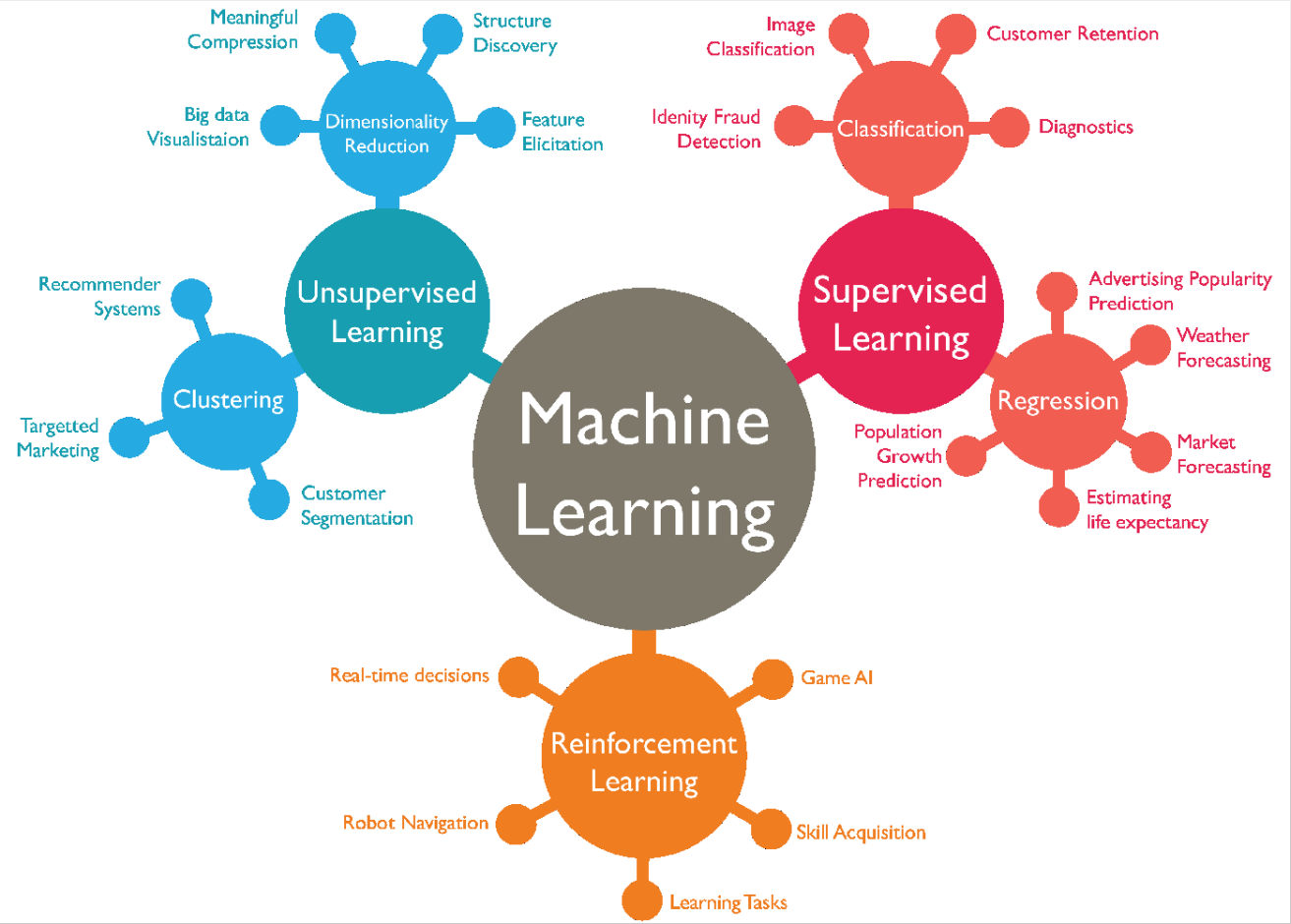


"Is it just one big Sears Catalog?"

– Gerry Fodor

1. Unsupervised Learning: $\mathrm{D} = \{X\}_{i=1}^{N}$
2. Supervised Learning: $\mathrm{D} = \{X, y\}_{i=1}^{N}$
3. Reinforcement Learning:

$$S = \{s\}_{i=1}^{N}, A = \{a\}_{i=1}^{M}, P(s_{t+1}|s_t, a)$$



THE BIG PICTURE

"Is it just one big Sears Catalog?"

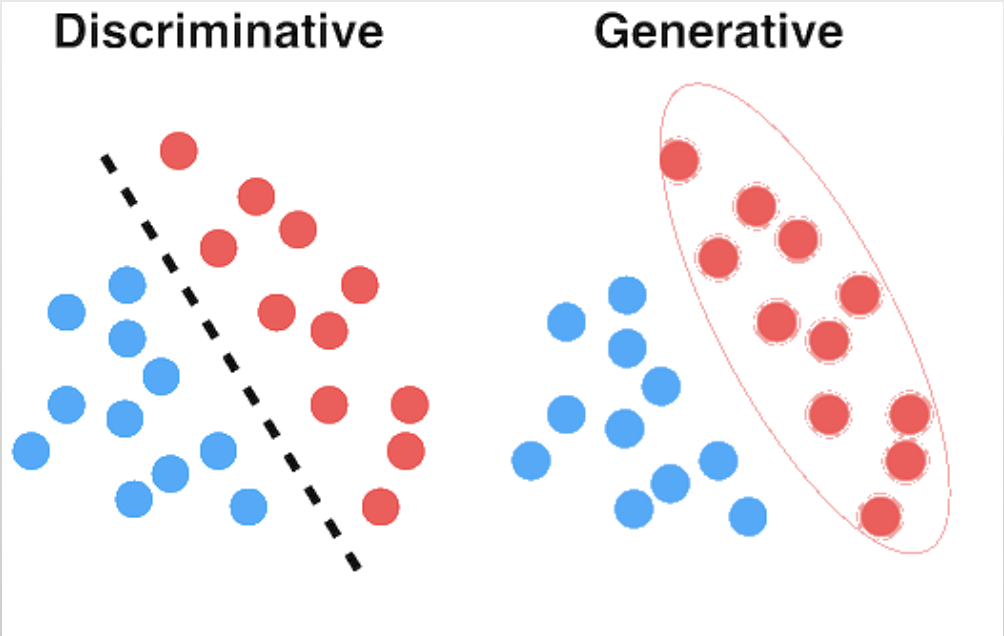
– Gerry Fodor

1. Discriminative models: $P(y|x)$

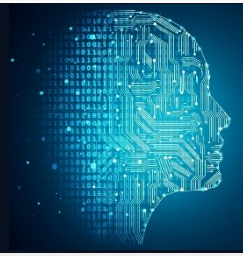
   (example: image classification)

2. Generative models: $P(x, y)$

   (example: art generation)

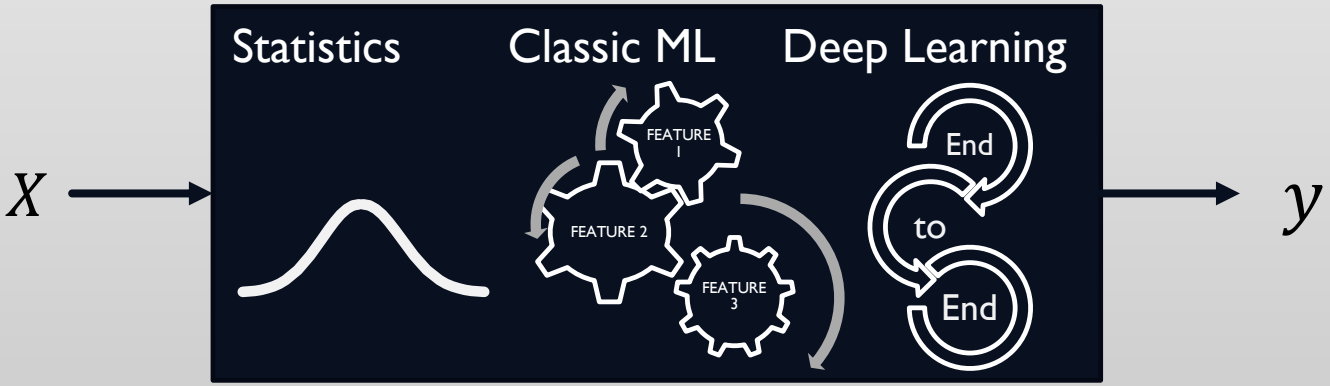


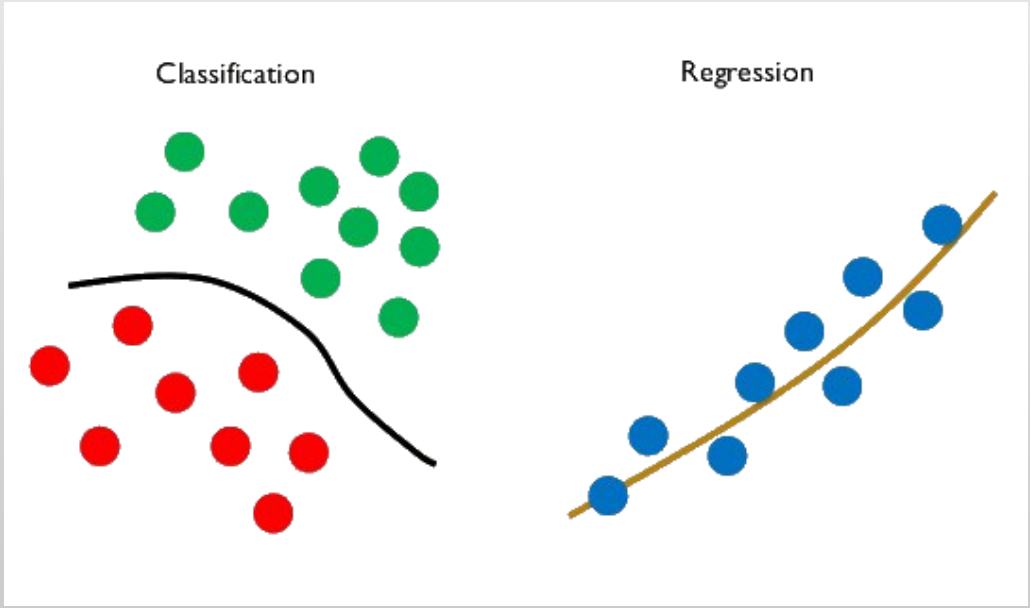
THE BIG PICTURE



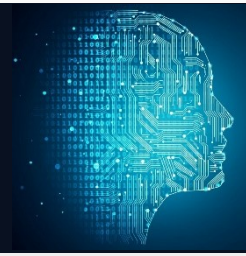
Statistics vs Classic ML vs Deep Learning

Model:

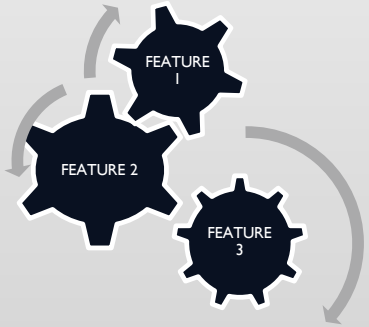




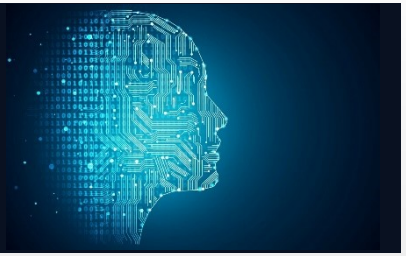
THE BIG PICTURE

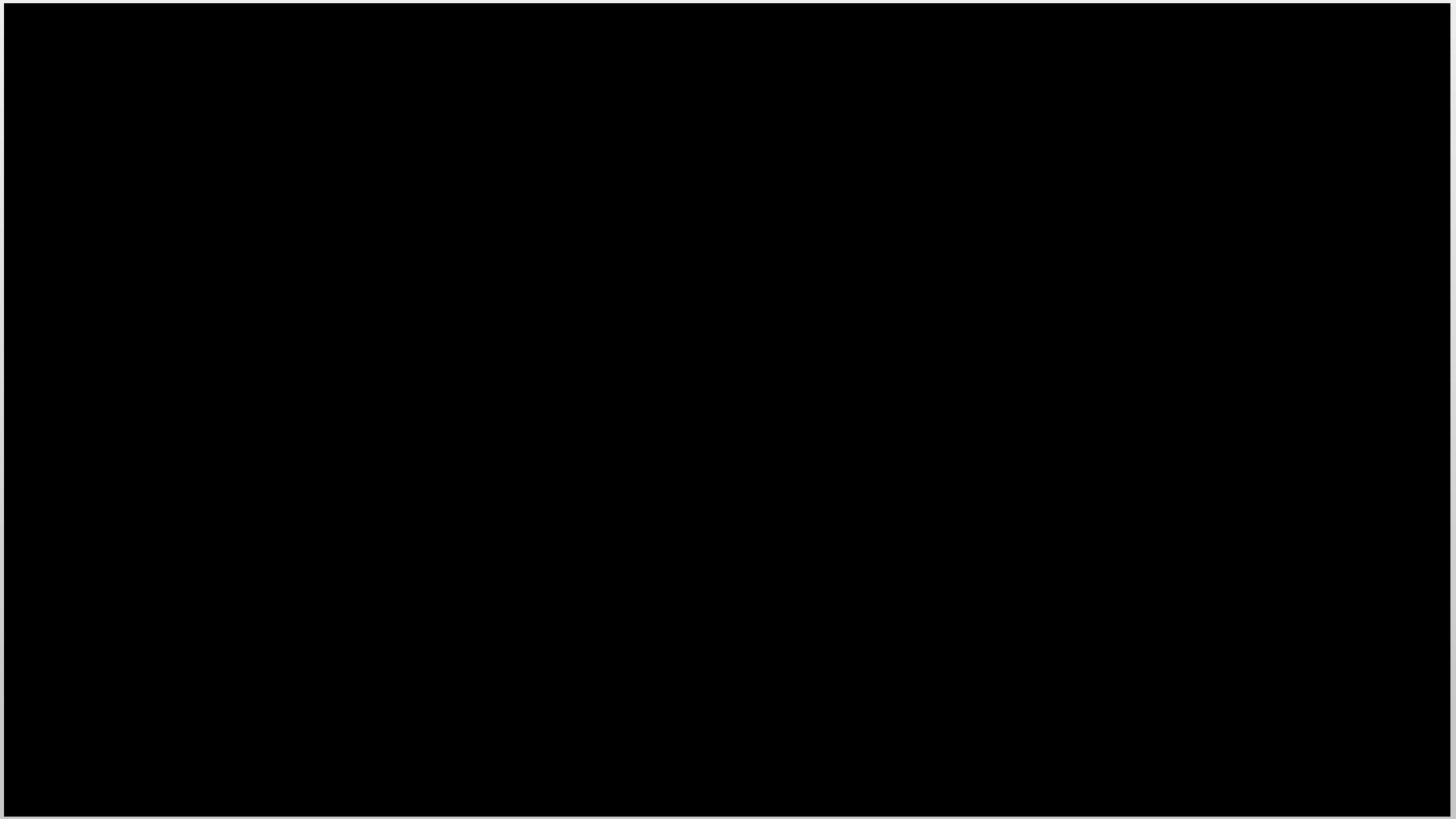


Feature Engineering:

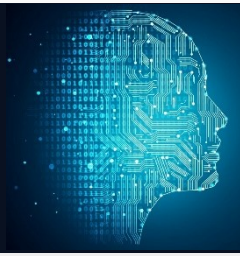


USE CASES: MUSIC CLASSIFICATION

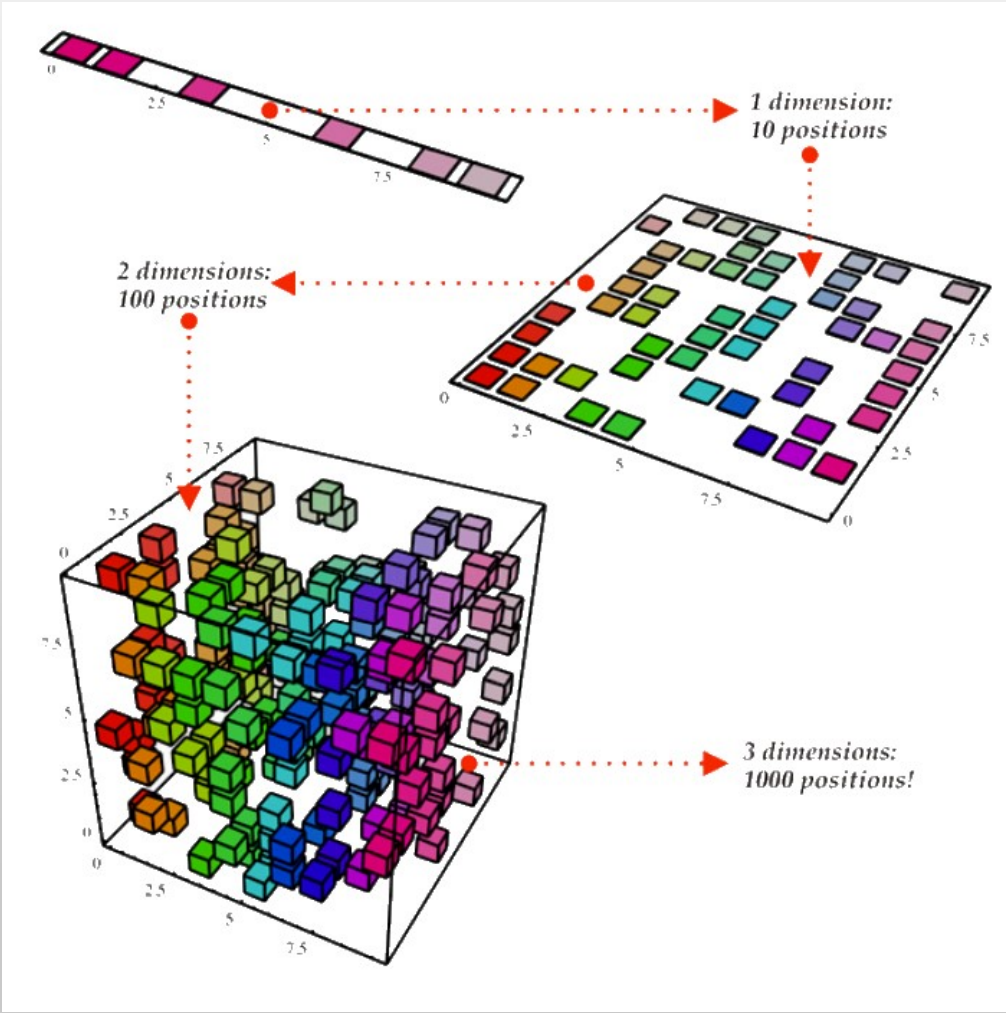




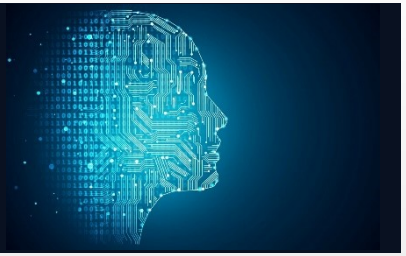
EMBEDDING SPACE

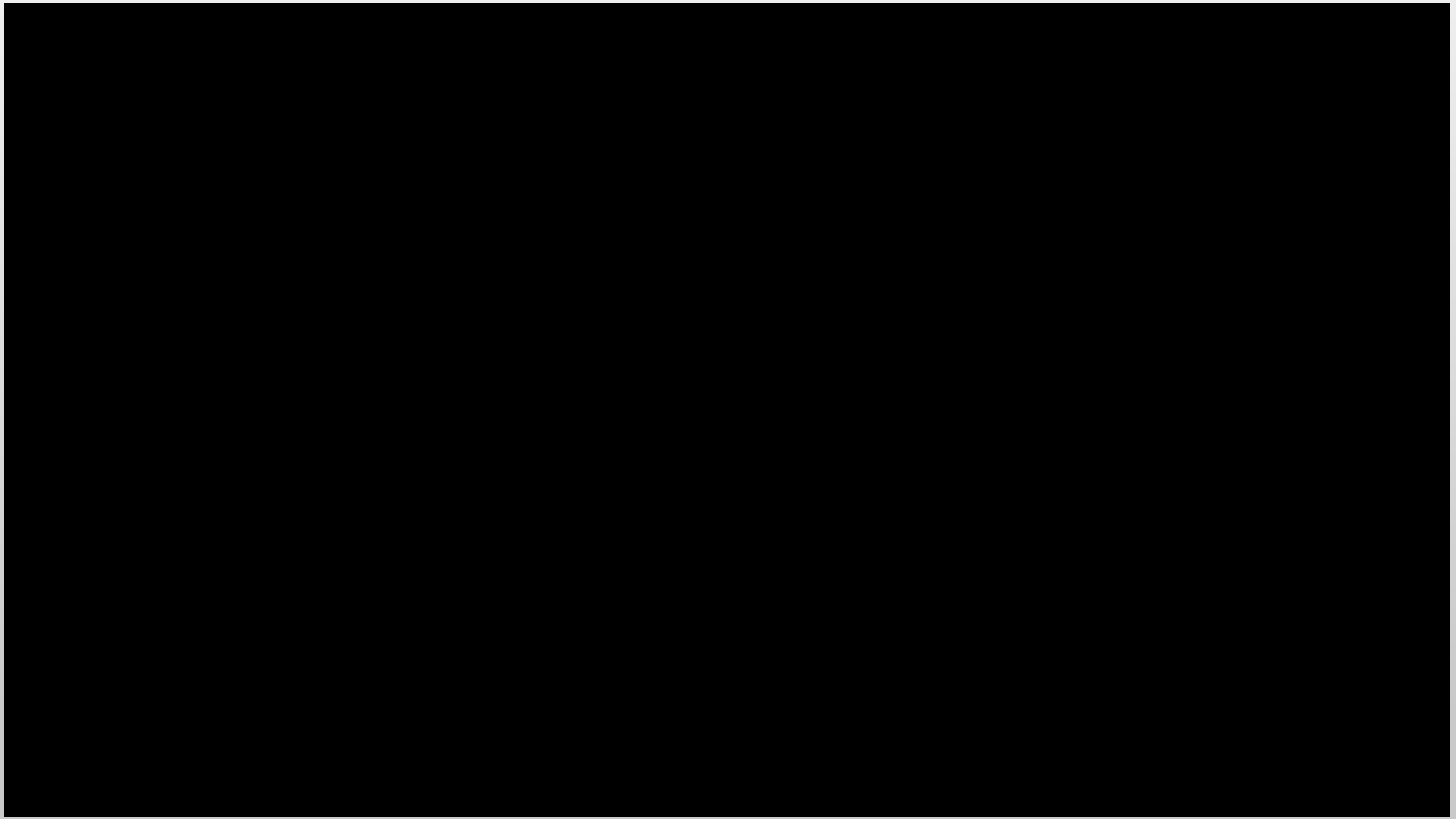


- Dimensionality Reduction
- Manifold Embedding
- Feature Extraction

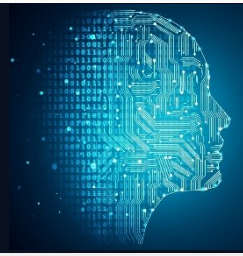


USE CASES: DATA VISUALIZATION





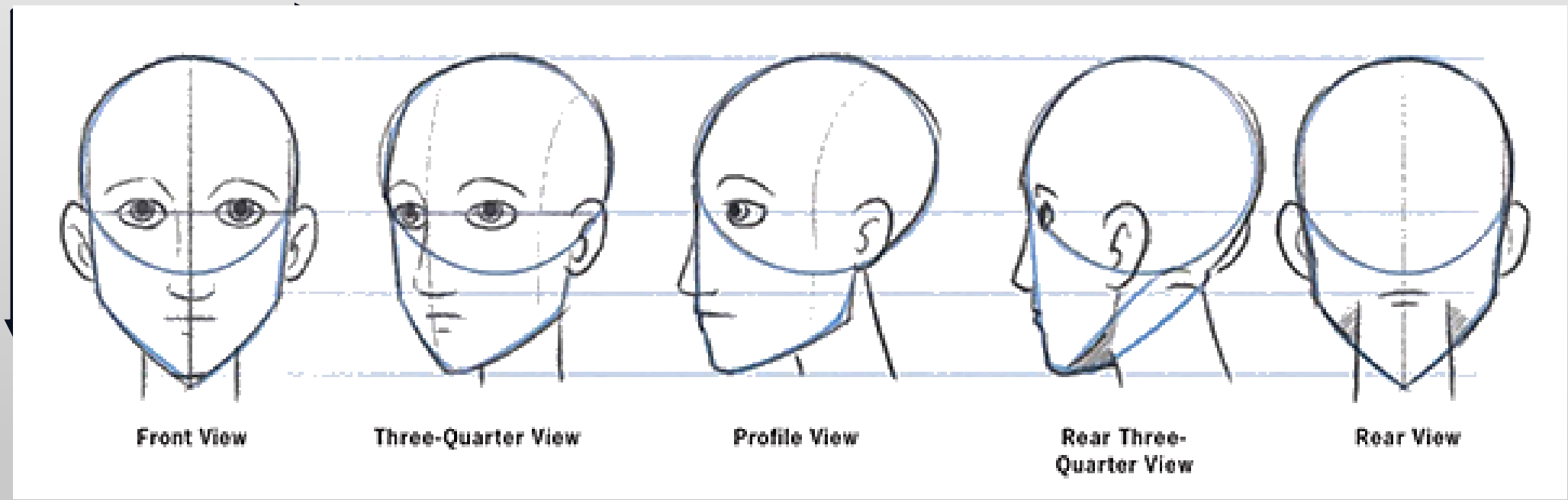
EMBEDDING SPACE



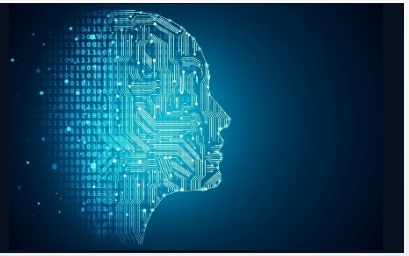
What is the dimension of the problem? (series of images)

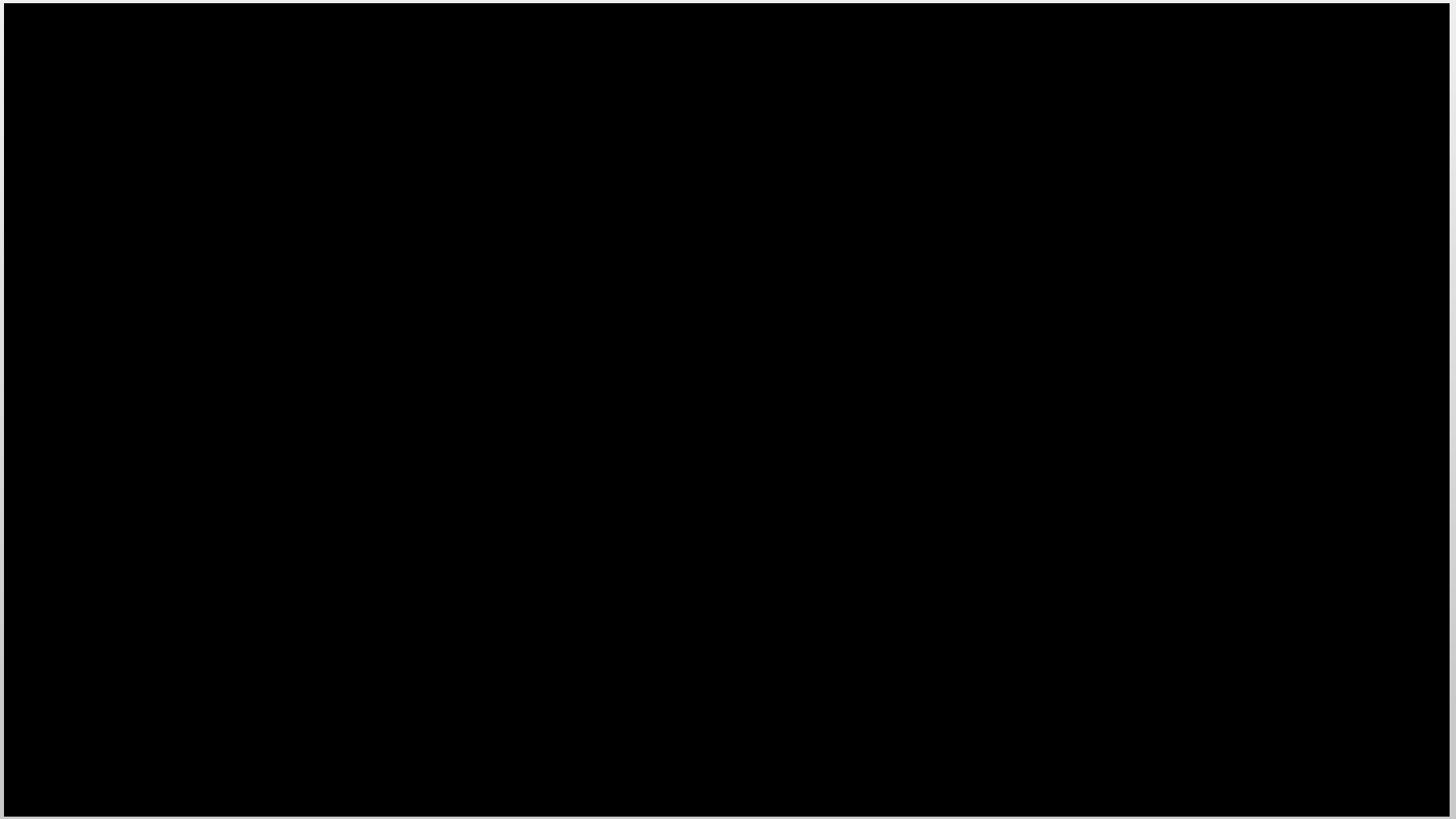
100 pixels

100 pixels

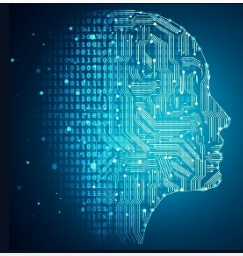


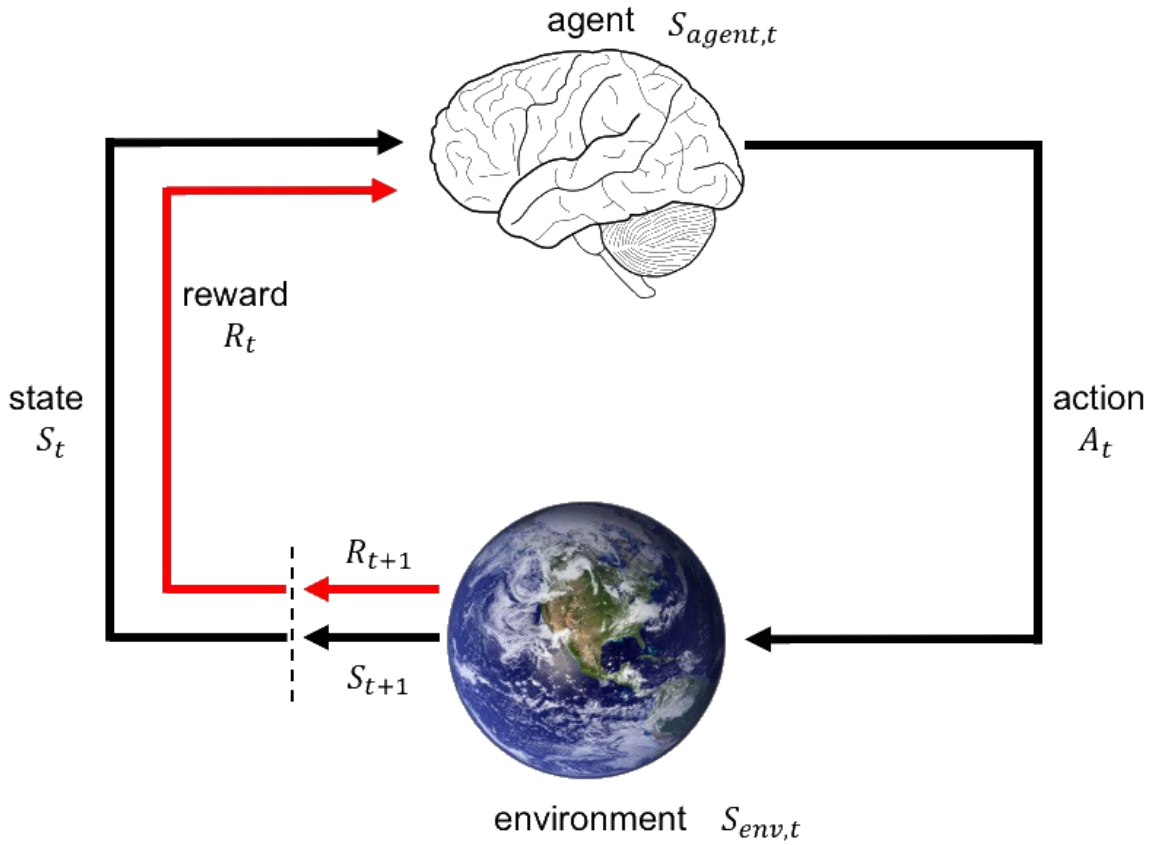
USE CASES: INTELLIGENT AGENT



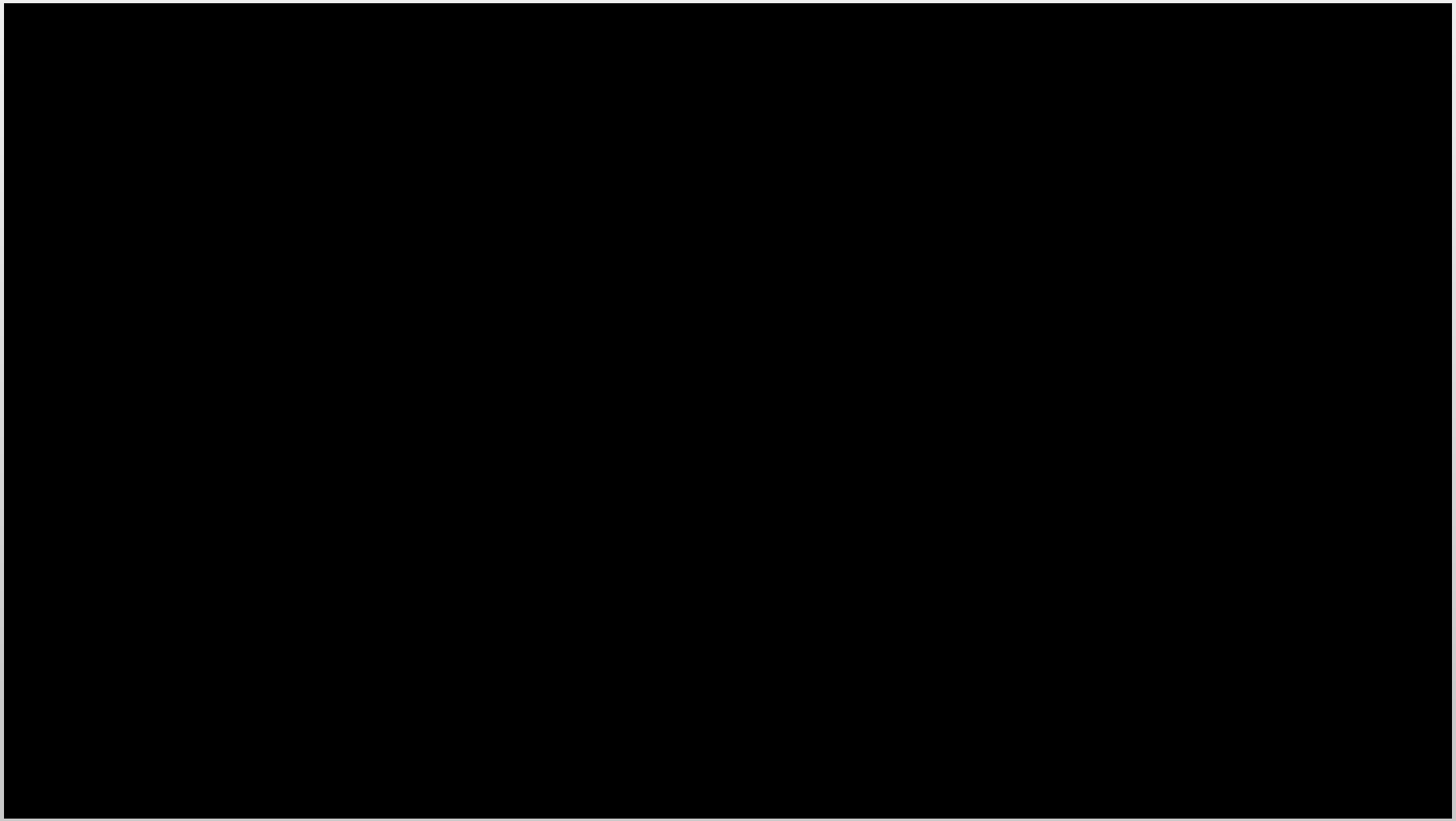


EMBEDDING SPACE - DECISIONS IN REAL TIME!



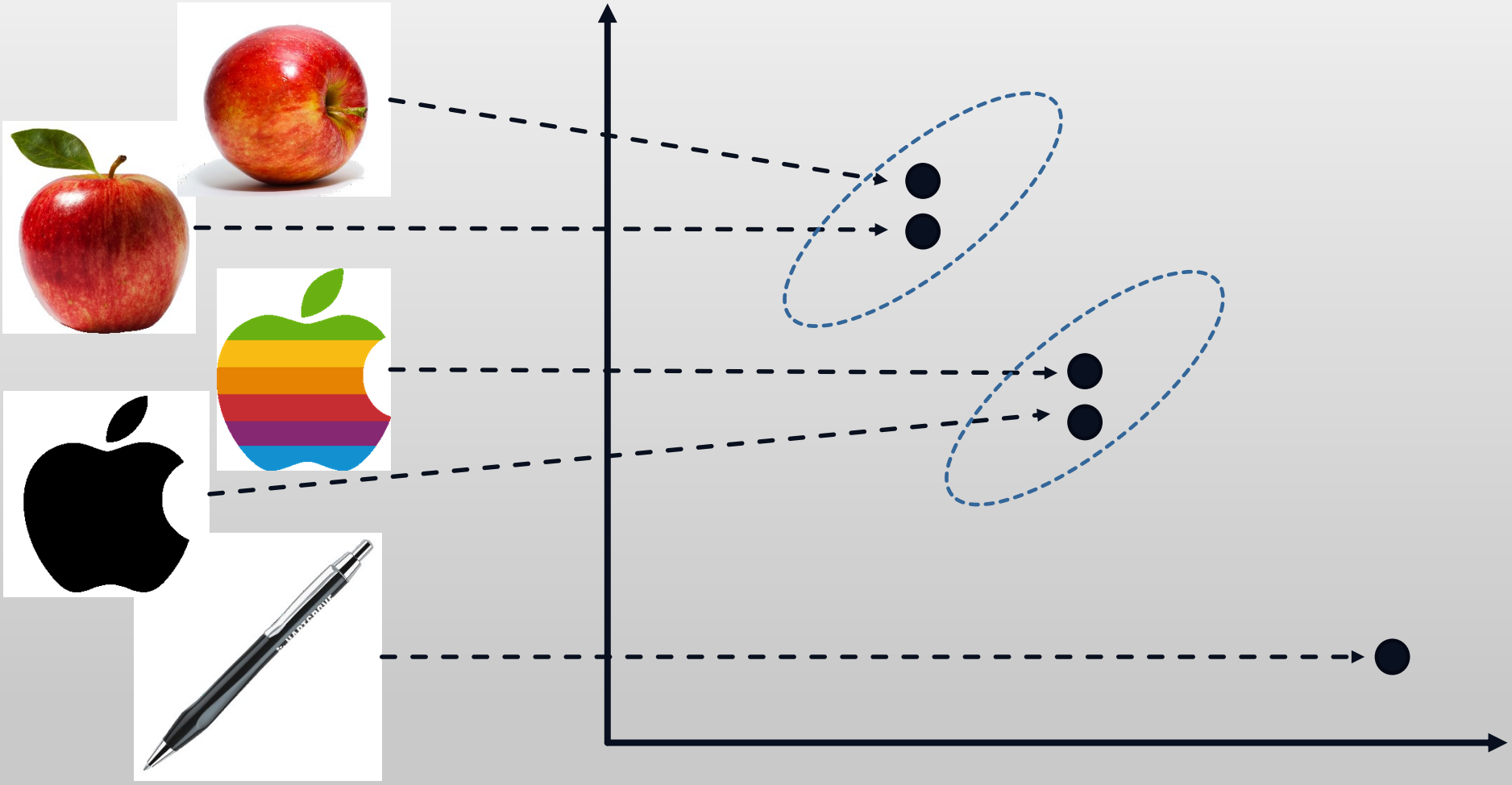


USE CASES: AUTONOMOUS VEHICLE - DARPA CHALLENGE

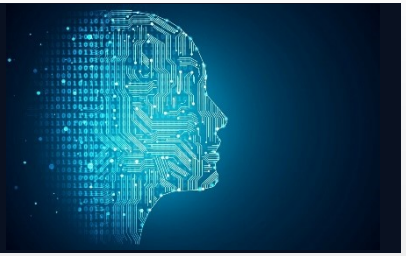


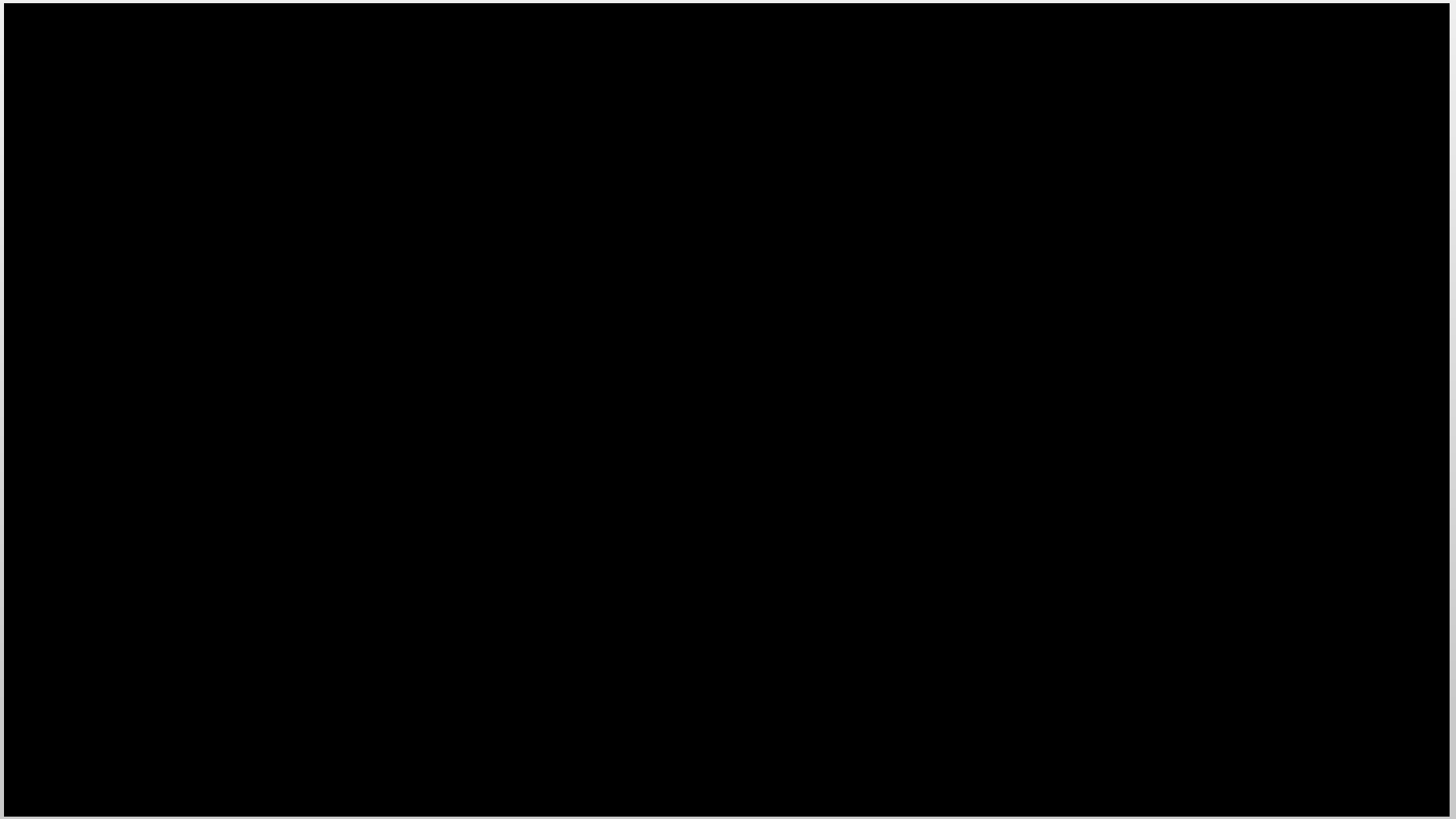
USE CASES: AUTONOMOUS VEHICLE

SIMILARITY AND EMBEDDING ARE IMPORTANT

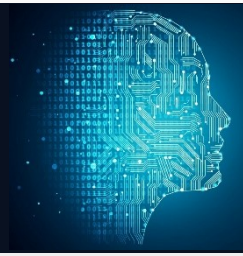


USE CASES: ART GENERATION





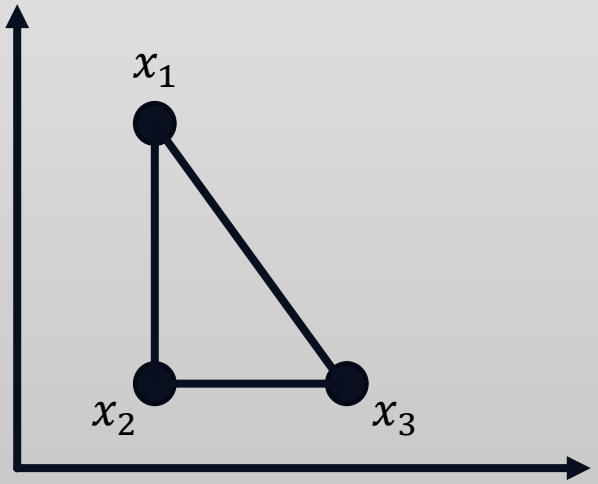
STATISTICS 101 – METRIC SPACES

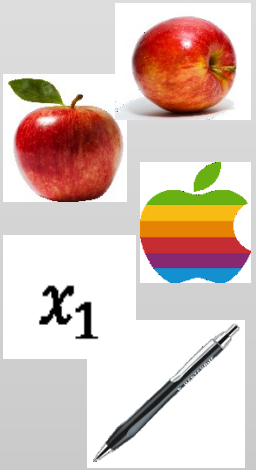


A metric space (X, d) is a set for which distances between all members of the set are defined.

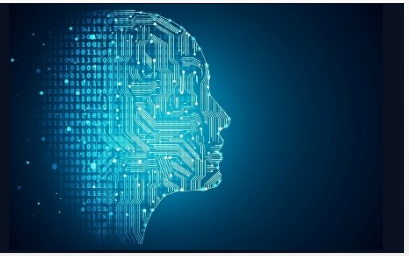
d is a function such that for any $x_1, x_2, x_3 \in X$ the following holds:

1. $d(x_1, x_2) \geq 0$ (non-negativity)
2. $d(x_1, x_2) = 0 \leftrightarrow x_1 = x_2$ (discernable)
3. $d(x_1, x_2) = d(x_2, x_1)$ (symmetric)
4. $d(x_1, x_3) \leq d(x_1, x_2) + d(x_2, x_3)$





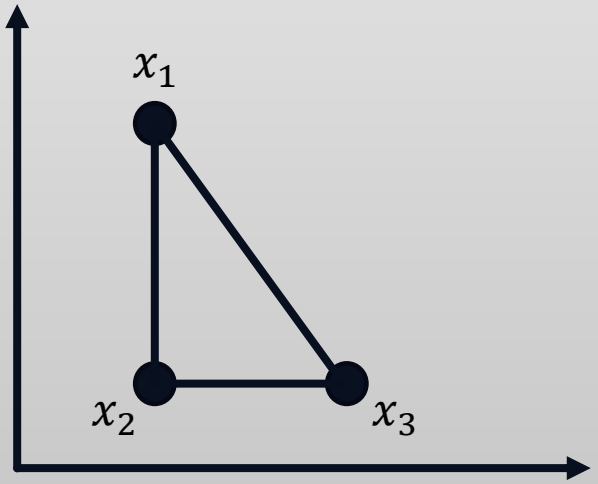
STATISTICS 101 – METRIC SPACES



A pseudo metric space (X, d) does not require (2).

1. $d(x_1, x_2) \geq 0$ (non-negativity)
2. ~~$d(x_1, x_2) = 0 \leftrightarrow x_1 = x_2$ (discernable)~~
3. $d(x_1, x_2) = d(x_2, x_1)$ (symmetric)
4. $d(x_1, x_3) \leq d(x_1, x_2) + d(x_2, x_3)$

$\rightarrow d(x_1, x_2) = 0 \; for \; x_1 \neq x_2$

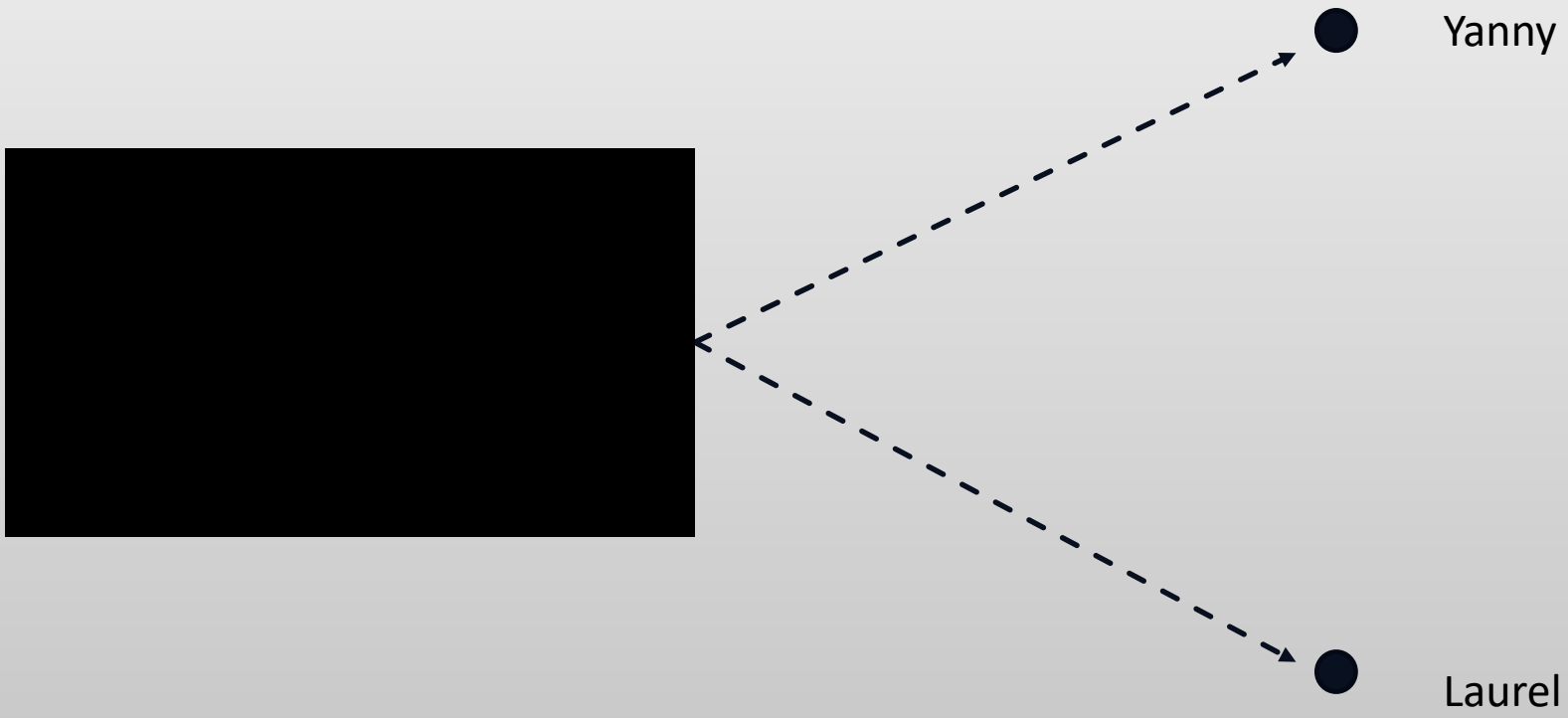


STATISTICS 101 – METRIC SPACES

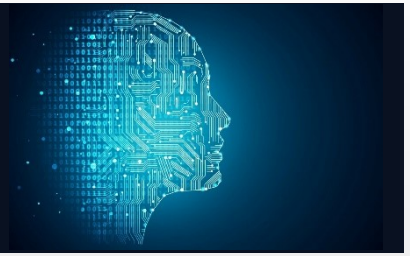
White and Gold

Blue and Black

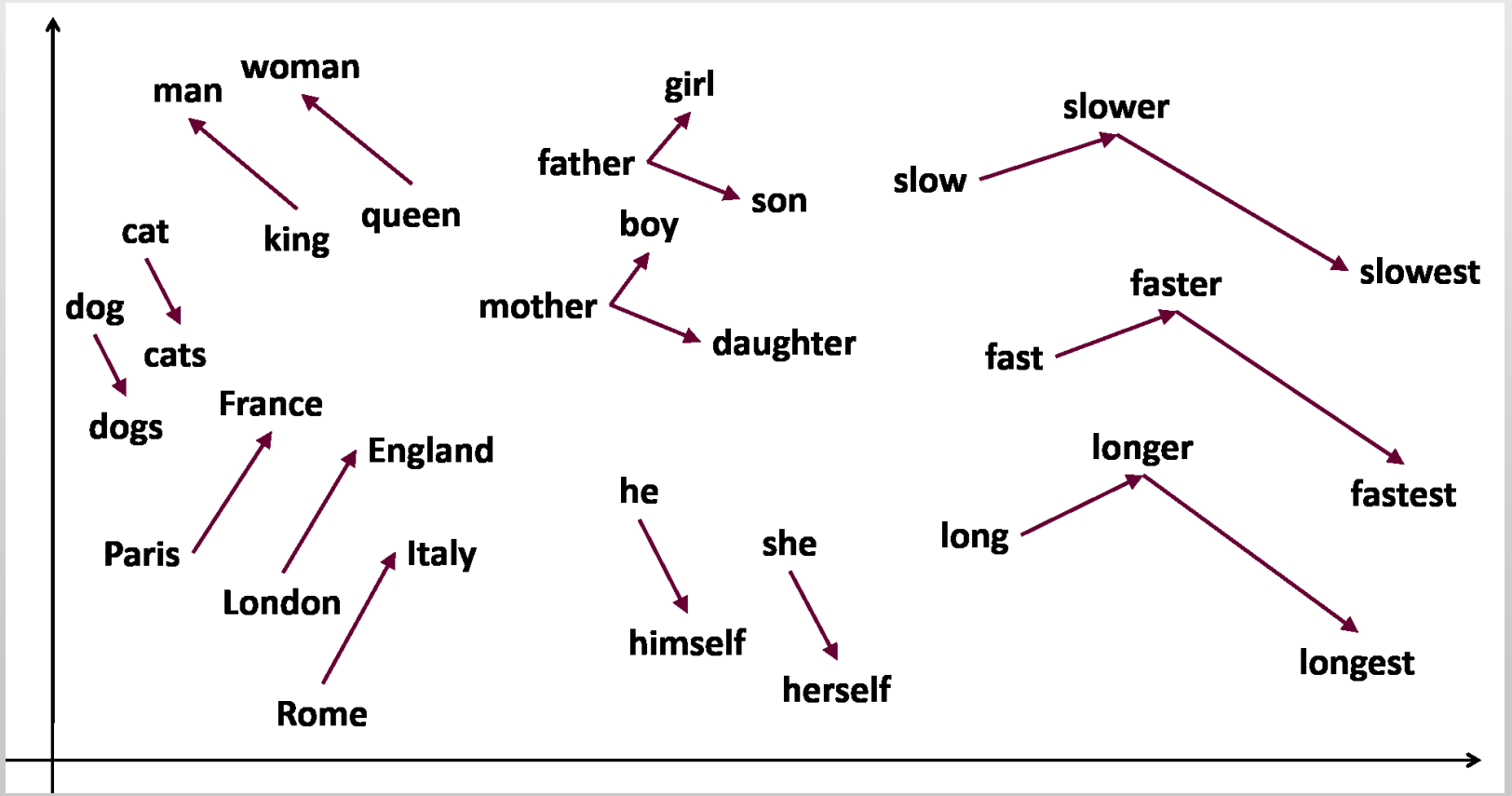
STATISTICS 101 – METRIC SPACES



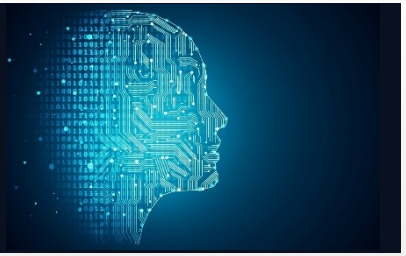
SIMILARITY AND EMBEDDING ARE IMPORTANT



Embedding space in natural language processing (NLP)



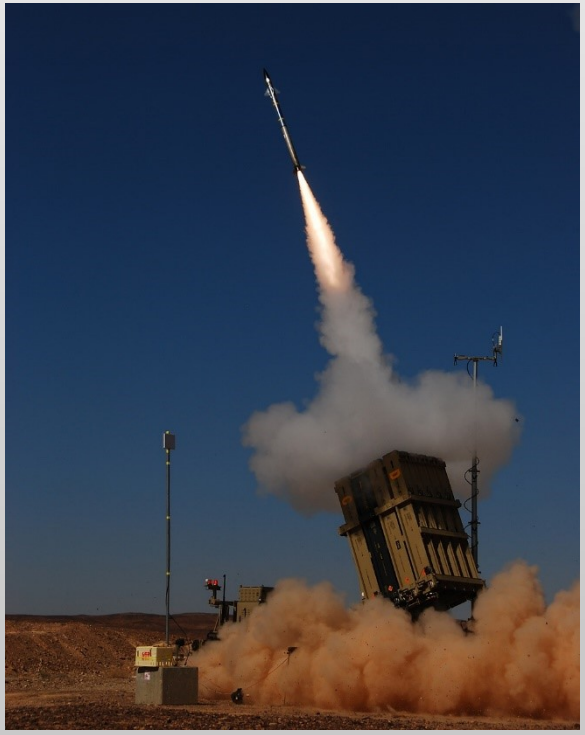
MACHINE LEARING INTRO



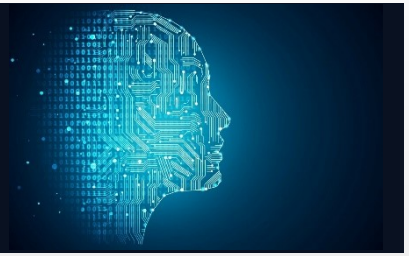
IphoneX Face ID

Mobileye

Iron Dome



STATISTICS 101 – SPECTRAL THEOREM



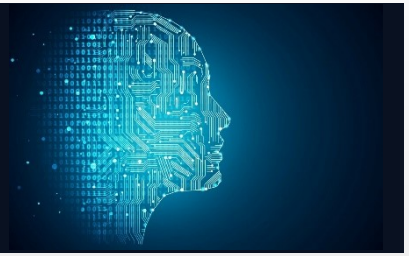
In linear algebra, one is often interested in the **canonical forms** of a linear transformation. Given a particularly nice basis for the **vector space** in which one is working, the matrix of a linear transformation may also be particularly nice, revealing some information about how the transformation operates on the vector space.

Note: A vector space is similar to a metric space, addition and multiplication are defined (instead of distance).

A matrix X with entries $x \in R$ is called symmetric if $X^* = X$. If X is symmetric then the following canonical form holds: $X = U^* D U$ with D a diagonal matrix of the form $D = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{pmatrix}$

The **spectral theorem** states that any symmetric matrix is diagonalizable.

STATISTICS 101 – SVD AND PCA



SVD – Singular Value Decomposition goes into details where and how the spectral theorem holds. In particular, it tells us how to compute $X = U^*DU$.

Example:

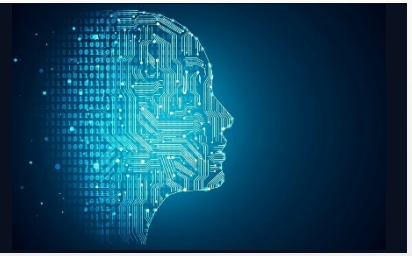
$X = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{bmatrix}$. In order to compute U^*, we compute $X * X^* = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{bmatrix} * \begin{bmatrix} 3 & 2 \\ 2 & 3 \\ 2 & -2 \end{bmatrix} = \begin{bmatrix} 17 & 8 \\ 8 & 17 \end{bmatrix} = W$

$\rightarrow Wx = \alpha x \rightarrow (W - I\alpha)x = 0 \rightarrow det(W - I\alpha) = 0$

$\rightarrow \begin{vmatrix} 17 - \alpha & 8 \\ 8 & 17 - \alpha \end{vmatrix} = 17^2 - 17\alpha - 17\alpha + \alpha^2 - 8^2 = 225 - 34\alpha + \alpha^2 = 0$

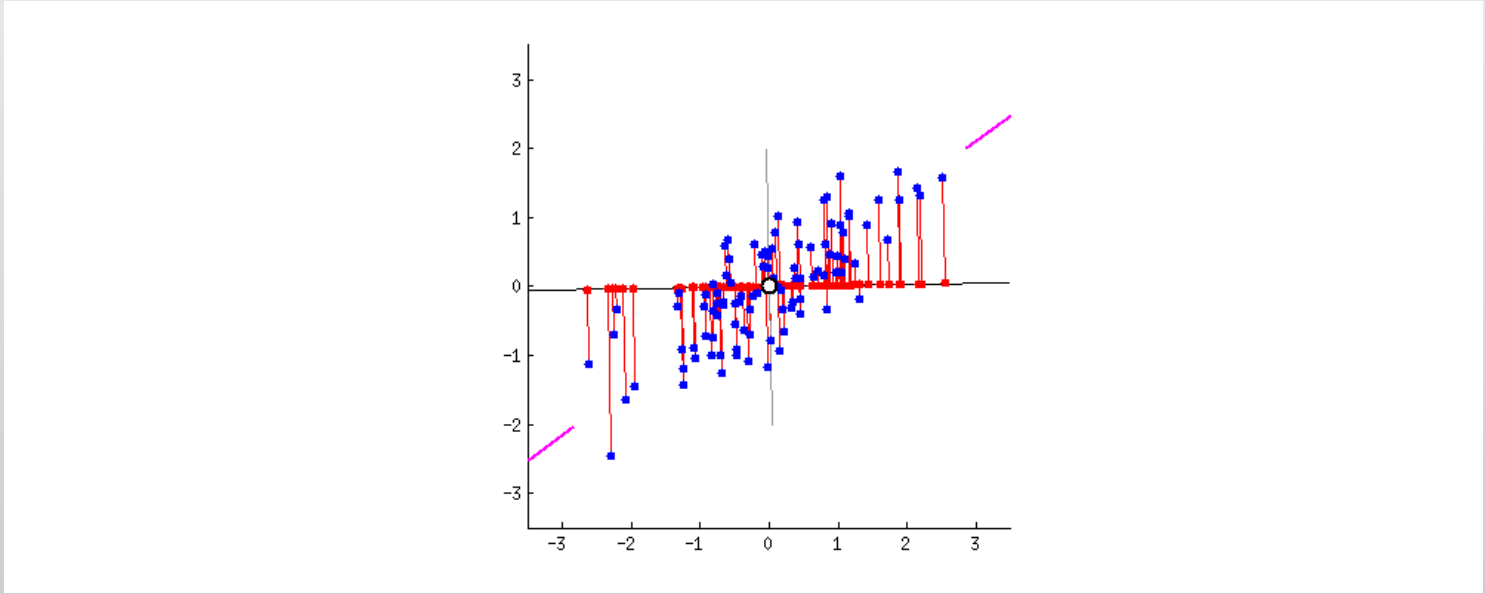
$\rightarrow \alpha = 25{,}9 \rightarrow \sigma_1 = 5, \sigma_2 = 3$

STATISTICS 101 – SVD AND PCA



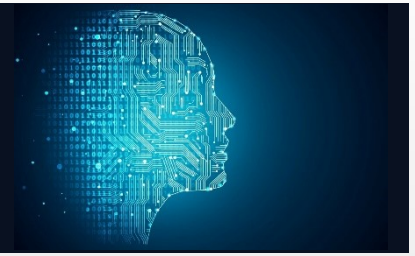
PCA – The SVD of the correlation matrix $Corr = E[(X - \mu_x) \cdot (X - \mu_x)]$ (this is one way to compute PCA, it is numerically stable)

Example 1:



Example 2: link

STATISTICS 101 –PCA FEATURE SELECTION



PCA – The SVD of the correlation matrix $Corr = E[(X - \mu_x) \cdot (X - \mu_x)]$ (this is one way to compute PCA, it is numerically stable)

$X = U^* D U$

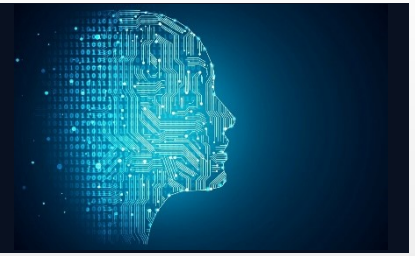
We can relate the diagonal elements of $D = \begin{bmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \ddots \end{bmatrix}$ as relating to the amount of variance of a certain "feature" or column of X.

Why? (We'll explain with real numbers, 3x3 matrix…)

$$X = U^T \cdot \begin{bmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \ddots \end{bmatrix} \cdot U = \begin{bmatrix} u_1^1 & u_2^1 & u_3^1 \\ u_1^2 & u_2^2 & u_3^2 \\ u_1^3 & u_2^3 & u_3^3 \end{bmatrix}^T \cdot \begin{bmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{bmatrix} \cdot \begin{bmatrix} u_1^1 & u_2^1 & u_3^1 \\ u_1^2 & u_2^2 & u_3^2 \\ u_1^3 & u_2^3 & u_3^3 \end{bmatrix} = \begin{bmatrix} u_1^1 & u_1^2 & u_1^3 \\ u_2^1 & u_2^2 & u_2^3 \\ u_3^1 & u_3^2 & u_3^3 \end{bmatrix} \cdot \begin{bmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{bmatrix} \cdot \begin{bmatrix} u_1^1 & u_2^1 & u_3^1 \\ u_1^2 & u_2^2 & u_3^2 \\ u_1^3 & u_2^3 & u_3^3 \end{bmatrix} =$$

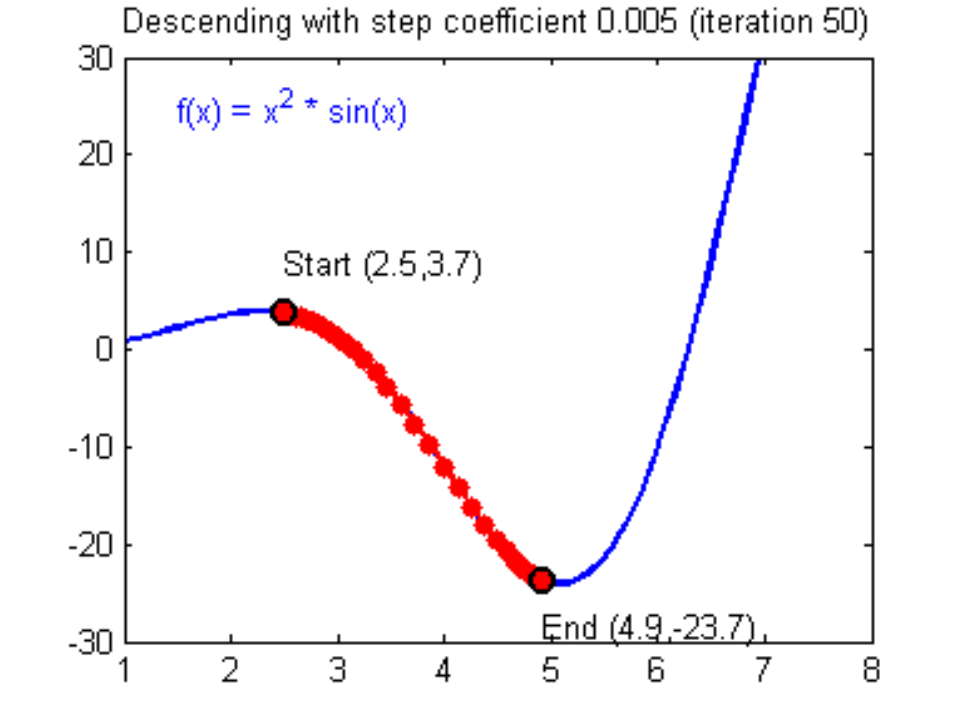
$$\begin{bmatrix} u_1^1 \cdot \alpha_1 & u_1^2 \cdot \alpha_2 & u_1^3 \cdot \alpha_3 \\ \vdots & \ddots & \vdots \\ \vdots & \cdots & \ddots \end{bmatrix} \cdot \begin{bmatrix} u_1^1 & u_2^1 & u_3^1 \\ u_1^2 & u_2^2 & u_3^2 \\ u_1^3 & u_2^3 & u_3^3 \end{bmatrix} = \begin{bmatrix} u_1^1 \cdot \alpha_1 \cdot u_1^1 + u_1^2 \cdot \alpha_2 \cdot u_1^2 + u_1^3 \cdot \alpha_3 \cdot u_1^3 & \cdots & \cdots \\ \vdots & \ddots & \vdots \\ \vdots & \cdots & \ddots \end{bmatrix}$$

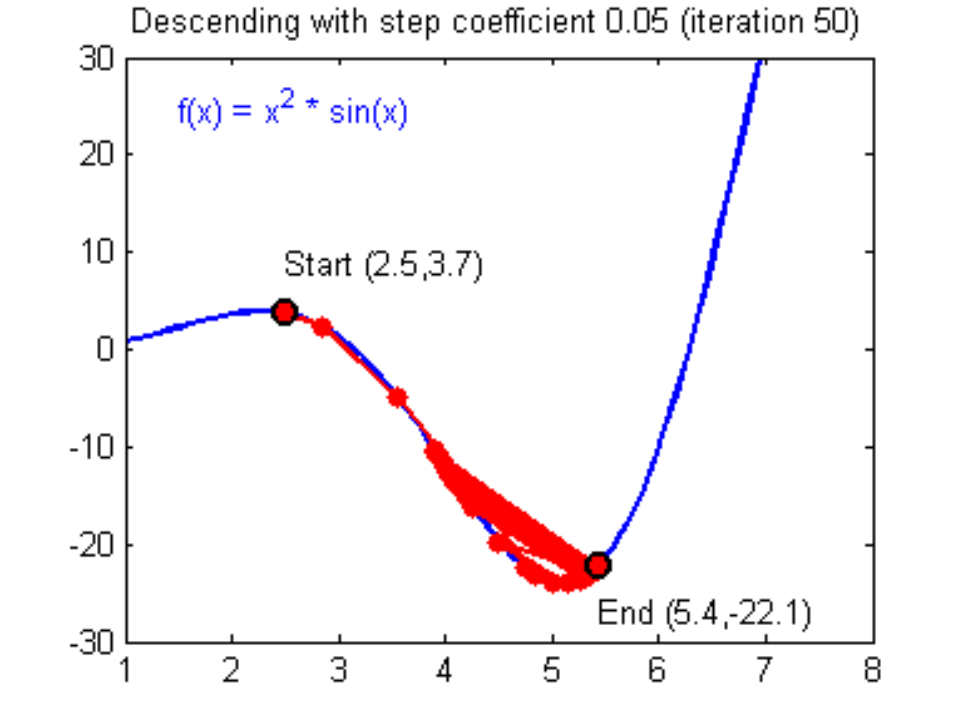
TRAINING A MODEL – OPTIMIZATION



Optimization – An algorithm for selecting the best element from a selection of elements.

Example: Assume the "best" element is the element with the **lowest** value.





EXPLORATORY DATA ANALYSIS

Let’s examine two methods for exploring data.

The first: PCA. The second, TSNE (T-Stochastic Neighborhood Embedding) which uses optimization as a tool.

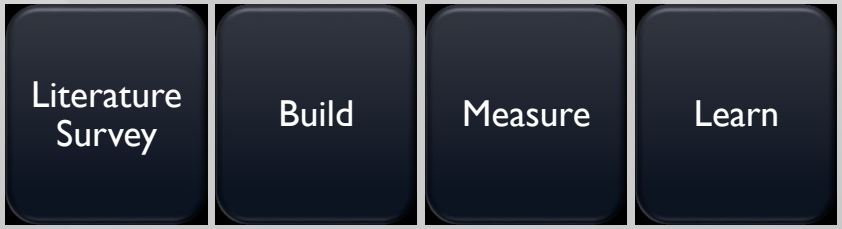
These tools each have their strong and weak points.

Python Example - Visualization

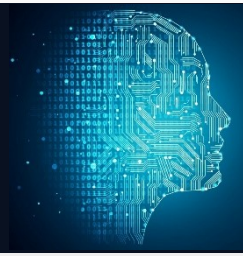
DATA PREPARATION

Each challenge in data science is different. For image recognition we have one set of tools, for recommendation system we have another. If we had one great set of tools for all problems, then we would have true AI (AGI). But we don't (yet).

So how do we prepare our data?

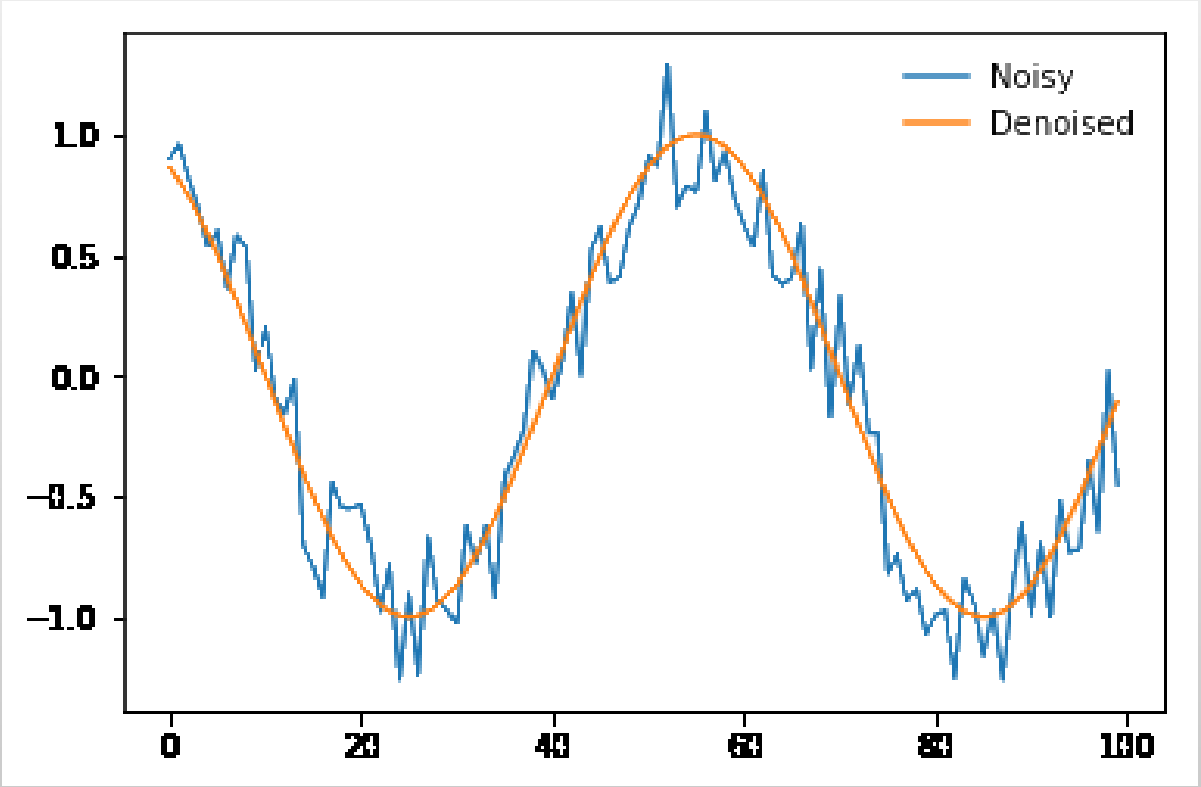


DATA PREPARATION – CLEANING A DATASET

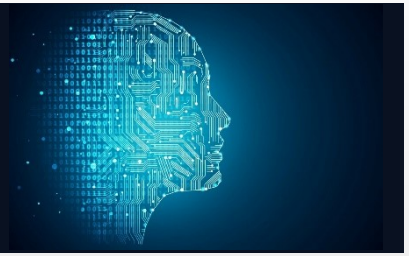


How can we denoise the wine dataset?

Exercise 1 – How can we use PCA for feature selection in practice? What makes a good wine? (-:



DATA PREPARATION – FILTERING AND SCALING



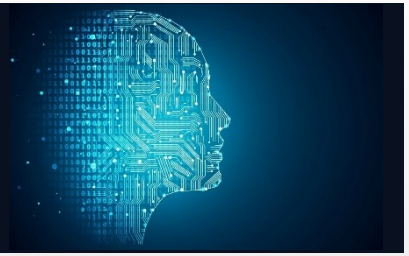
Each “modality” has its own methods. In text retrieval or generally *information retrieval* we have a method called *tf-idf* which we will see in a moment.

How do we filter or scale text???

There are three basic processes an information retrieval system has to support: the representation of documents, the representation of a user request, and the comparison of these two representations.

-Djord Hiemstra

DATA PREPARATION – FILTERING AND SCALING



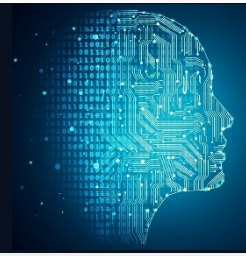
The baseline representation for recommendation systems and text analysis is called the *"bag of words"* model.

Lets take the following document:

The course Data Science and Machine Learning by Leeor is a really great course.

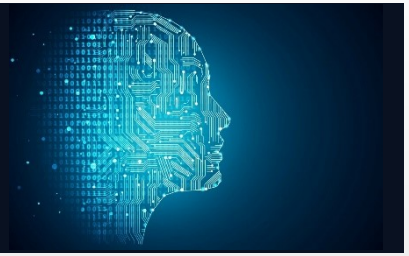
a, 1
and, 1
by, 1
course, 2
data, 1
great, 1
is, 1
learning, 1
machine, 1
really, 1
science,1
the, 1

DATA PREPARATION – FILTERING AND SCALING



Exercise 2 – Create Bag of Words model for IMDB movie reviews

DATA PREPARATION – FILTERING AND SCALING



Tf Idf: Term Frequency Inverse Document Frequency

A transformation that models how **important** a word is.

$$tf - idf(t,d,D) = tf(t,d) \cdot idf(t,D)$$

$$tf(t,d) = \frac{n_d}{n_t}$$

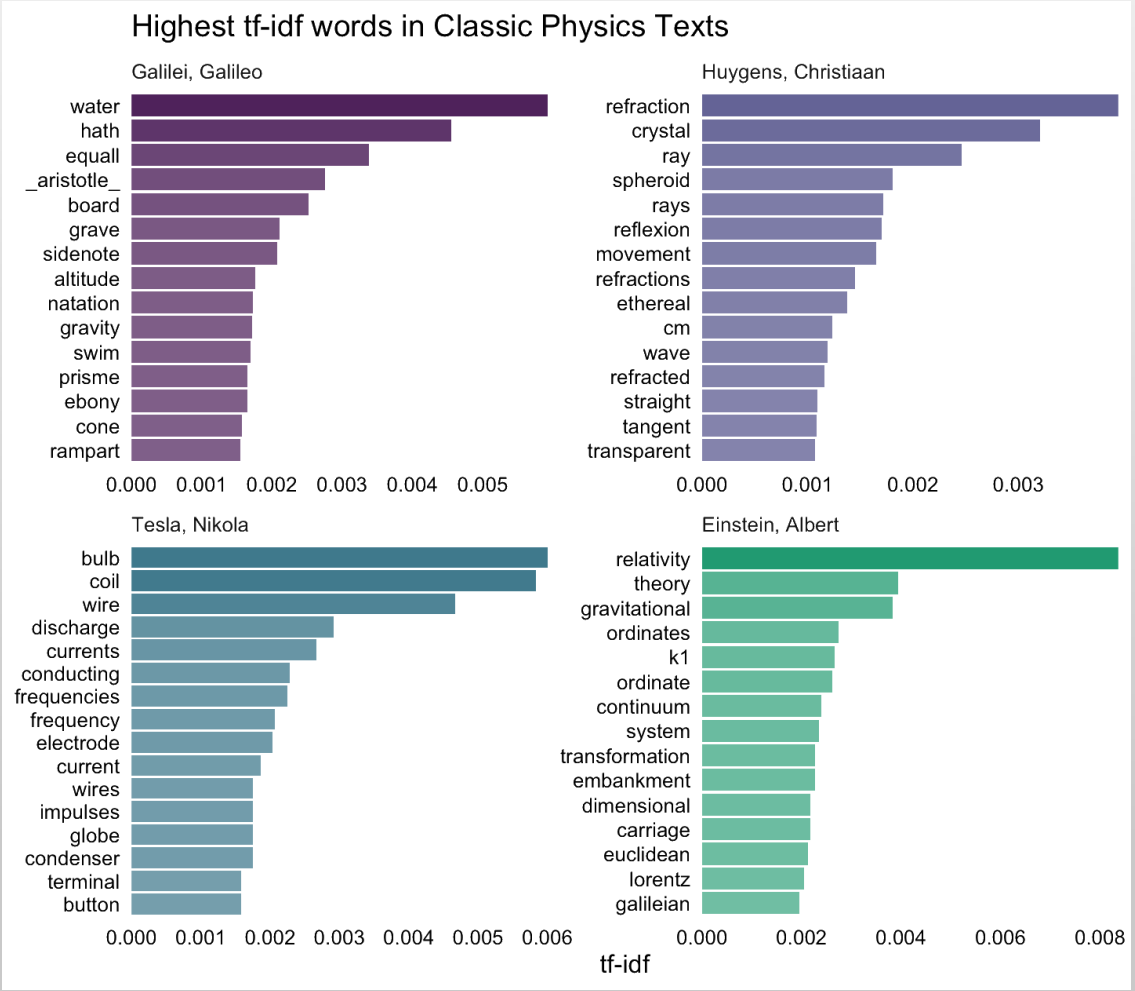
$$idf(t,D) = \log\left(\frac{N_D}{N_t}\right)$$

n_d total number of words in doc d

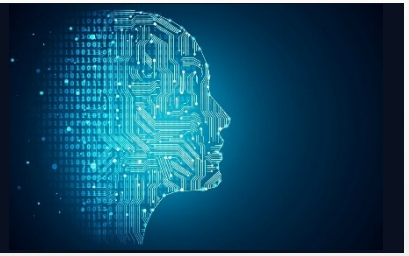
n_t number of occurrences of term t in doc d

N_D total number of words in all docs

N_t total number of occurrences of term t in all docs



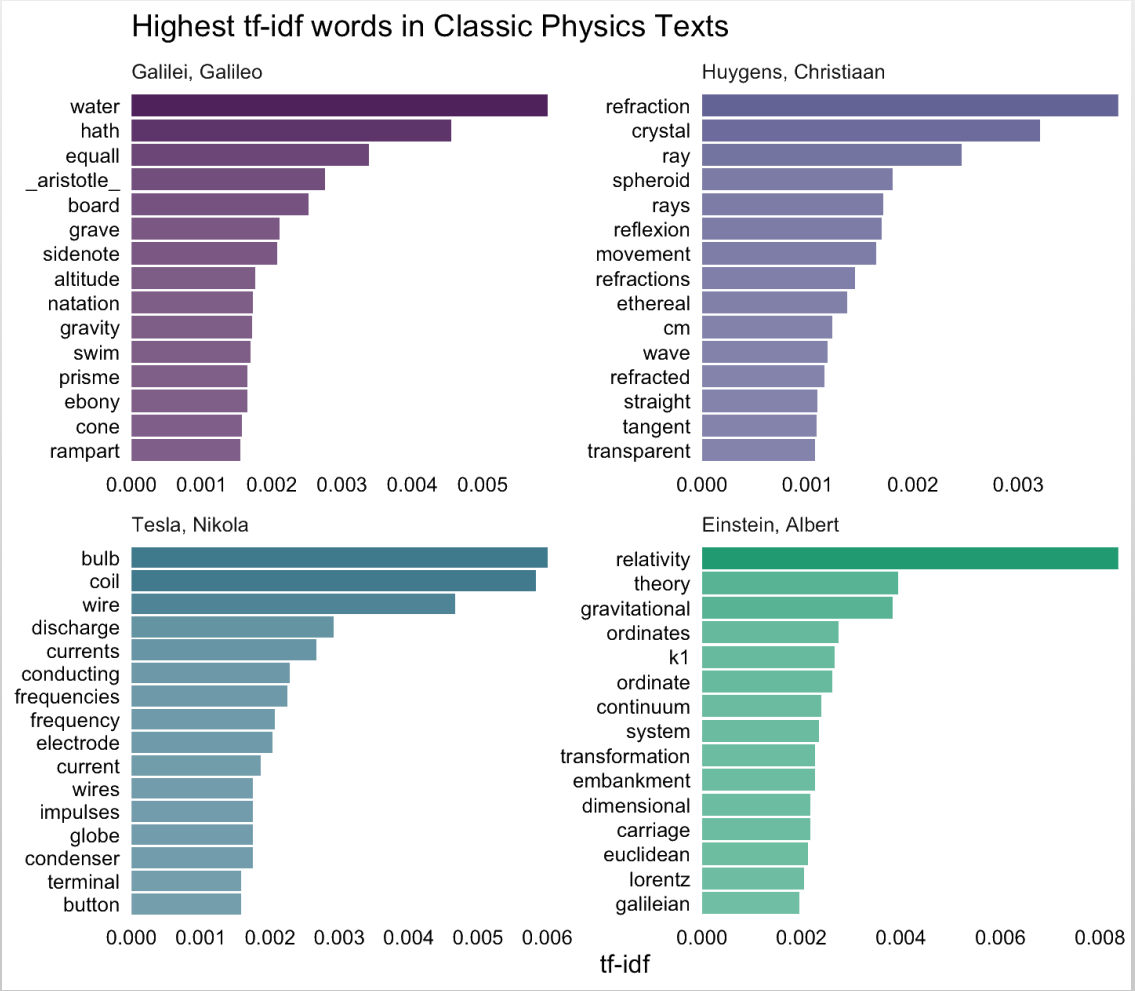
DATA PREPARATION – FILTERING AND SCALING



Exercise 3 – Transform model to tf-idf scores and check out which words make a big difference

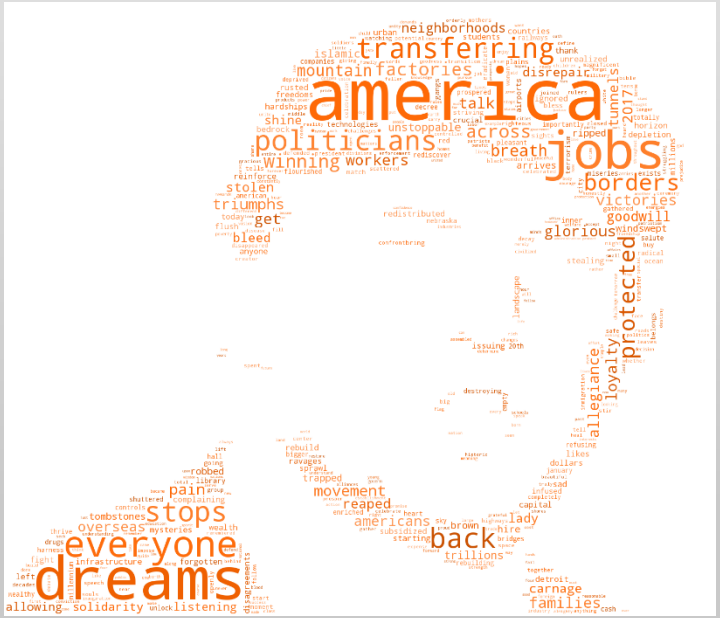
Discuss:

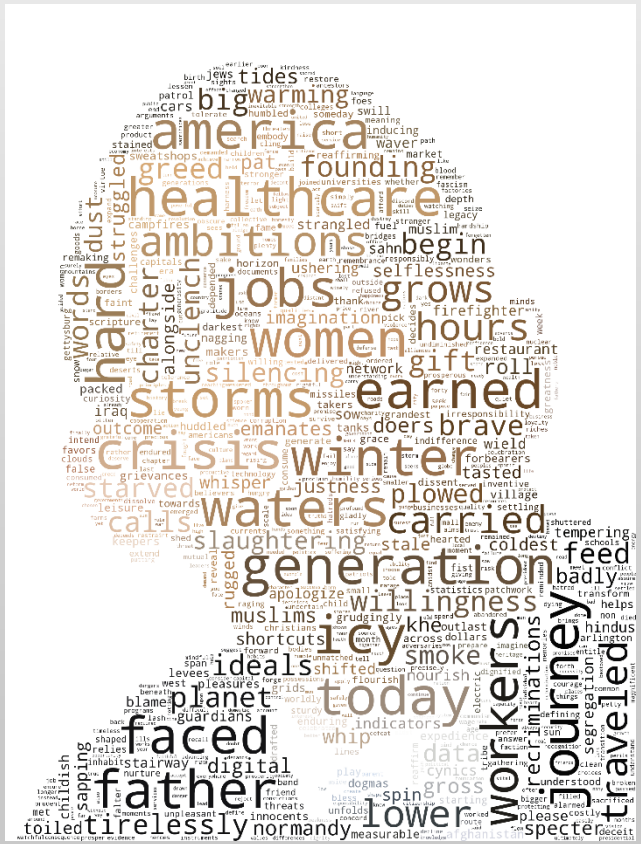
- What is the noise in our results?
- How would we filter such noise?

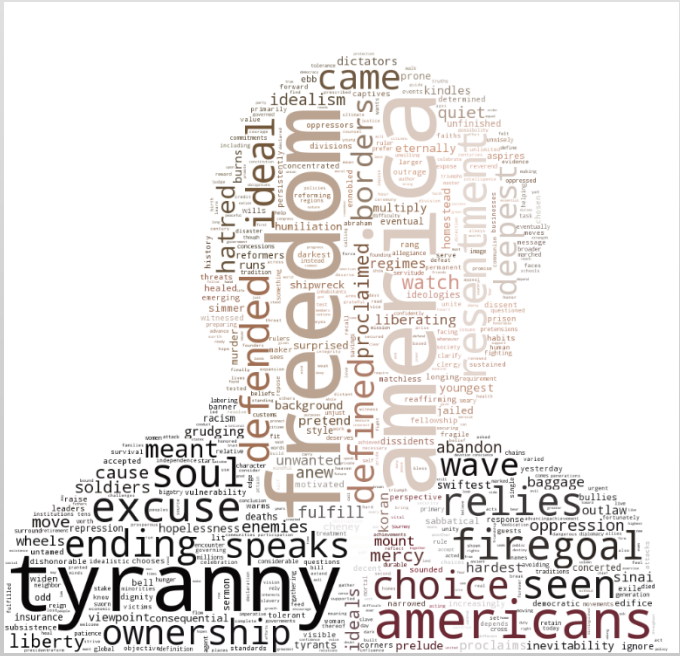


DATA PREPARATION – FILTERING AND SCALING

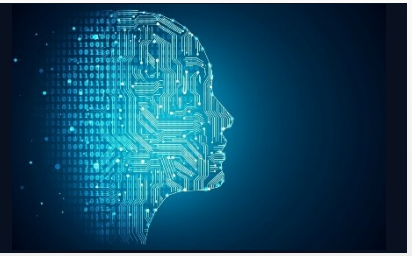
Inauguration word cloud:







OUTLIERS

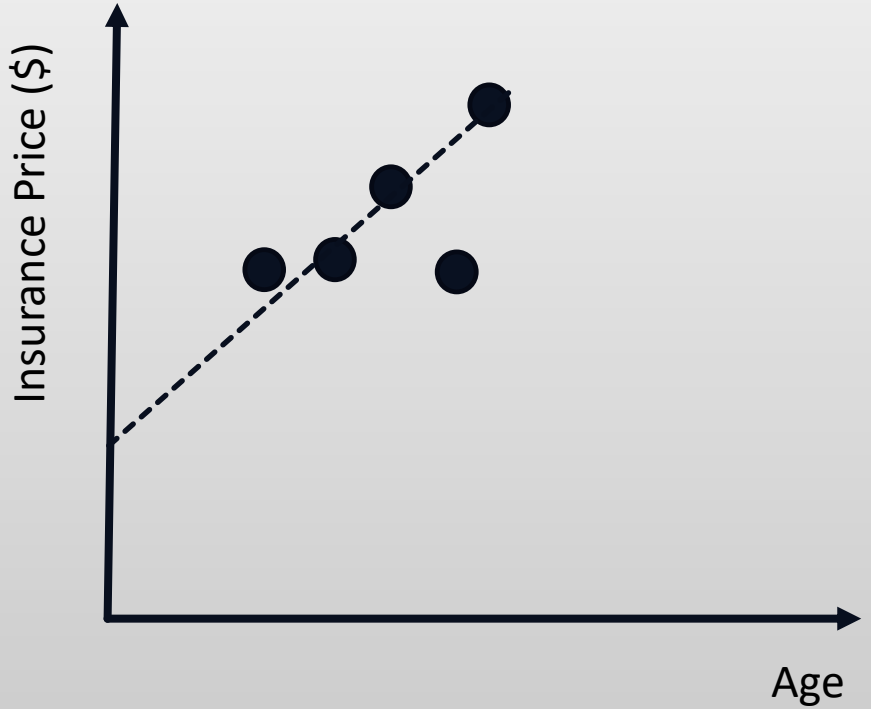


Define a model:

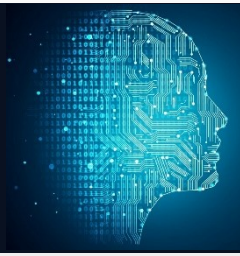
- Linear Regressor
- Outliers are samples that don't fit your model

Evaluate your model:

- There is no mathematical formula for outlier detection, it's an open problem (finding which observations are outliers)
- What makes a good feature on "inauguration campaign" (freedom!) may be a meaningless feature on a different dataset (maybe prison documents? (-:)



DATA PREPARATION – NORMALIZATION



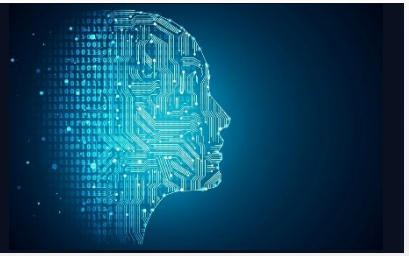
- Why do we normalize our data?

We do this as humans so naturally, for example when we compare elements.

For example, when comparing two peoples height, we immediately notice that a child and an adult are from different scales… Their height comparison is not the same as two adults.

- How do we normalize our data?
  - Z-Score: $z = \frac{x-\mu}{\sigma}$, mean and variance of population
  - Feature Scaling: $x' = \frac{x-x_{min}}{x_{max}-x_{min}}$

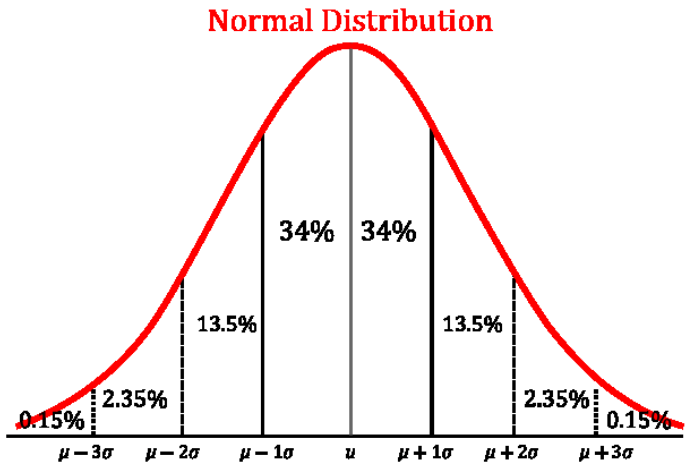
DATA PREPARATION – NORMALIZATION

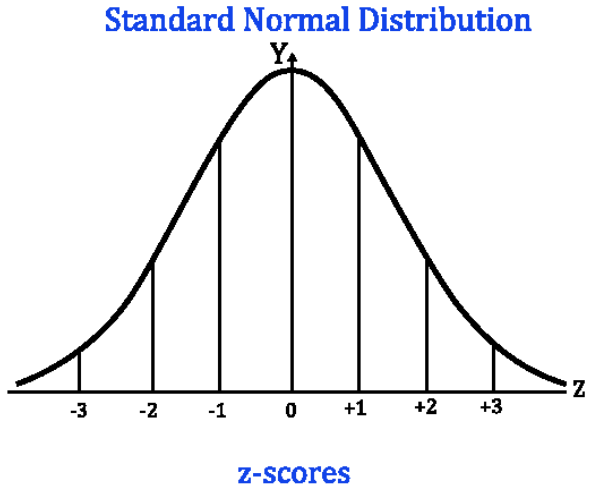


- Why do we normalize our data?
- How do we normalize our data?
  - Z-Score: $z = \frac{x-\mu}{\sigma}$, mean and variance of population
  - Tf-Idf: can also be considered normalization.
  - The score is formally a cross entropy metric:

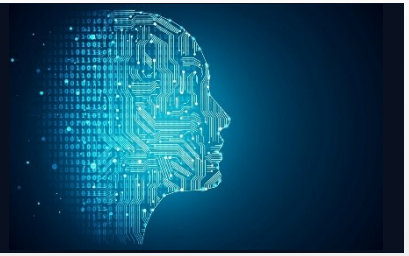
$$H(p,q) = -\sum_{x \in X} p(x) \cdot log(q(x))$$

  - Each problem usually has its own unique normalization method (and other methods…)

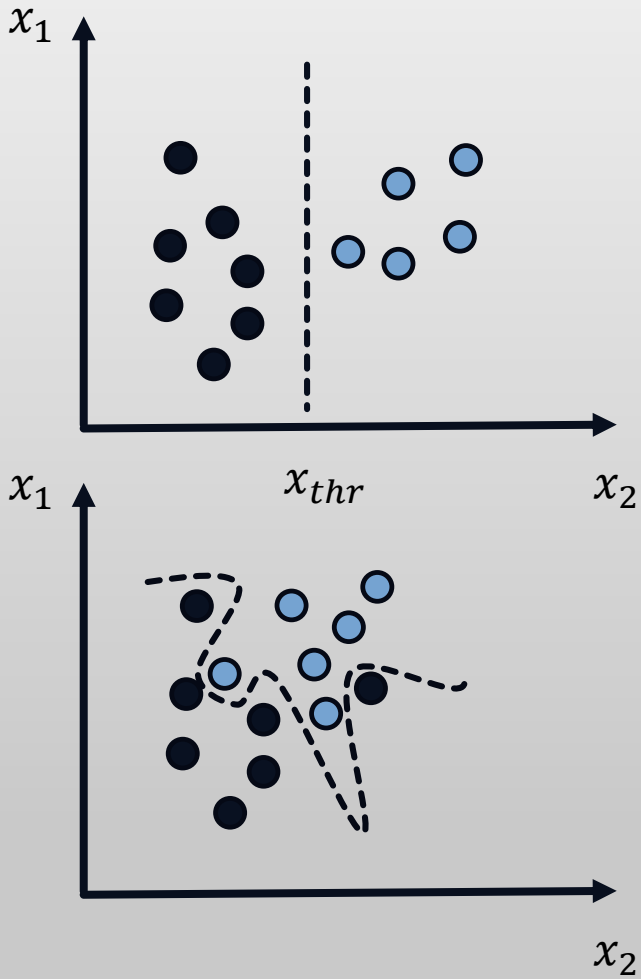




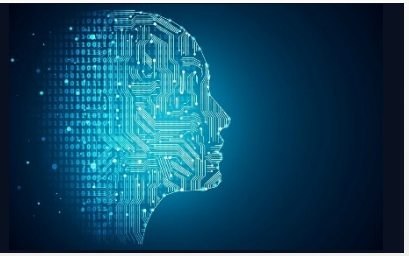
MAKING DECISIONS WITH ALGORITHMS



- Supervised Learning: $\mathrm{D} = \{X, y\}_{i=1}^{N}$
- The first plot has an easy decision formula
- What about the second?

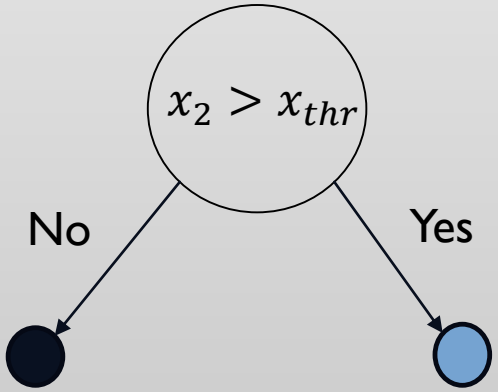


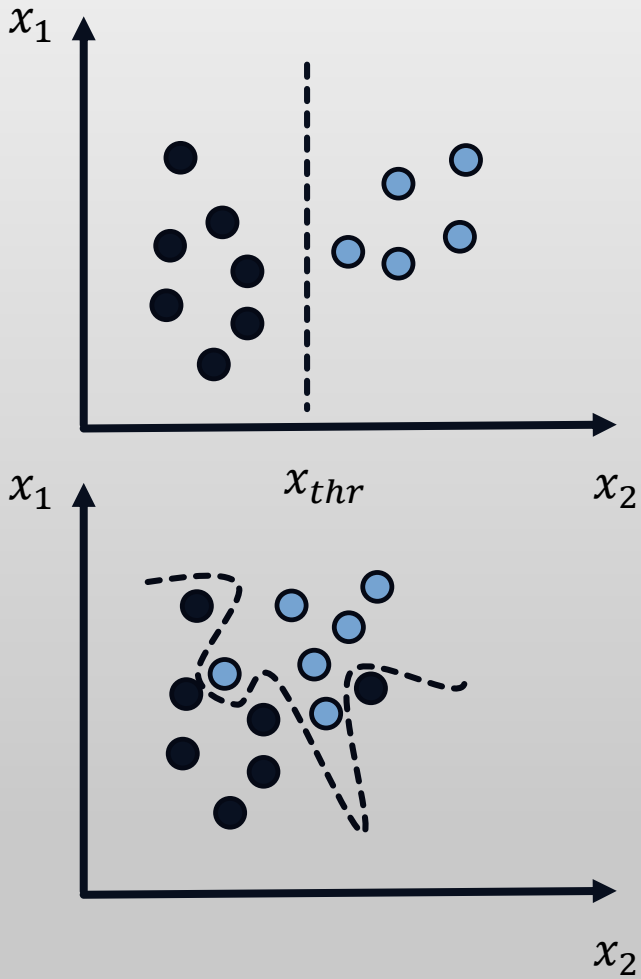
MAKING DECISIONS WITH ALGORITHMS



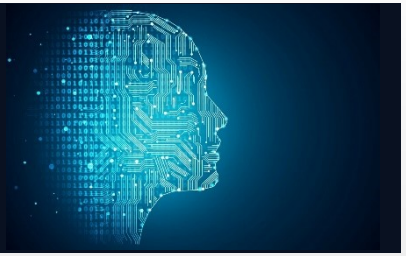
Supervised Learning: $\mathrm{D} = \{X, y\}_{i=1}^{N}$

Decision Stump:

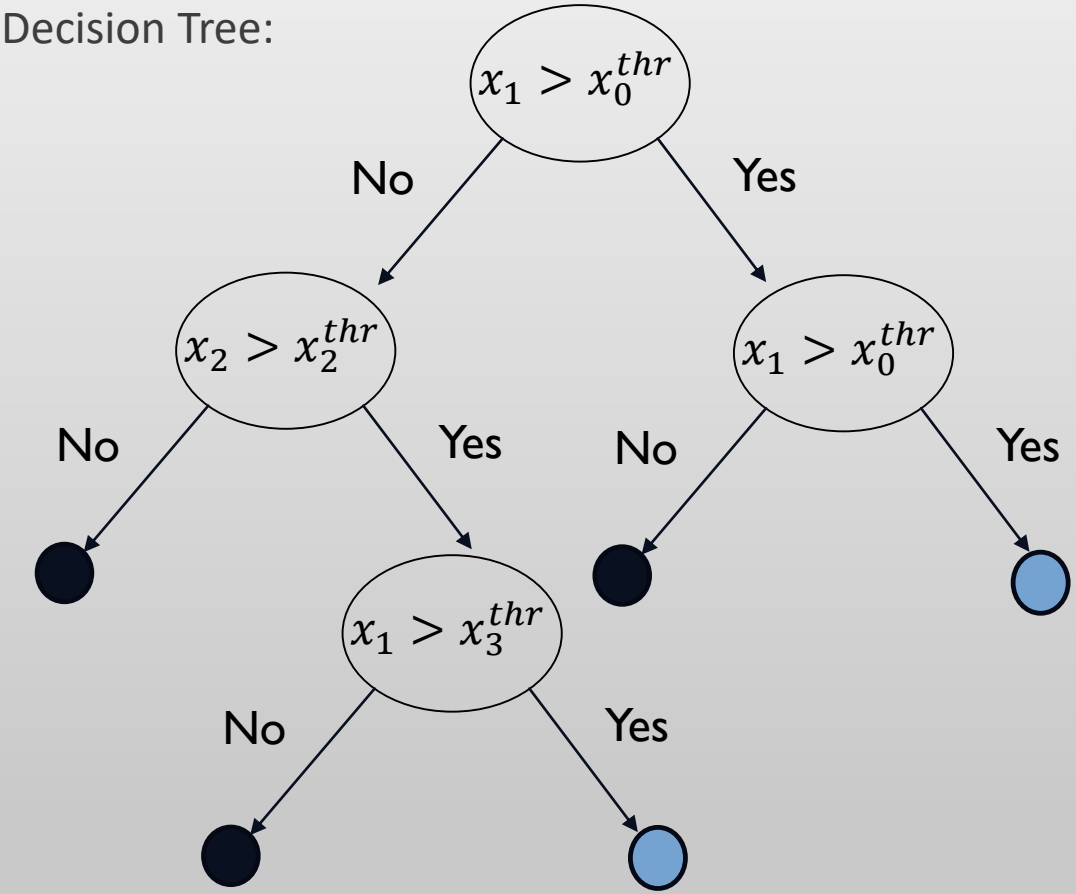


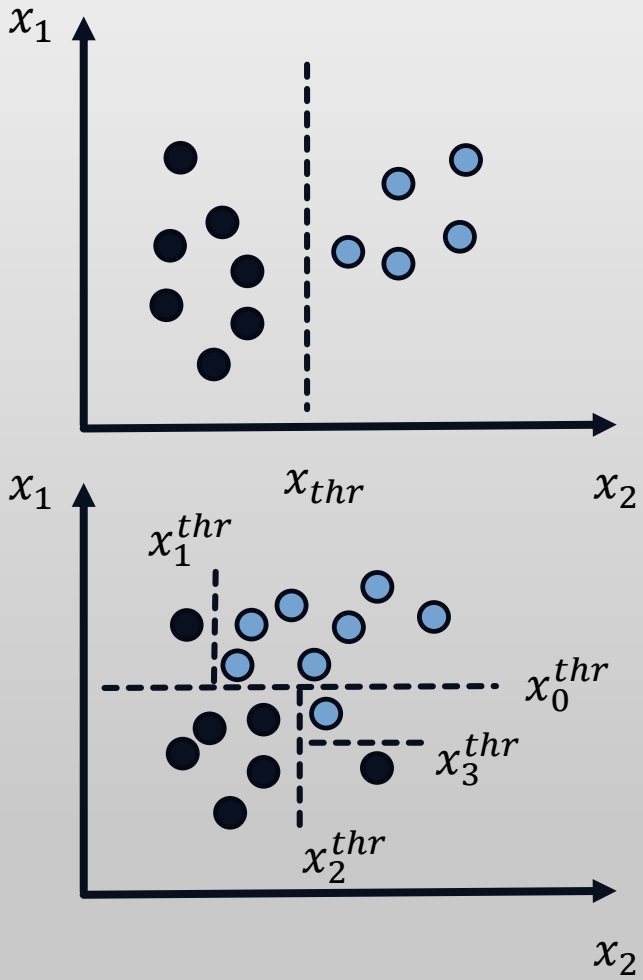


DECISION TREES

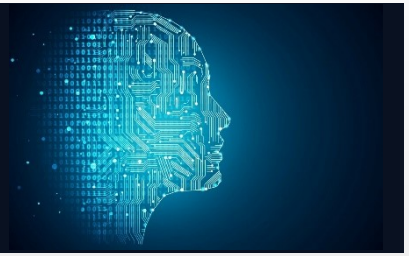


Decision Tree:



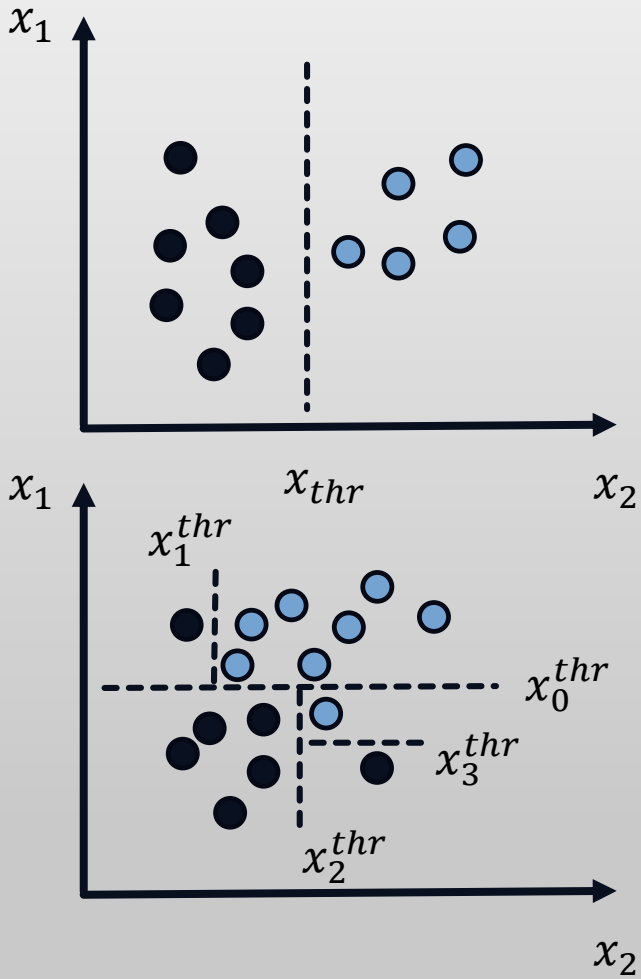


DECISION TREES

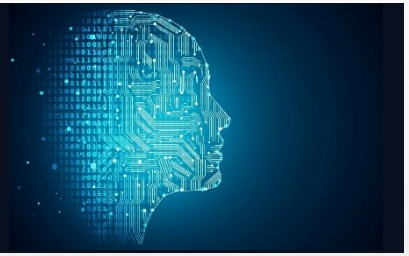


Decision Tree:

- Decision trees turn out to be unstable in practice
- A group of such trees do work well in practice
- Random Forest (Breiman et al) introduced the first such implementation



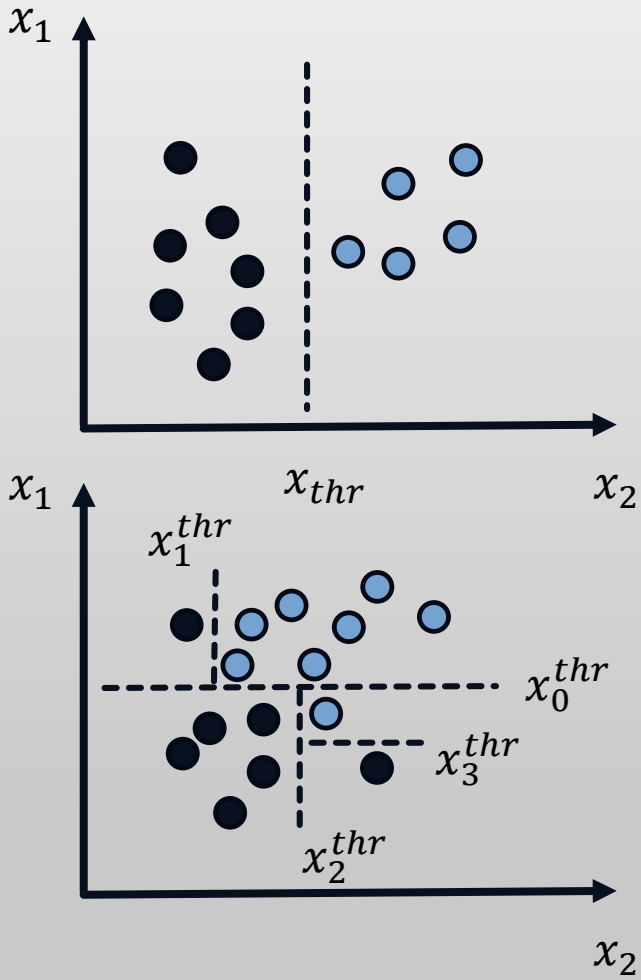
RANDOM FOREST



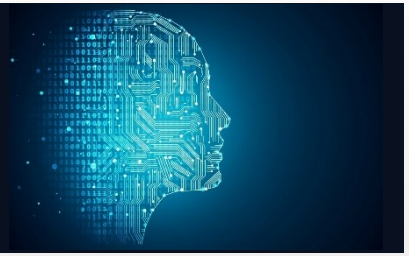
Assume we've modeled our data into features and we and we have labeled data. So, we have a supervised learning problem: $\mathrm{D} = \{X, y\}_{i=1}^{N}$

Random Forest uses a unique mix of brute force optimization and heuristic formulas:

- Pick a random subset from db (bagging-see next slide)
- Choose "best feature" from x_0, x_1 … (brute force optimization)
- Calculate "best threshold" x^{thr} (impurity heuristic)
- Continue until reaching max depth or until region contains min examples



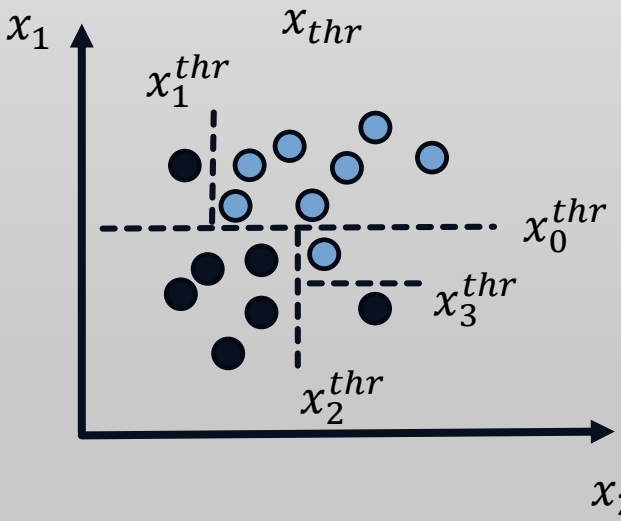
RANDOM FOREST - BAGGING



Bagging:

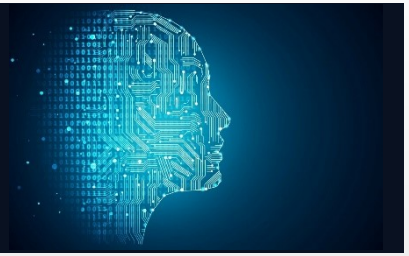
- Pick a random set of examples
- Pick a random set of features

→ This process results in a form of smoothing or variance reduction (see proof)



Observations\ Features	x_1	x_2	x_3	x_4
1				
2				
3				
4				
5				
6				
7				

RANDOM FOREST – OUT OF BAG EXAMPLE



Bagging:

- Pick a random set of examples
- Pick a random set of features

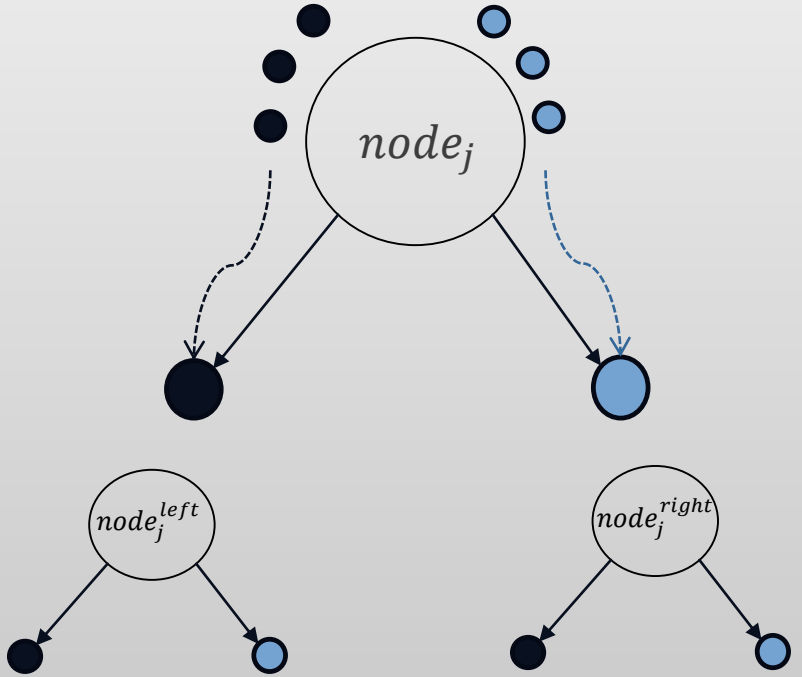
Out of Bag:

- Examples and features which are “left out” can be used to evaluate the model

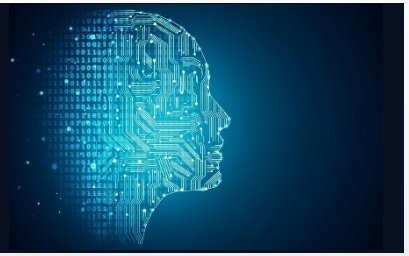
Feature importance f_j:

$$node_j = w_j \cdot c_j - w_j^{left} \cdot c_j^{left} - w_j^{right} \cdot c_j^{right}$$

- $node_j$ importance of node j
- w_j weighted number of samples reaching node j
- c_j impurity value of node j



RANDOM FOREST – OUT OF BAG EXAMPLE

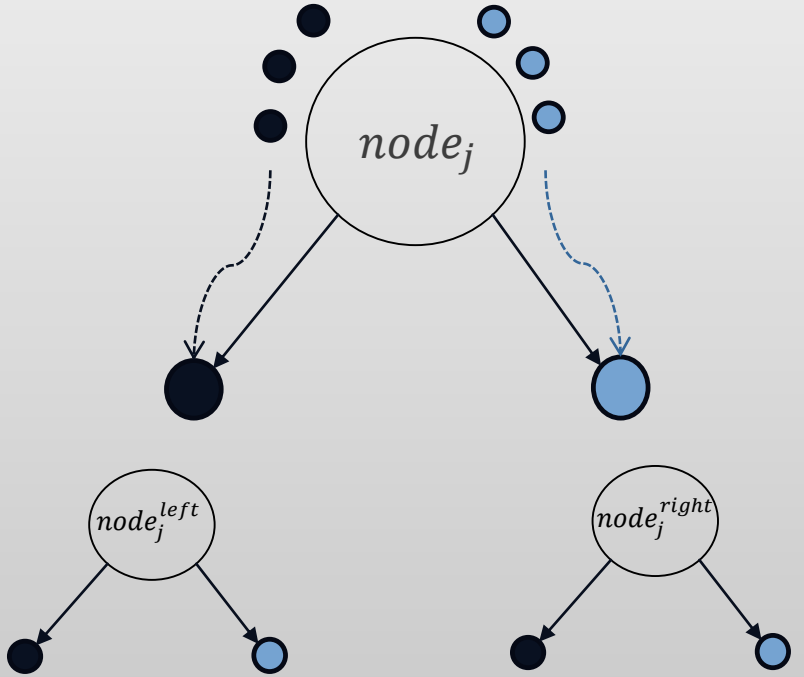


Feature importance f_j:

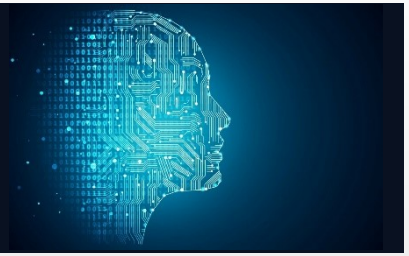
$$node_j = w_j \cdot c_j - w_j^{left} \cdot c_j^{left} - w_j^{right} \cdot c_j^{right}$$

- $node_j$ importance of node j
- w_j weighted number of samples reaching node j
- c_j impurity value of node j

$$f_j = \frac{\sum_{j:node_j\ splits\ on\ feature\ i} node_j}{\sum_{k \in all\ nodes} node_j^i}$$



RANDOM FOREST



Random Forest can be used for *classification* or *regression*:

- In the previous tree we chose a final class ●

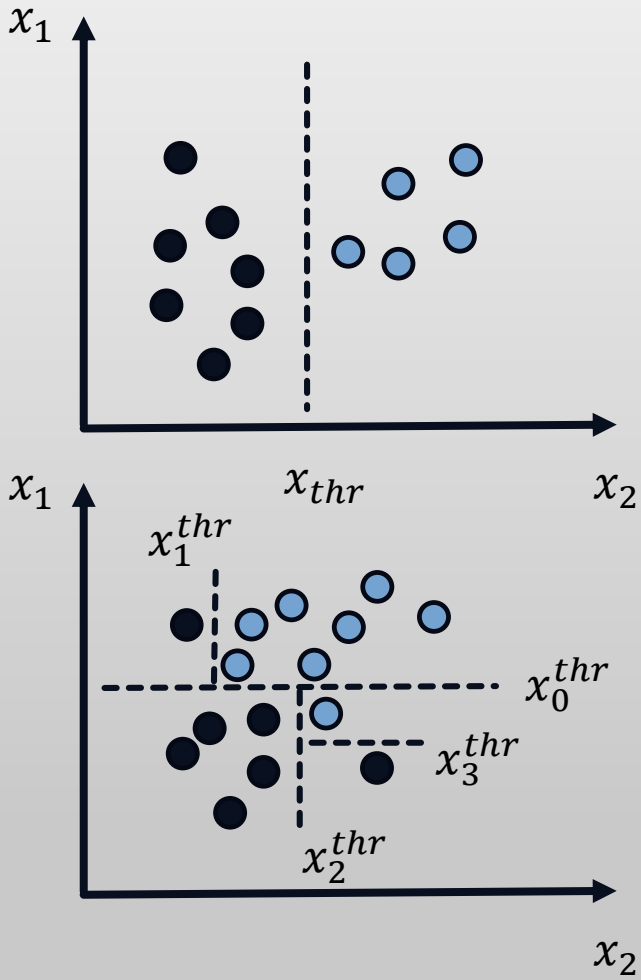
or ○

- For regression we can use the proportion of classes

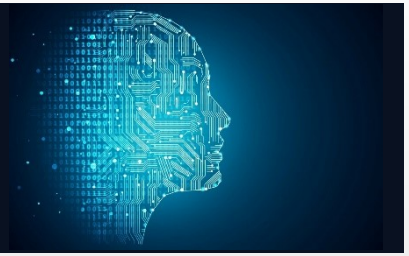
to form a number (between 0 and 1)

Exercise 4: Random Forest – Use feature selection (importance) of random forest classifier to determine what makes a good red wine.

- Discuss: What are the differences between PCA and Random Forest results.
- Discuss: Are red and white wine similar in this aspect?



STATISTICAL REASONING



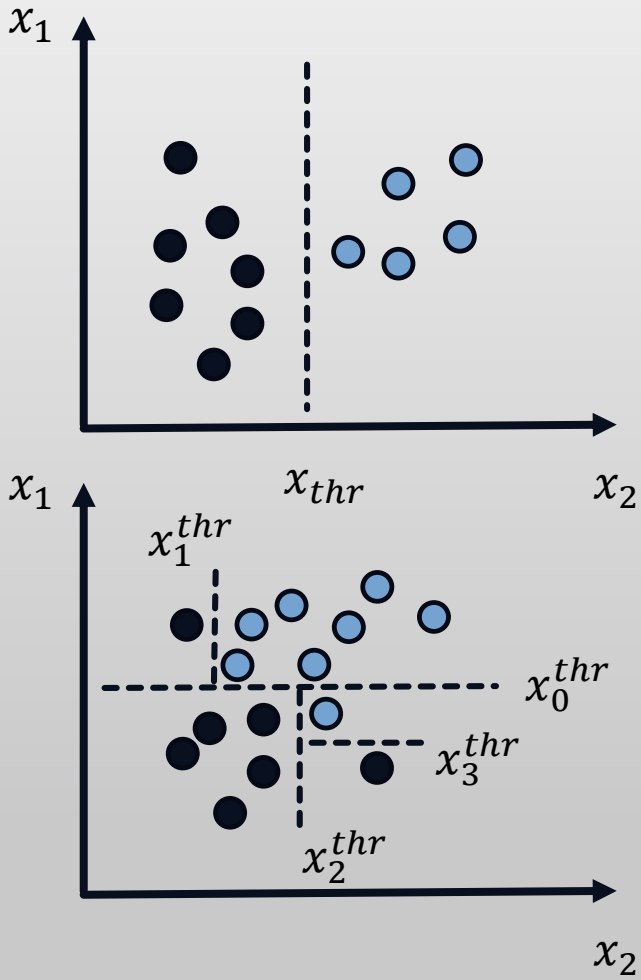
"Statistical reasoning is the way people reason with statistical ideas and make sense of statistical information. Statistical reasoning may involve connecting one concept to another (e.g., center and spread) or may combine ideas about data and chance. Reasoning means understanding and being able to explain statistical processes, and being able to fully interpret statistical results."

-Joan Garfield

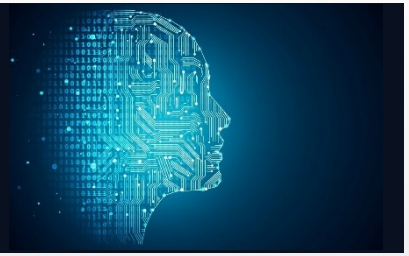
Recall what we learned:

- Normalization
- Feature Importance
- Supervised Classification and Regression

These are some of the basics of statistical reasoning in practice!



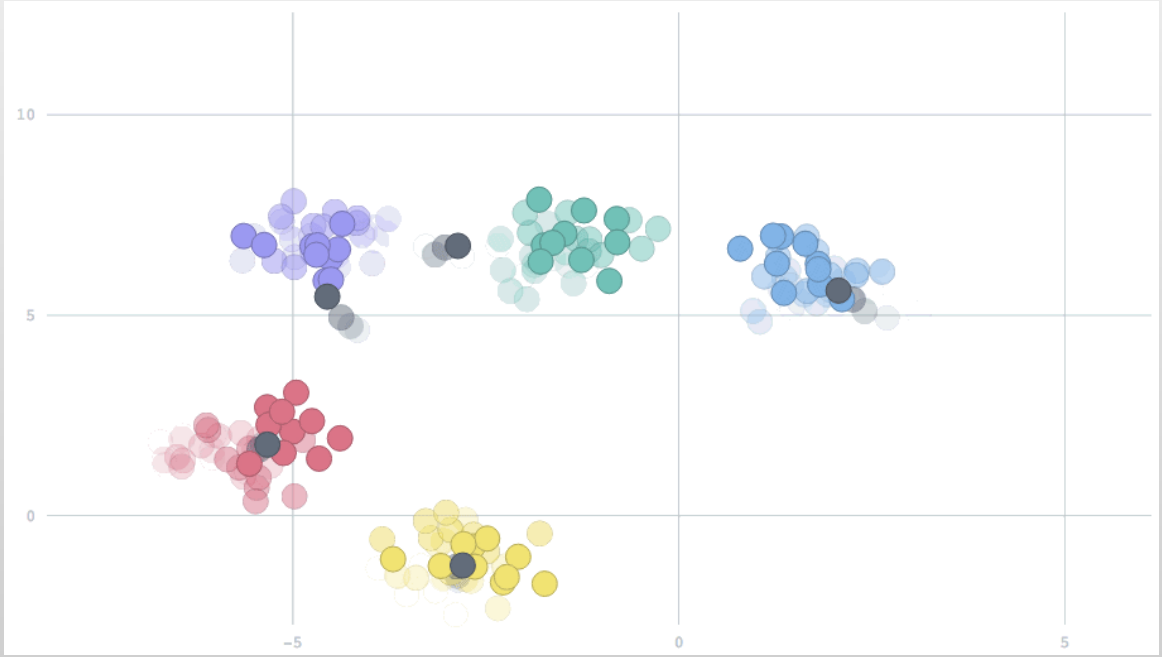
CLUSTERING



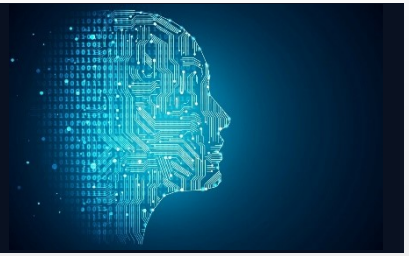
“Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense) to each other than to those in other groups (clusters)

-Wikipedia

Exercise 5: Random Forest – Use feature selection (importance) for clustering good and bad wines.



CLUSTERING – PCA VS TSNE

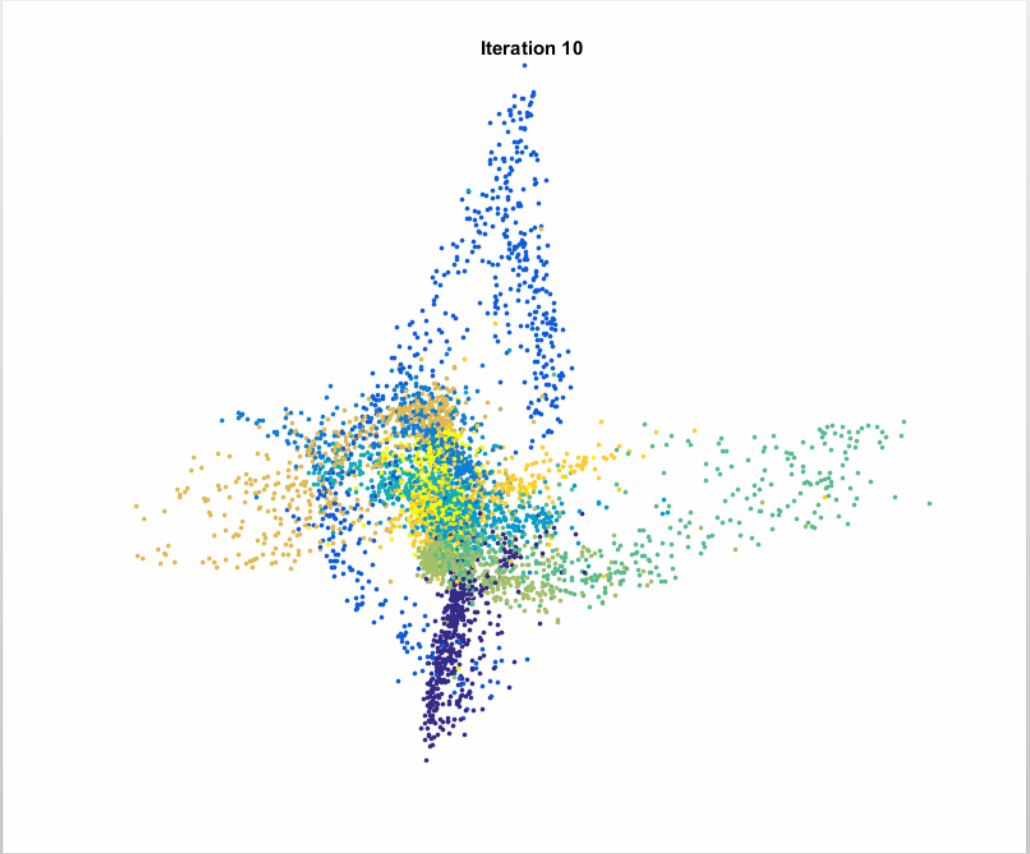


PCA uses the spectral theorem as a basis for dimensionality reduction. **TSNE** (T-Distributed Stochastic Neighboorhood Embedding) is another method with a completely different approach.

In a nutshell, it minimizes a (usually difficult) objective function and tries to understand which points are "close by" each other.

Notice the *runtime difference* and the different *results*!

Example: PCA vs TSNE



RANDOM FOREST

Exercise 6: Mini Project – IMDB Movie Recommendation System

- Implement a classifier on the IMDB dataset which decides if a review is positive or negative
- Implement noise removal and feature selection methods
- Test the performance on an independent test set