EURO²

Fundamental Concepts of Generative Machine Learning

Erdem Akagündüz, PhD
Graduate School of Informatics, METU, Türkiye

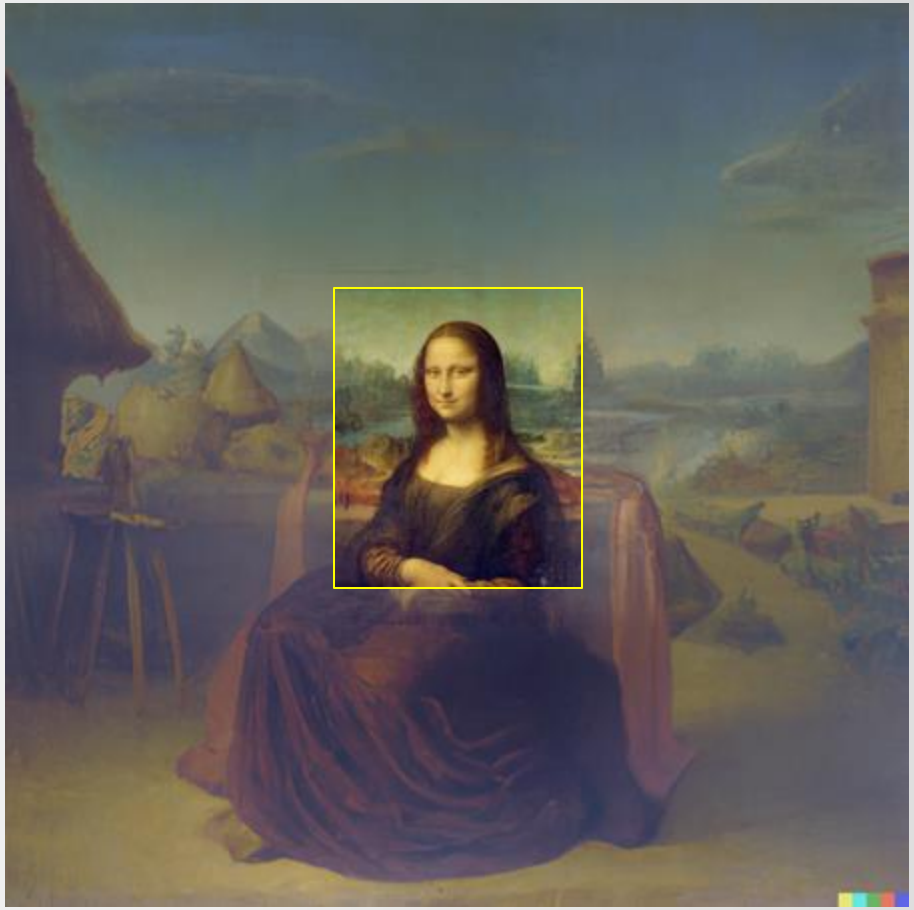
ncc@ulakbim.gov.tr

Lesson 2: Latent Spaces

Welcome to Part II: “Latent Spaces”

This part includes two subsections:

- **(The Curse of) Dimensionality, Deep Features vs. Latent Spaces**
- Latent Space properties: Continuity, Entanglement, etc



Dimensionality

In the context of generative models (and deep learning), dimensionality refers to the number of input features or variables that describe the input.

High-dimensional data poses challenges such as increased computational complexity and difficulty in understanding and visualizing the data.

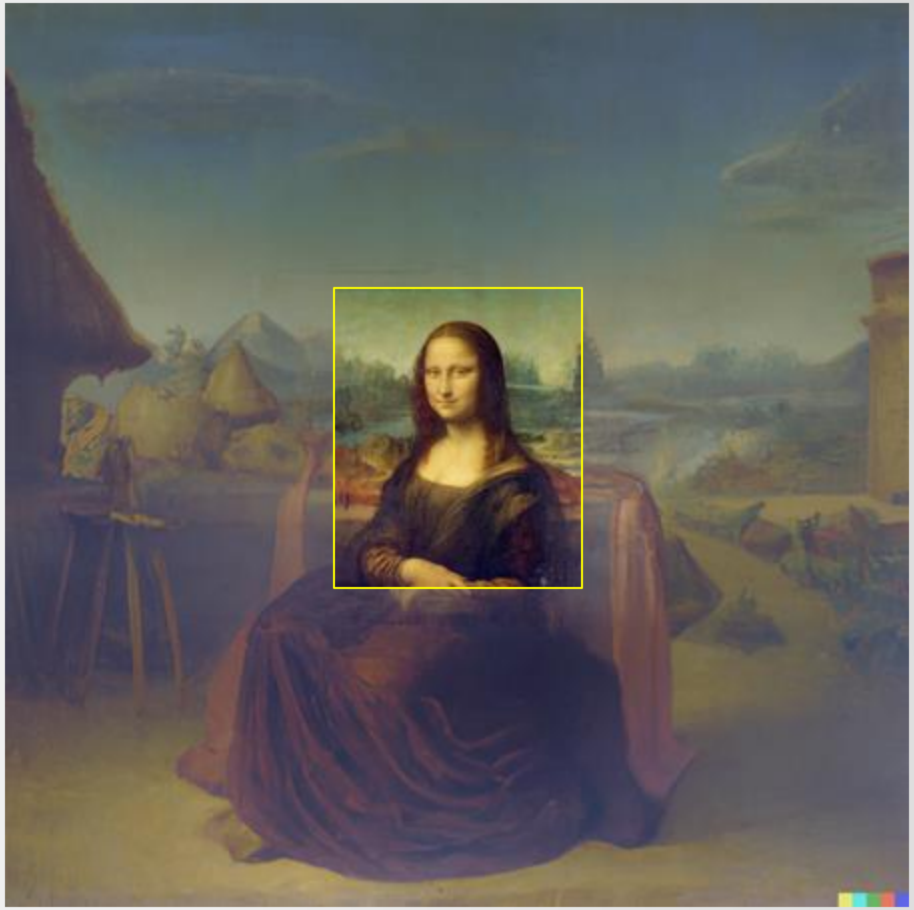


Image by openai.org

Dimensionality

Dimensionality reduction techniques play a vital role in mitigating these challenges by reducing the number of dimensions while preserving important information.

We'll dive into the motivations, techniques, and benefits of dimensionality reduction in generative modeling. This will lead us to the idea of a "latent space".

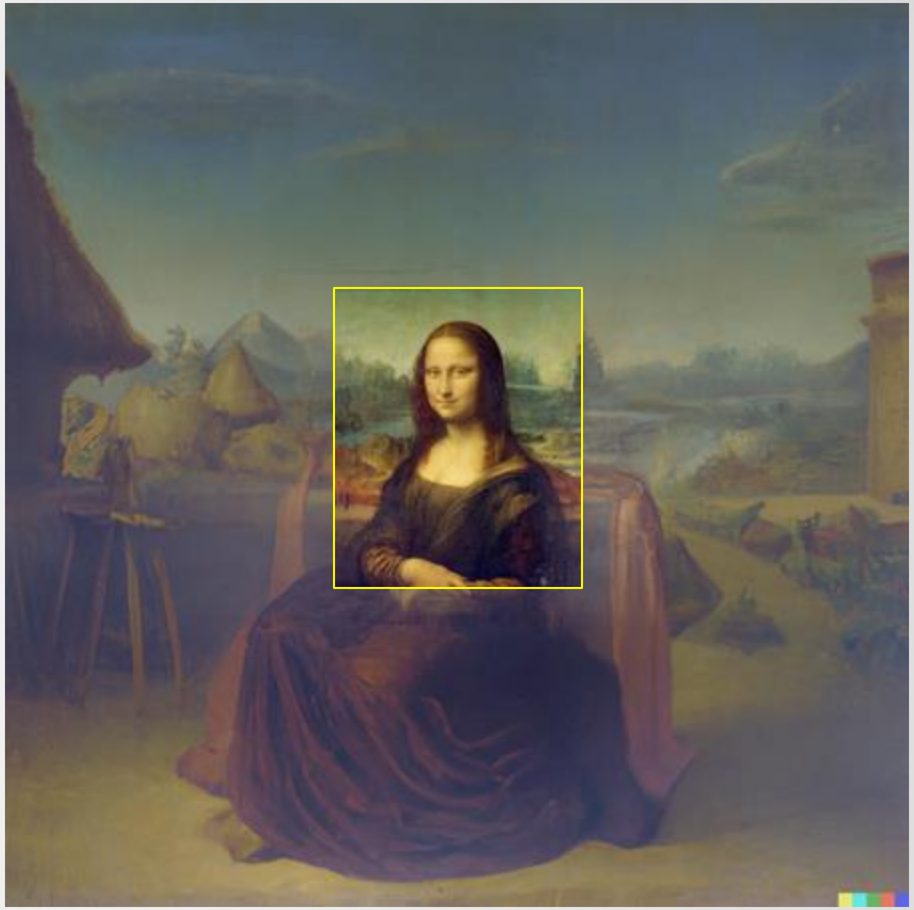
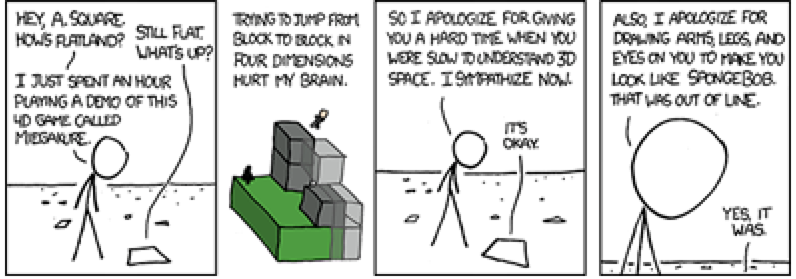


Image by openai.org

Curse of Dimensionality



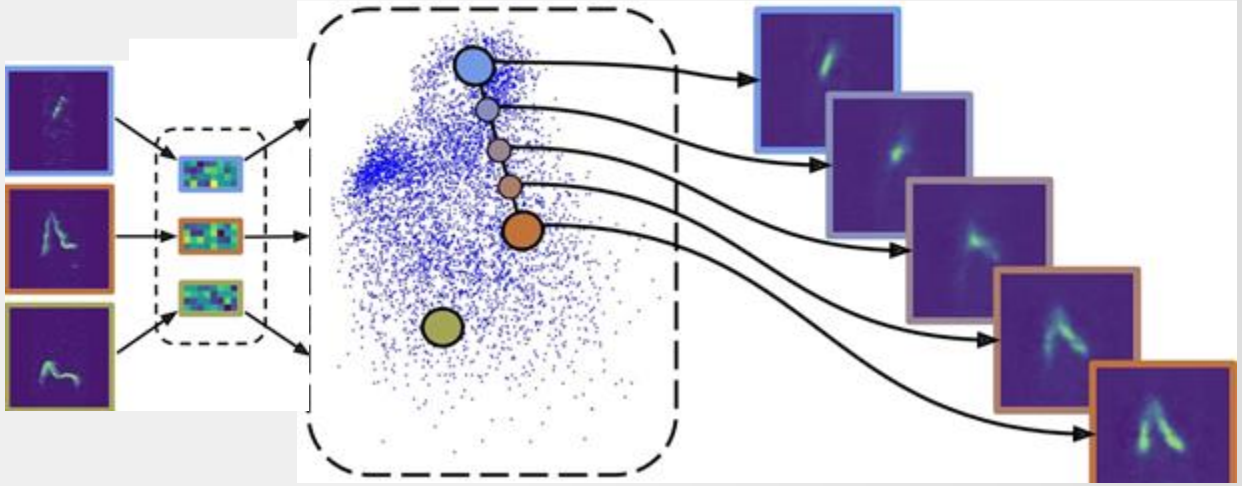
- In generative modeling (and in machine learning general), the curse of dimensionality refers to the challenges that arise when working with high-dimensional data.
- High-dimensional data refers to datasets with a large number of features or variables that describe each data point.
- CoD causes many problems in generative modelling such as “mode collapse” in GANs (which we’ll learn later)
- The curse of dimensionality necessitates the need for dimensionality reduction techniques to address these challenges.

figure by miegakure.com

Sampling from a Vector Space

- In generative modeling, we require a intermediate vector space that is both **continuous** and **representative** of the underlying data distribution for effective sampling from a generator.
- A continuous vector space enables smooth interpolation and exploration of the data, allowing for **seamless transitions** and generation of new samples.

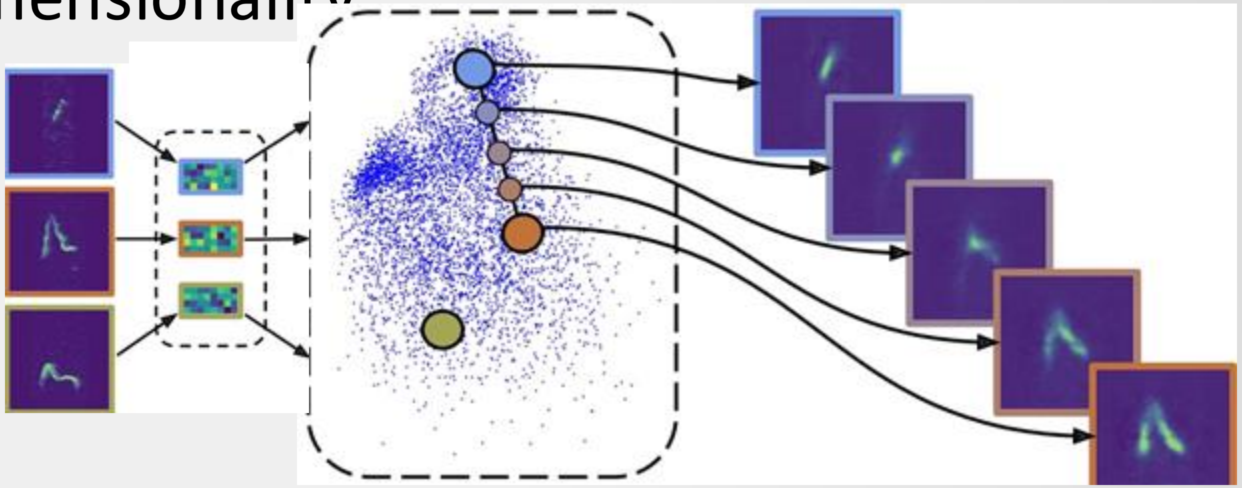
figure from https://doi.org/10.7554/eLife.67855



Sampling from a Vector Space

- Representativeness implies that similar samples or concepts in the original data space should be close to each other in the vector space, facilitating accurate modeling and sampling.
- However, directly working with the high-dimensional data space poses challenges in achieving a continuous and representative vector space due to the curse of dimensionality.
- We need techniques that maps high-dimensional data to a lower-dimensional vector space.

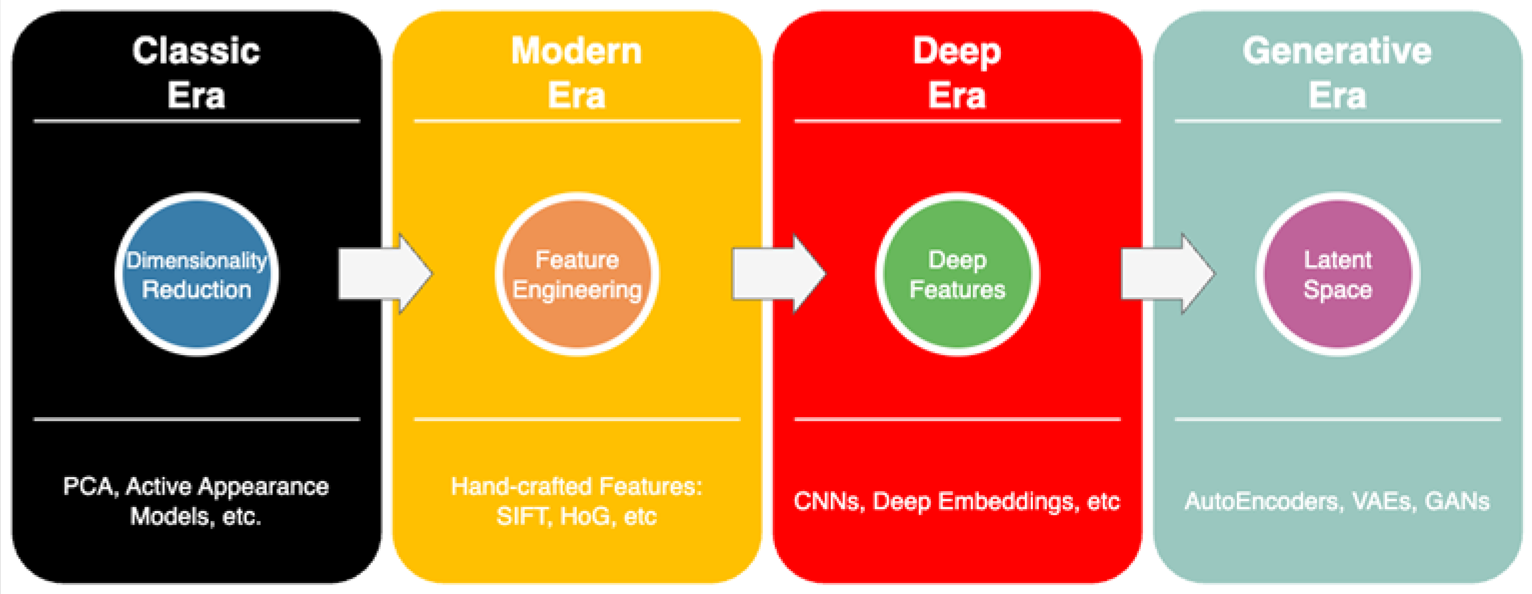
figure from https://doi.org/10.7554/eLife.67855



Evolution

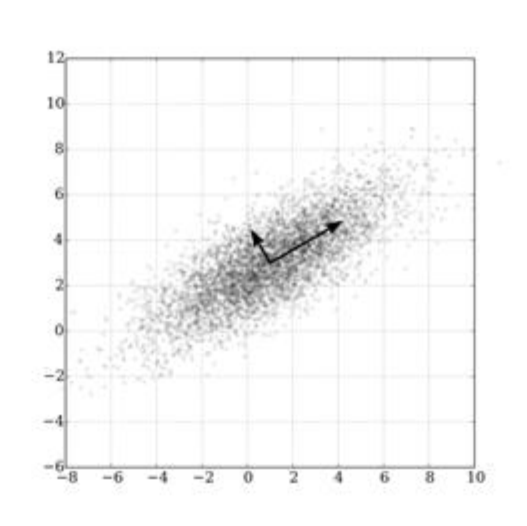
This is a categorisation of eras I came up with myself. I am open to ideas, discussions and improvement.

- In the field of generative modeling, the quest to find a suitable vector space evolved through distinct eras, each characterized by different techniques ar



Classical Methods

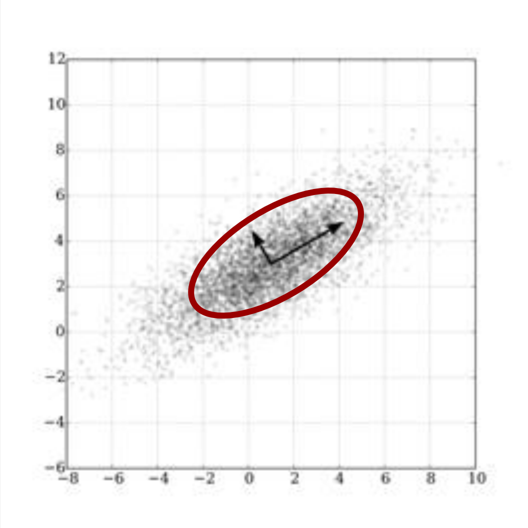
- Classical methods like Principal Component Analysis (PCA) were primarily developed for dimensionality reduction.
- The goal was to capture the most important components of the data, reducing the dimensionality while preserving as much information as possible.
- The resulting principal components capture the directions of maximum variance in the **original data**.



Classical Methods

- Classical methods like Principal Component Analysis (PCA) were primarily developed for dimensionality reduction.
- The goal was to capture the most important components of the data, reducing the dimensionality while preserving as much information as possible.
- To create a distribution representing the dataset, you can model the principal components as a multivariate normal distribution.

$$p(\boldsymbol{x}) = \frac{1}{\left|\sqrt{2\pi\boldsymbol{\Sigma}}\right|} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}-\boldsymbol{\mu})}$$

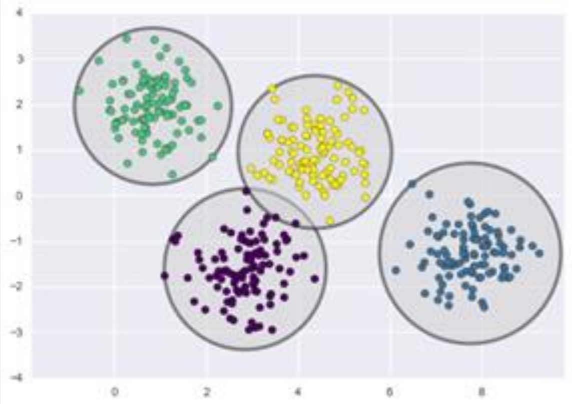


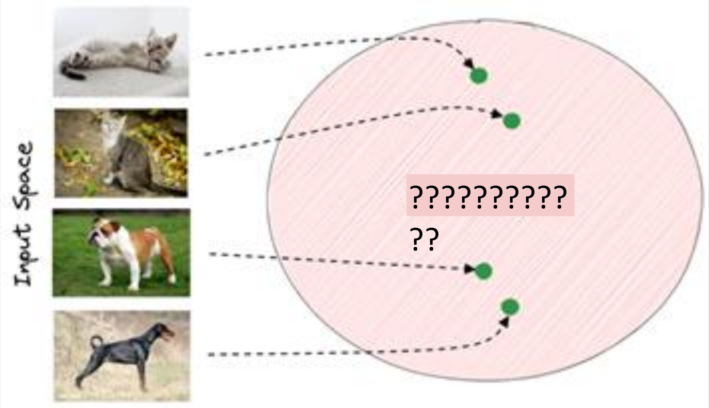
Generation with Classical Methods

- For complex data (like images etc) this idea failed miserably because:

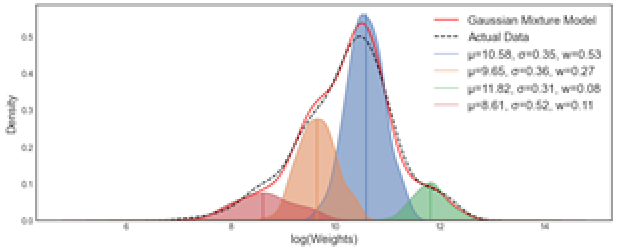
1 What if the data is multi-modal? PCA assumes that the data follows a unimodal Gaussian distribution along the principal components

2 Classical methods may not capture all the complex and high-level semantics of the data. They focus on capturing statistical variations rather than semantic meaning. The generated samples may resemble the statistical properties of the original data, but they may not capture the full complexity or exhibit higher-level semantics.





Gaussian Mixture Models (GMM)



A Gaussian mixture model (GMM) is a generative probabilistic model that consists of multiple Gaussian distributions.

- It is used to model complex data that cannot be represented by a single Gaussian distribution.
- GMMs are commonly used in clustering and density estimation tasks.
- The parameters of a GMM include the number of mixture components, the mean and standard deviation of each component, and the mixing coefficients that determine the weight of each component.

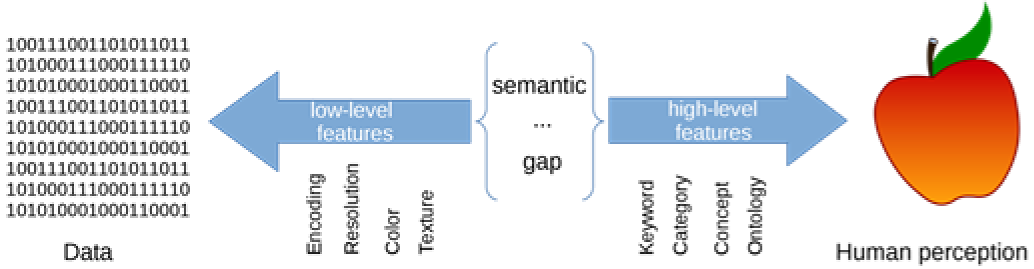
GMM for Generation?

What kinds of problems focus on utilizing simple generation techniques like GMMs for data-related tasks ?

(rather than explicitly incorporating semantic or higher-level feature spaces)

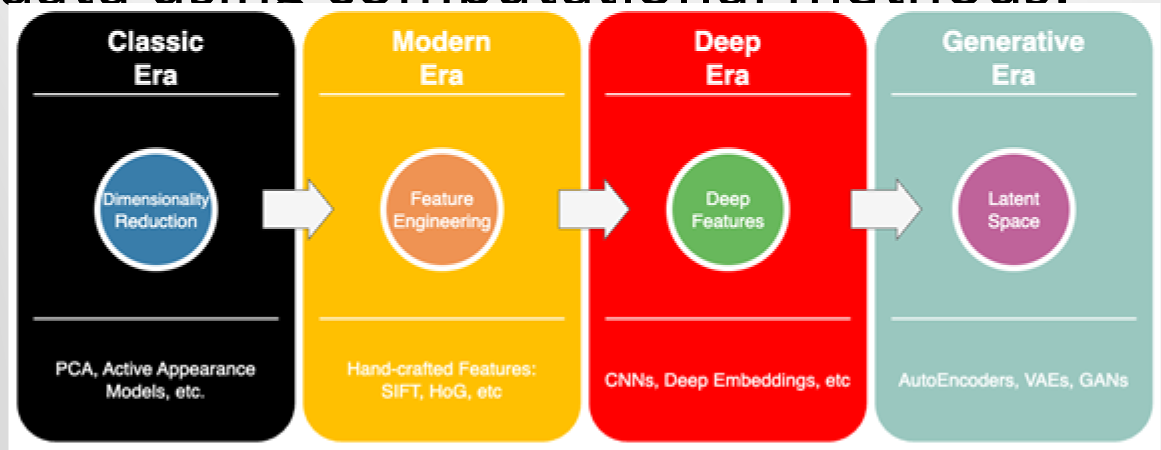
- Simulations
- Games/Animation
- Outlier Detection/Statistical Analysis
- Compression

Semantic Gap

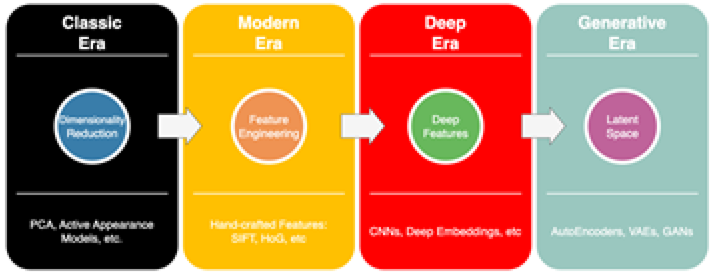


The "semantic gap" is a well-known problem in the field of AI that refers to the disparity between the low-level, perceptual representation of data and the high-level, semantic understanding and interpretation of that data.

- In simpler terms, the semantic gap problem highlights the difficulty in capturing and representing the rich and complex semantics, meaning, and context of data using computational methods.



Feature Space Evaluation



EURO²

- Modern Era: Pattern Recognition and Handcrafted Features
  - Focus on the use of handcrafted features, such as SIFT and HOG,
  - Dimensionality Reduction → Feature Engineering:
    - (Active Appearance Models)
- Deep Era: Deep Feature Learning
  - By training a deep encoder on a large-scale dataset, automatically extracting hierarchical and abstract features from the input data.
  - Feature Engineering → Feature Learning:
    - (Alexnet)
- Generative Era: The Latent Space
  - Deep features shift into latent spaces for generative purposes
  - Feature Learning → Latent Space:
    - (VAEs)

Next lecture:

- PART II: “Latent Spaces”

- (The Curse of) Dimensionality, Deep Features vs. Latent Spaces
- **Latent Space properties: Continuity, Entanglement, etc**

Thanks

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 101101903. The JU receives support from the Digital Europe Programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Türkiye, Republic of North Macedonia, Iceland, Montenegro, Serbia