

# Deep Learning & Applied AI

Data, features, and embeddings

Emanuele Rodolà  
[rodola@di.uniroma1.it](mailto:rodola@di.uniroma1.it)



SAPIENZA  
UNIVERSITÀ DI ROMA

# Data awareness

Machine learning involves dealing with **data**.

What do you do when you have a problem involving data?

## Data awareness

Machine learning involves dealing with **data**.

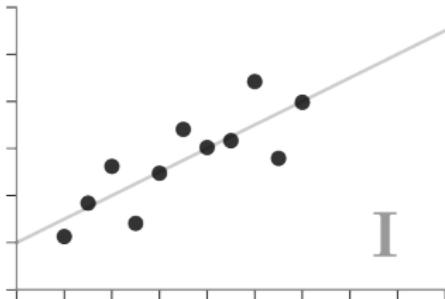
What do you do when you have a problem involving data?

First thing: **look at the data!**

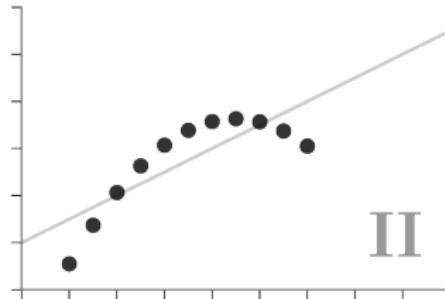


## Anscombe's Quartet

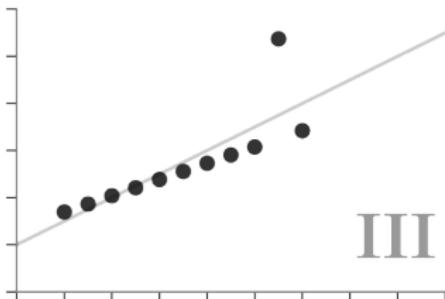
Each dataset has the same summary statistics (mean, standard deviation, correlation), and the datasets are *clearly different*, and *visually distinct*.



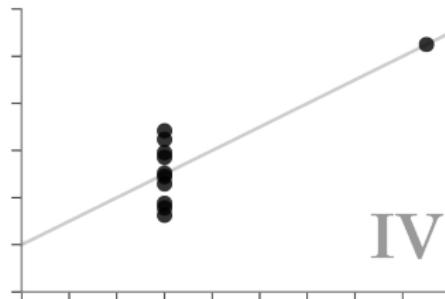
I



II



III

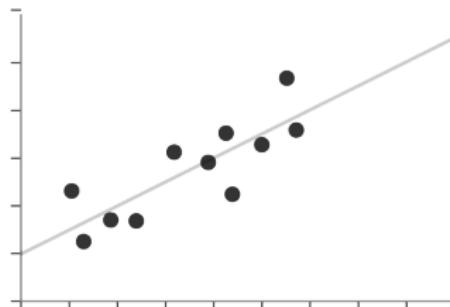
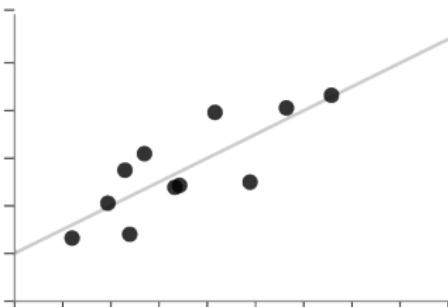
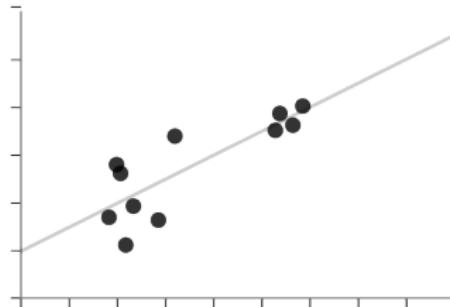
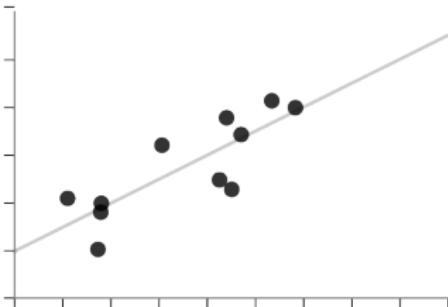


IV



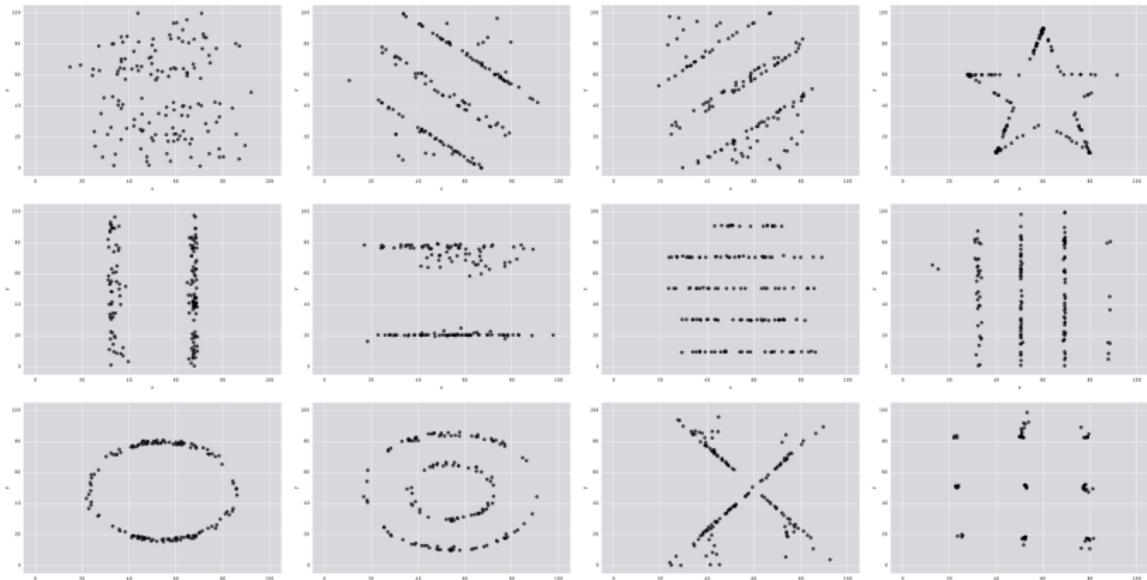
## Unstructured Quartet

Each dataset here also has the same summary statistics. However, they are not *clearly different* or *visually distinct*.



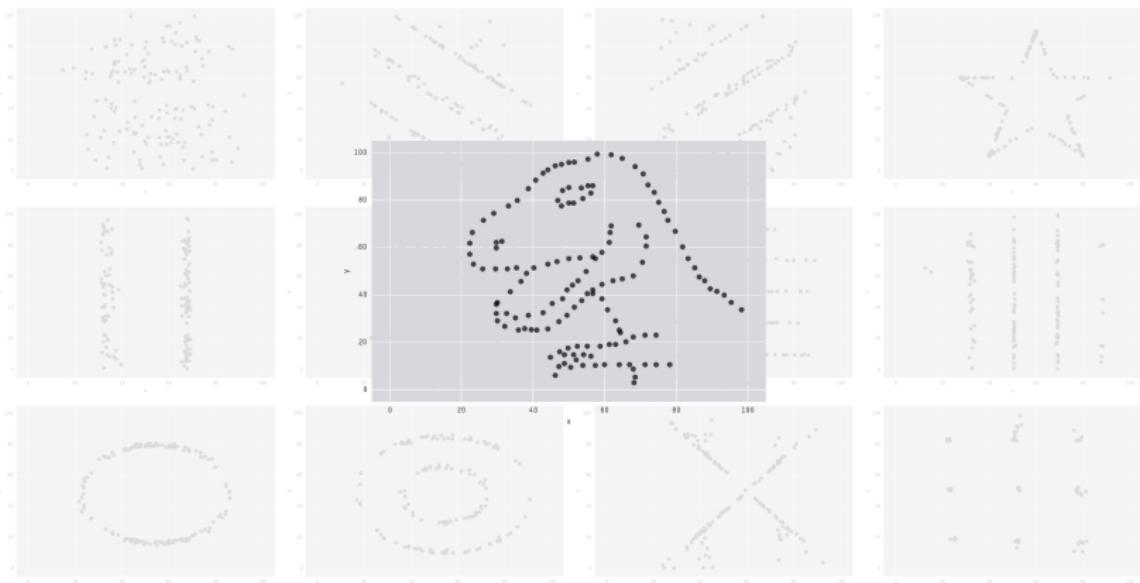
# The datasaurus dozen

All these [datasets](#) have the same summary stats to 2 decimal places:



# The datasaurus dozen

All these [datasets](#) have the same summary stats to 2 decimal places:



## Data awareness

Machine learning involves dealing with **data**.

What do you do when you have a problem involving data?

First thing: **look at the data!**

Never trust summary statistics alone;  
when possible, visualize your data

## Data awareness

Machine learning involves dealing with **data**.

What do you do when you have a problem involving data?

First thing: **look at the data!**

Never trust summary statistics alone;  
when possible, visualize your data

It will not always be easy to visualize.

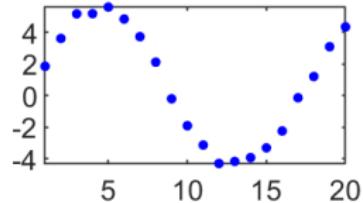
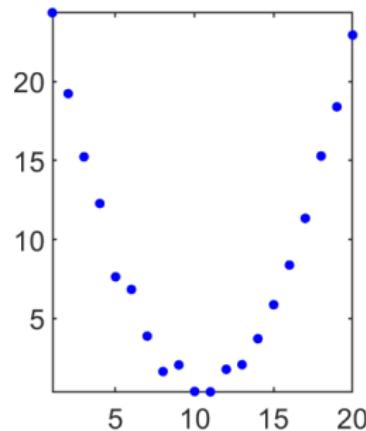
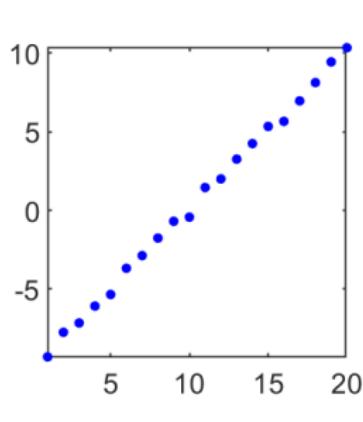
Difficult cases: **high-dimensional** data, **no physical access** to data, **implicit access** to data (e.g. latent spaces).

## Models for describing the data

Learning is about **describing** data, or more specifically, describing the **process**, or **model**, that yields a given output from a given input.

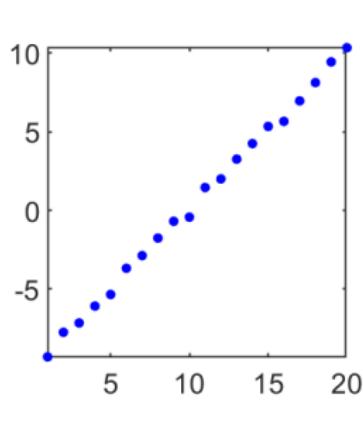
# Models for describing the data

Learning is about **describing** data, or more specifically, describing the **process**, or **model**, that yields a given output from a given input.

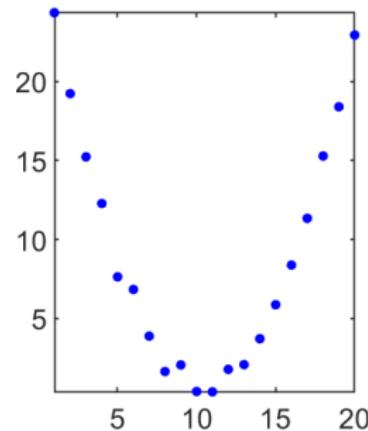


# Models for describing the data

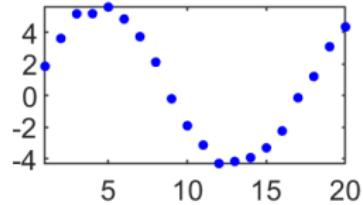
Learning is about **describing** data, or more specifically, describing the **process**, or **model**, that yields a given output from a given input.



$$y = ax + b$$



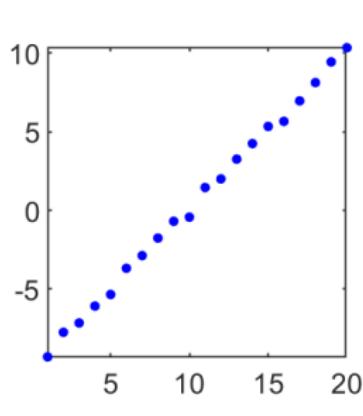
$$y = ax^2 + bx + c$$



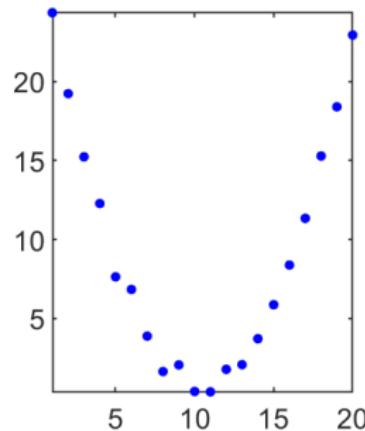
$$y = ax^3 + bx^2 + cx + d$$

# Models for describing the data

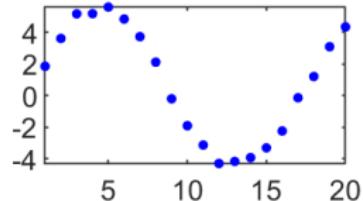
Learning is about **describing** data, or more specifically, describing the **process**, or **model**, that yields a given output from a given input.



$$y = ax + b$$



$$y = ax^2 + bx + c$$



$$y = a \sin(x) + bx + c$$

Our model might use **prior knowledge** on the data.

For example, in the third plot, we might know *a priori* that the data actually comes from a periodic process.

## Modeling prior knowledge

Main idea: Look at the world, identify what knowledge we have about it, and use this knowledge to construct our model.

## Modeling prior knowledge

Main idea: Look at the world, identify what knowledge we have about it, and use this knowledge to construct our model.

Some forms of prior knowledge:

- Data distribution

## Modeling prior knowledge

Main idea: Look at the world, identify what knowledge we have about it, and use this knowledge to construct our model.

Some forms of prior knowledge:

- Data distribution
- Energy function

## Modeling prior knowledge

Main idea: Look at the world, identify what knowledge we have about it, and use this knowledge to construct our model.

Some forms of prior knowledge:

- Data distribution
- Energy function
- Constraints

## Modeling prior knowledge

Main idea: Look at the world, identify what knowledge we have about it, and use this knowledge to construct our model.

Some forms of prior knowledge:

- Data distribution
- Energy function
- Constraints
- Invariances

## Modeling prior knowledge

Main idea: Look at the world, identify what knowledge we have about it, and use this knowledge to construct our model.

Some forms of prior knowledge:

- Data distribution
- Energy function
- Constraints
- Invariances
- Input-output examples (data prior)

## Modeling prior knowledge

Main idea: Look at the world, identify what knowledge we have about it, and use this knowledge to construct our model.

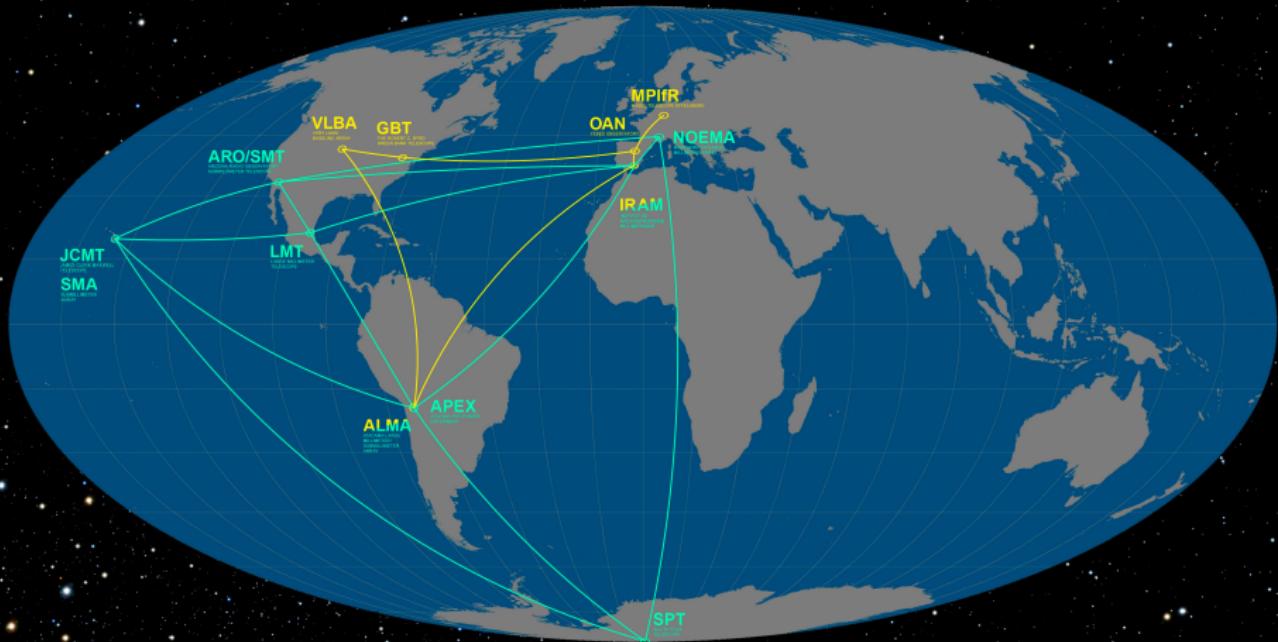
Some forms of prior knowledge:

- Data distribution
- Energy function
- Constraints
- Invariances
- Input-output examples (data prior)

All these encode, to different extents, some expected behavior.



# Event Horizon Telescope



## Reliability of the prior: imaging the black hole

**Problem:** reconstruct an image from a sparse set of spectral measurements (VLBI imaging).

# Reliability of the prior: imaging the black hole

**Problem:** reconstruct an image from a sparse set of spectral measurements (VLBI imaging).

It is an ill-posed [inverse problem](#):

- Infinite number of possible images explain the data

# Reliability of the prior: imaging the black hole

**Problem:** reconstruct an image from a sparse set of spectral measurements (VLBI imaging).

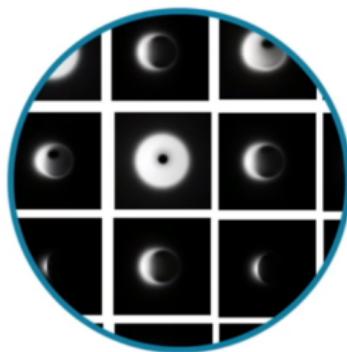
It is an ill-posed [inverse problem](#):

- Infinite number of possible images explain the data
- Optimization heavily relies on [priors](#).  
Find an explanation that respects prior assumptions about the “visual” universe while still satisfying the observed data.

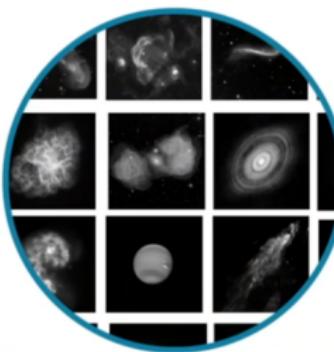
# Reliability of the prior: imaging the black hole

**Problem:** reconstruct an image from a sparse set of spectral measurements (VLBI imaging).

Which of these datasets would you use?



black holes



astronomy

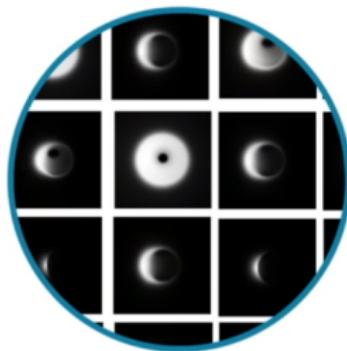


everyday

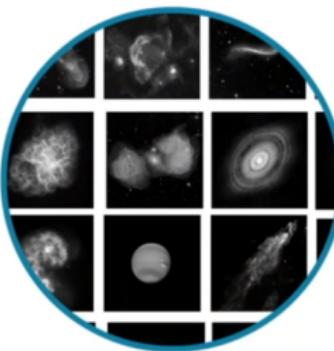
# Reliability of the prior: imaging the black hole

**Problem:** reconstruct an image from a sparse set of spectral measurements (VLBI imaging).

Which of these datasets would you use?



black holes  
**unreliable**



astronomy



everyday

Black holes are dangerous! They will yield what one **expects** to obtain.

## Reliability of the prior: fairness

AI is objective only in the sense of learning what humans teach.

The data provided by human can be **highly biased**.

## Reliability of the prior: fairness

AI is objective only in the sense of learning what humans teach.

The data provided by human can be **highly biased**.

	WHITE	AFRICAN AMERICAN
Labeled Higher Risk, But Didn't Re-Offend	23.5%	44.9%
Labeled Lower Risk, Yet Did Re-Offend	47.7%	28.0%

*Overall, Northpointe's assessment tool correctly predicts recidivism 61 percent of the time. But blacks are almost twice as likely as whites to be labeled a higher risk but not actually re-offend. It makes the opposite mistake among whites: They are much more likely than blacks to be labeled lower risk but go on to commit other crimes. (Source: ProPublica analysis of data from Broward County, Fla.)*

# Reliability of the prior: fairness

AI is objective only in the sense of learning what humans teach.

The data provided by human can be **highly biased**.

Search query	Work experience	Education experience	Profile views	Candidate	Xing ranking
Brand Strategist	146	57	12992	male	1
Brand Strategist	327	0	4715	female	2
Brand Strategist	502	74	6978	male	3
Brand Strategist	444	56	1504	female	4
Brand Strategist	139	25	63	male	5
Brand Strategist	110	65	3479	female	6
Brand Strategist	12	73	846	male	7
Brand Strategist	99	41	3019	male	8
Brand Strategist	42	51	1359	female	9
Brand Strategist	220	102	17186	female	10

## Reliability of the prior: fairness

AI is objective only in the sense of learning what humans teach.

The data provided by human can be **highly biased**.

Bias in the training dataset is still an open research problem!

Some possible causes:

- **Skewed sample:** a tiny initial bias grows over time, since future observations confirm prediction.

## Reliability of the prior: fairness

AI is objective only in the sense of learning what humans teach.

The data provided by human can be **highly biased**.

Bias in the training dataset is still an open research problem!

Some possible causes:

- **Skewed sample:** a tiny initial bias grows over time, since future observations confirm prediction. Example: Police intercept crime more densely in areas they watch.

## Reliability of the prior: fairness

AI is objective only in the sense of learning what humans teach.

The data provided by human can be **highly biased**.

Bias in the training dataset is still an open research problem!

Some possible causes:

- **Skewed sample:** a tiny initial bias grows over time, since future observations confirm prediction. Example: Police intercept crime more densely in areas they watch.
- **Tainted examples:** data produced by a human decision can be biased, and the bias is replicated by the system.

## Reliability of the prior: fairness

AI is objective only in the sense of learning what humans teach.

The data provided by human can be **highly biased**.

Bias in the training dataset is still an open research problem!

Some possible causes:

- **Skewed sample:** a tiny initial bias grows over time, since future observations confirm prediction. Example: Police intercept crime more densely in areas they watch.
- **Tainted examples:** data produced by a human decision can be biased, and the bias is replicated by the system.
- **Sample size disparity:** training data for a minority group is much less than the majority group.

## Reliability of the prior: fairness

AI is objective only in the sense of learning what humans teach.

The data provided by human can be **highly biased**.

Bias in the training dataset is still an open research problem!

Some possible causes:

- **Skewed sample**: a tiny initial bias grows over time, since future observations confirm prediction. Example: Police intercept crime more densely in areas they watch.
- **Tainted examples**: data produced by a human decision can be biased, and the bias is replicated by the system.
- **Sample size disparity**: training data for a minority group is much less than the majority group.

Assessing **data and prior reliability** is crucial for any learning-based system.

# Explaining the data

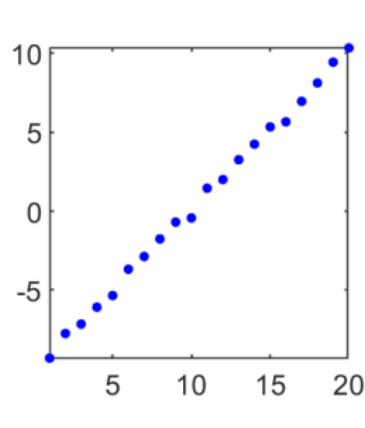
Learning is about discovering a **map** from input to output.

“Finding a model explaining the data” means determining the map.

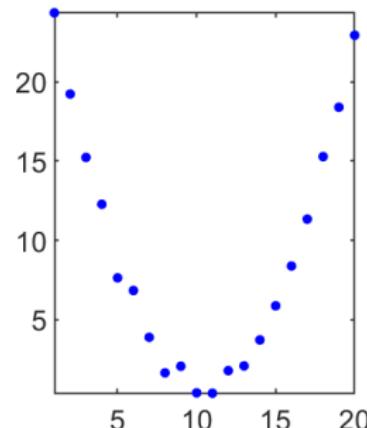
# Explaining the data

Learning is about discovering a **map** from **input** to **output**.

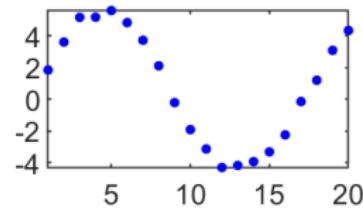
“Finding a model explaining the data” means determining the map.



$$y = ax + b$$



$$y = ax^2 + bx + c$$



$$y = a \sin(x) + bx + c$$

## Explaining the data

Learning is about discovering a **map** from input to output.

“Finding a model explaining the data” means determining the map.

**Key assumption:** the data has an **underlying structure**.

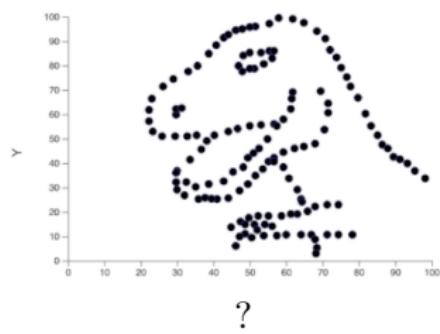
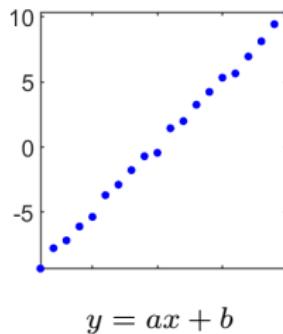
# Explaining the data

Learning is about discovering a **map** from input to output.

“Finding a model explaining the data” means determining the map.

**Key assumption:** the data has an **underlying structure**.

This structure is almost never captured by a simple expression.



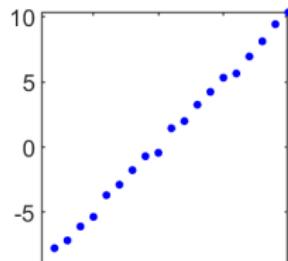
# Explaining the data

Learning is about discovering a **map** from input to output.

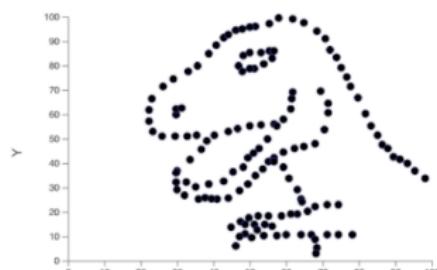
“Finding a model explaining the data” means determining the map.

**Key assumption:** the data has an **underlying structure**.

This structure is almost never captured by a simple expression.



$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_x \\ a_y \end{pmatrix} t + \begin{pmatrix} b_x \\ b_y \end{pmatrix}$$



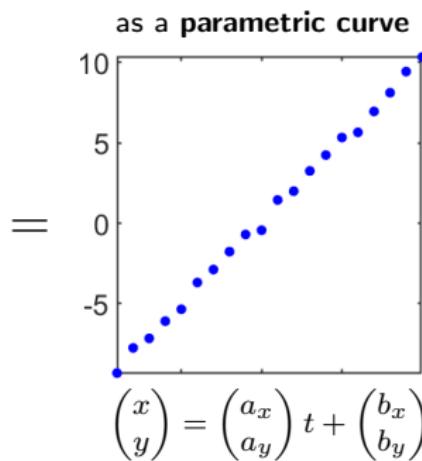
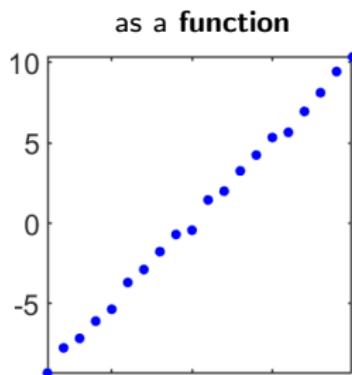
(not a function in 1D)

Clearly, data is not always one-dimensional.

# Choosing a representation

The same data can be described in different ways.

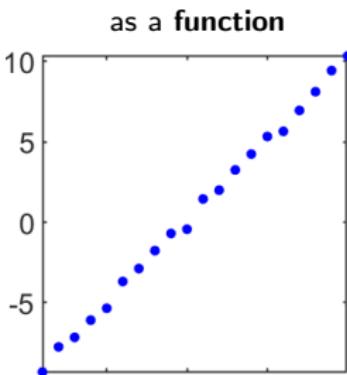
What is the “right” way?



# Choosing a representation

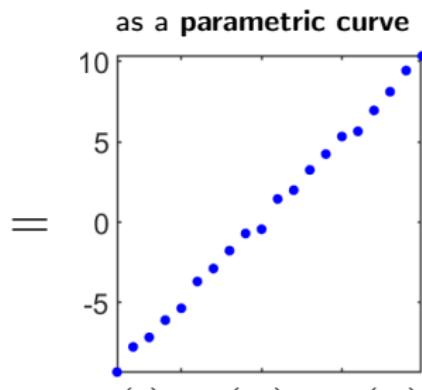
The same data can be described in different ways.

What is the “right” way?



$$y = ax + b$$

2 weights



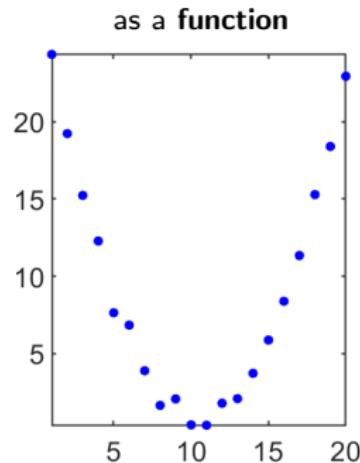
$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_x \\ a_y \end{pmatrix} t + \begin{pmatrix} b_x \\ b_y \end{pmatrix}$$

4 weights

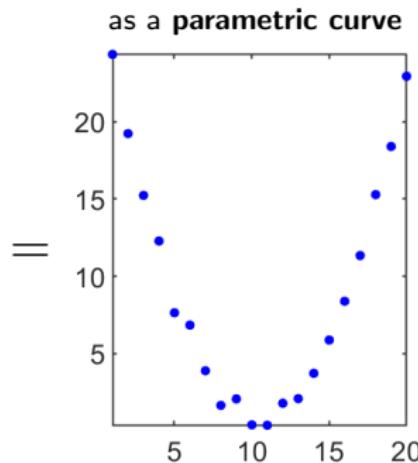
# Choosing a representation

The same data can be described in different ways.

What is the “right” way?



$$y = ax^2 + bx + c$$



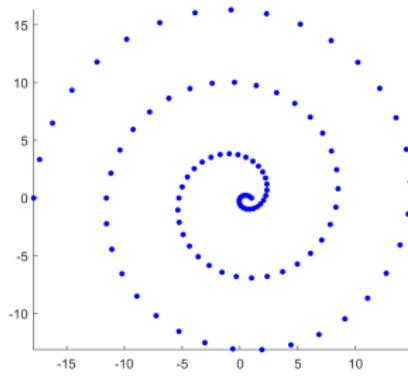
$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_x \\ a_y \end{pmatrix} t^2 + \begin{pmatrix} b_x \\ b_y \end{pmatrix} t + \begin{pmatrix} c_x \\ c_y \end{pmatrix}$$

# Choosing a representation

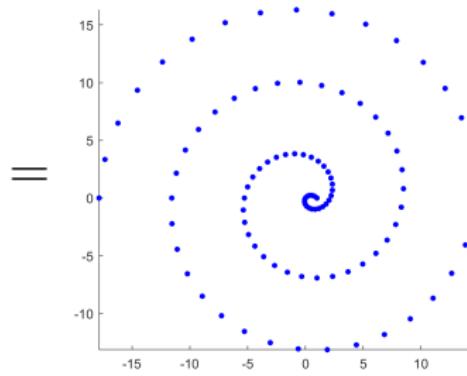
The same data can be described in different ways.

What is the “right” way?

as a **function**



as a **parametric curve**

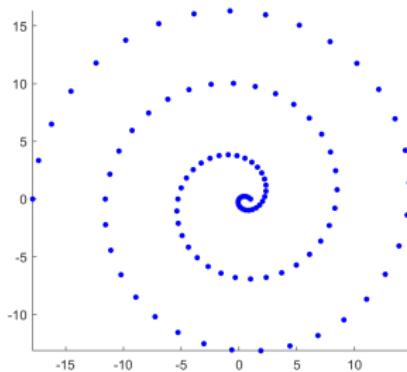


# Choosing a representation

The same data can be described in different ways.

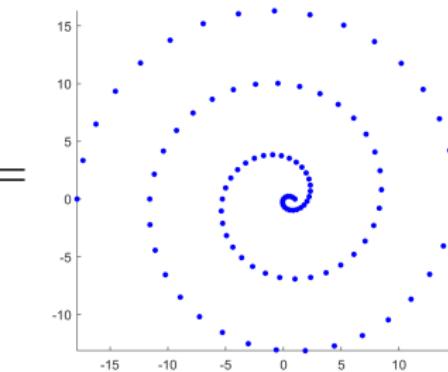
What is the “right” way?

as a **function**



*y* is not a function of *x*

as a **parametric curve**



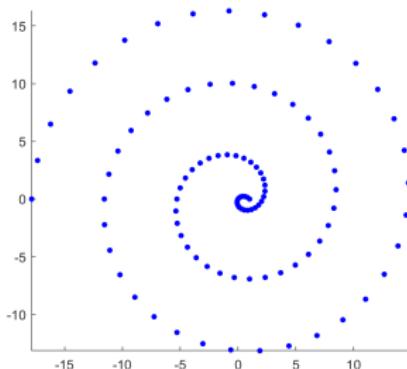
$$\begin{pmatrix} x \\ y \end{pmatrix} = a \begin{pmatrix} \cos t \\ \sin t \end{pmatrix} (a - t)$$

# Choosing a representation

The same data can be described in different ways.

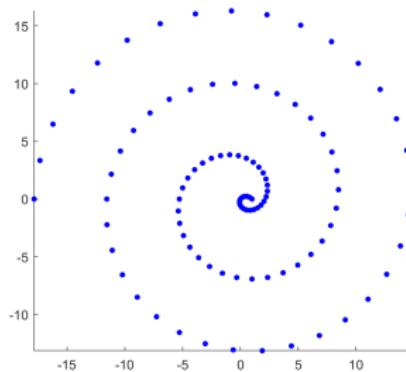
What is the “right” way?

as a **function**



$$r = a\theta \quad (\text{polar coordinates})$$

as a **parametric curve**



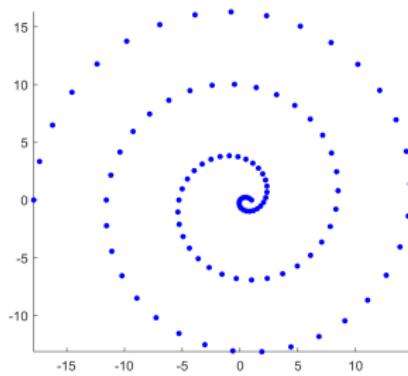
$$\begin{pmatrix} x \\ y \end{pmatrix} = a \begin{pmatrix} \cos t \\ \sin t \end{pmatrix} (a - t)$$

# Choosing a representation

The same data can be described in different ways.

What is the “right” way?

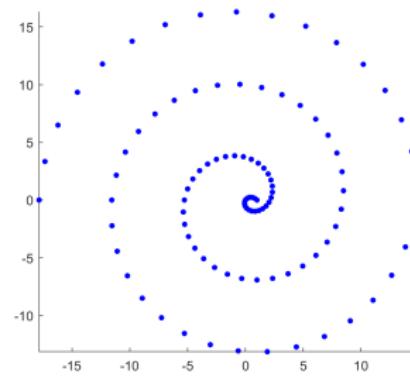
as a **function**



$$r = a\theta$$

**linear!**

as a **parametric curve**



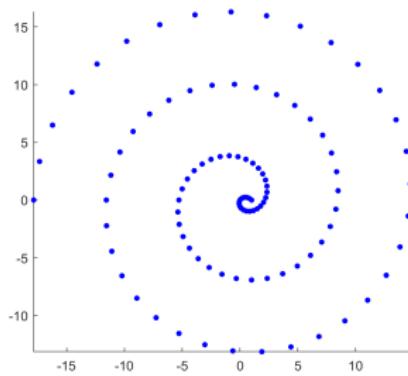
$$\begin{pmatrix} x \\ y \end{pmatrix} = a \begin{pmatrix} \cos t \\ \sin t \end{pmatrix} (a - t)$$

# Choosing a representation

The same data can be described in different ways.

What is the “right” way?

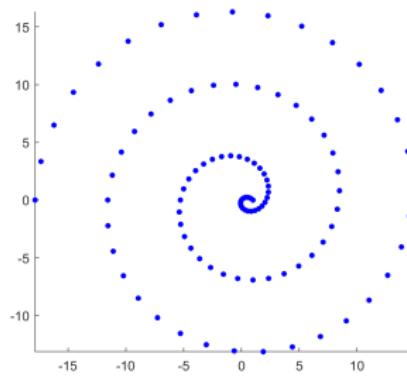
as a **function**



$$r = a\theta$$

**linear!**

as a **parametric curve**



$$\begin{pmatrix} x \\ y \end{pmatrix} = a \begin{pmatrix} \cos t \\ \sin t \end{pmatrix} (a - t)$$

Trade-off between #weights and simplicity

# The curse of dimensionality

Of course, data can have more than 1 or 2 dimensions.

# The curse of dimensionality

Of course, data can have more than 1 or 2 dimensions.

For example, a  $w \times h$  image has  $wh$  dimensions.



$$\in \mathbb{R}^{w \times h} \cong \mathbb{R}^{wh}$$

Example:  $\sim 1$  megapixel photo (grayscale) has  $\sim 10^6$  dimensions.

# The curse of dimensionality

Of course, data can have more than 1 or 2 dimensions.

For example, a  $w \times h$  image has  $wh$  dimensions.

The value of each coordinate is given by the gray value at that pixel.  
Then, the entire image is **one point** in a  $wh$ -dimensional space.



$$\in \mathbb{R}^{w \times h} \cong \mathbb{R}^{wh}$$

Example:  $\sim 1$  megapixel photo (grayscale) has  $\sim 10^6$  dimensions.

Are all those dimensions significant?

# The curse of dimensionality

For simplicity, consider  $1 \times 1$  images, i.e., consisting of one single pixel.  
Each image is a point in one dimension.



# The curse of dimensionality

For simplicity, consider  $1 \times 1$  images, i.e., consisting of one single pixel.  
Each image is a point in one dimension.



Similarly, with 2 pixels we get:

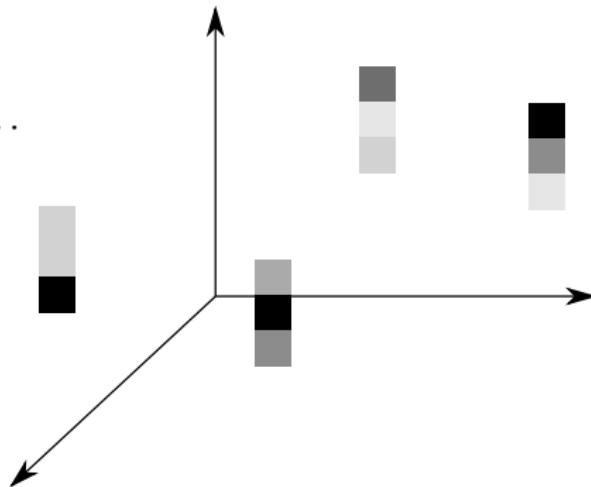


# The curse of dimensionality

For simplicity, consider  $1 \times 1$  images, i.e., consisting of one single pixel.  
Each image is a point in one dimension.



Similarly, with 3 pixels we get:



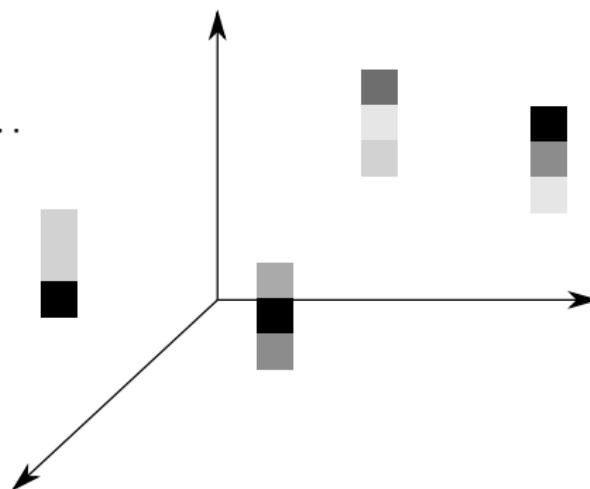
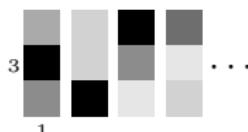
# The curse of dimensionality

For simplicity, consider  $1 \times 1$  images, i.e., consisting of one single pixel.

Each image is a point in one dimension.



Similarly, with 3 pixels we get:



Each new dimension increases **sparsity** of the point cloud.

## The curse of dimensionality

A dataset of natural images will be **extremely sparse** in  $\mathbb{R}^{w \times h}$ , since each region of space is **observed** very infrequently.

New samples are **less likely** to fall close to the previous ones.

## The curse of dimensionality

A dataset of natural images will be **extremely sparse** in  $\mathbb{R}^{w \times h}$ , since each region of space is **observed** very infrequently.

New samples are **less likely** to fall close to the previous ones.

To discover a pattern, we need **exponentially** many observations as we have dimensions!

## The curse of dimensionality

A dataset of natural images will be **extremely sparse** in  $\mathbb{R}^{w \times h}$ , since each region of space is **observed** very infrequently.

New samples are **less likely** to fall close to the previous ones.

To discover a pattern, we need **exponentially** many observations as we have dimensions!

If  $n$  data points cover well the space of 1-dimensional images, then  $n^d$  data points are required for  $d$ -dimensional images.

More data points make interesting structures emerge



## The curse of dimensionality

A dataset of natural images will be **extremely sparse** in  $\mathbb{R}^{w \times h}$ , since each region of space is **observed** very infrequently.

New samples are **less likely** to fall close to the previous ones.

To cover the space entirely, we need **exponentially** many observations as we have dimensions!

If  $n$  data points cover well the space of 1-dimensional images, then  $n^d$  data points are required for  $d$ -dimensional images.

Two options:

- ① Increase the dataset**
- ② Decrease the dimensions**

## Favor simplicity

Let's play a game:

2, 4, 8, . . .

Rules:

- **Task:** Discover the rule I used to produce the sequence
- Give me a number: I'll tell you if it's next in sequence or not
- **Once you're sure,** tell me the rule

# Favor simplicity

Let's play a game:

2, 4, 8, . . .

Rules:

- **Task:** Discover the rule I used to produce the sequence
- Give me a number: I'll tell you if it's next in sequence or not
- **Once you're sure,** tell me the rule

**Occam's razor:** Among competing hypotheses, select the one with the fewest assumptions.

# Favor simplicity

Let's play a game:

2, 4, 8, . . .

Rules:

- **Task:** Discover the rule I used to produce the sequence
- Give me a number: I'll tell you if it's next in sequence or not
- **Once you're sure,** tell me the rule

**Occam's razor:** Among competing hypotheses, select the one with the fewest assumptions.

Also: when feasible, add more data!

## Features

Assume each data point  $x \in \mathcal{D} \subset \mathbb{R}^n$  is the result of a synthesis process:

$$\sigma : F \mapsto x$$

which takes a set of **features**  $F$  and composes them to form  $x$ .

# Features

Assume each data point  $x \in \mathcal{D} \subset \mathbb{R}^n$  is the result of a synthesis process:

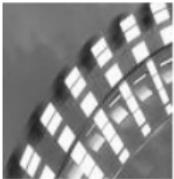
$$\sigma : F \mapsto x$$

which takes a set of **features**  $F$  and composes them to form  $x$ .

## Example

An image  $x \in \mathbb{R}^{w \times h}$  is composed by pixels.

If each pixel of  $x$  is a feature, then  $\sigma$  simply sums them up:


$$= \alpha_1 \begin{array}{|c|} \hline \cdot \\ \hline \end{array} + \alpha_2 \begin{array}{|c|} \hline \\ \hline \cdot \\ \hline \end{array} + \alpha_3 \begin{array}{|c|} \hline \\ \hline \\ \hline \cdot \\ \hline \end{array} + \dots$$

# Features

Assume each data point  $x \in \mathcal{D} \subset \mathbb{R}^n$  is the result of a synthesis process:

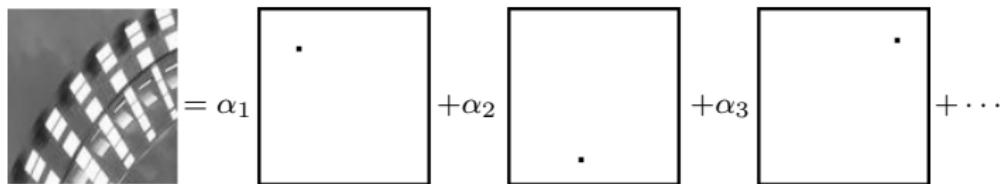
$$\sigma : F \mapsto x$$

which takes a set of **features**  $F$  and composes them to form  $x$ .

## Example

An image  $x \in \mathbb{R}^{w \times h}$  is composed by pixels.

If each pixel of  $x$  is a feature, then  $\sigma$  simply sums them up:


$$x = \alpha_1 \begin{matrix} \cdot \\ \boxed{\phantom{000}} \end{matrix} + \alpha_2 \begin{matrix} \cdot \\ \boxed{\phantom{000}} \end{matrix} + \alpha_3 \begin{matrix} \cdot \\ \boxed{\phantom{000}} \end{matrix} + \dots$$

In this case, the **feature space**  $F$  is spanned by individual pixels.

Each feature (each pixel) represents a dimension.

# Features

Assume each data point  $x \in \mathcal{D} \subset \mathbb{R}^n$  is the result of a synthesis process:

$$\sigma : F \mapsto x$$

which takes a set of **features**  $F$  and composes them to form  $x$ .

## Example

An image  $x \in \mathbb{R}^{w \times h}$  is composed by pixels.

If each pixel of  $x$  is a feature, then  $\sigma$  simply sums them up:

$$x = \sigma(F) = \sum_{f_i \in F} \alpha_i \cdot f_i$$

$\alpha_i$  are the **weights** in the representation of  $x$ .

# Features

Assume each data point  $x \in \mathcal{D} \subset \mathbb{R}^n$  is the result of a synthesis process:

$$\sigma : F \mapsto x$$

which takes a set of **features**  $F$  and composes them to form  $x$ .

## Example

An image  $x \in \mathbb{R}^{w \times h}$  is composed by pixels.

If each pixel of  $x$  is a feature, then  $\sigma$  simply sums them up:

$$x = \sigma(F) = \sum_{f_i \in F} \alpha_i \cdot f_i$$

$\alpha_i$  are the **weights** in the representation of  $x$ .

In this **particular case**, the feature space is a **vector space** and  $\sigma$  is **linear**.

# Features

Having one feature per pixel is extremely wasteful!

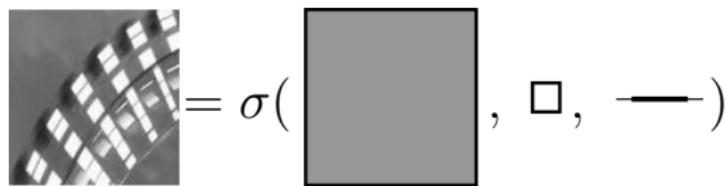
Curse of dimensionality: **features  $\gg$  observations**

# Features

Having one feature per pixel is extremely wasteful!

Curse of dimensionality: **features  $\gg$  observations**

What does really characterize our image?


$$\text{Image} = \sigma(\text{Large Gray Box}, \square, \text{---})$$

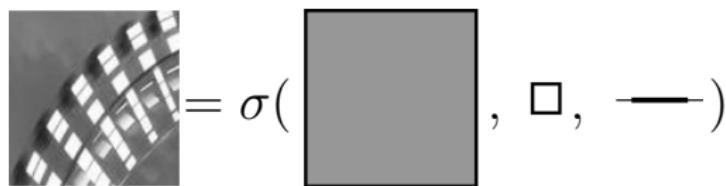
In general, the transformation  $\sigma$  acts **nonlinearly** on the features.

# Features

Having one feature per pixel is extremely wasteful!

Curse of dimensionality: **features  $\gg$  observations**

What does really characterize our image?



A grayscale image of a checkered surface is shown on the left, followed by an equals sign. To the right of the equals sign is the mathematical expression  $= \sigma($ , which is followed by three icons: a gray square, a white square with a black border, and a horizontal line.

In general, the transformation  $\sigma$  acts **nonlinearly** on the features.

The output of  $\sigma$  is called an **embedding** of the data point.

For the data point  $x \in \mathcal{D} \subset \mathbb{R}^n$ , the **embedding space** is  $\mathbb{R}^n$ .

## Invariances

In general, a given data point admits **many possible embeddings**.

# Invariances

In general, a given data point admits **many possible embeddings**.

Example: A sheet lives naturally in  $\mathbb{R}^2$



## Invariances

In general, a given data point admits **many possible embeddings**.

Example: A sheet lives naturally in  $\mathbb{R}^2$ , but is usually embedded in  $\mathbb{R}^3$ .

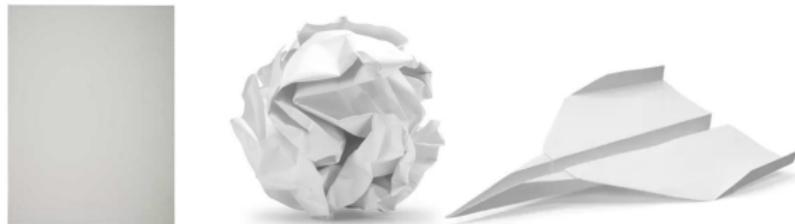


Three different embeddings of the **same** object

# Invariances

In general, a given data point admits **many possible embeddings**.

Example: A sheet lives naturally in  $\mathbb{R}^2$ , but is usually embedded in  $\mathbb{R}^3$ .



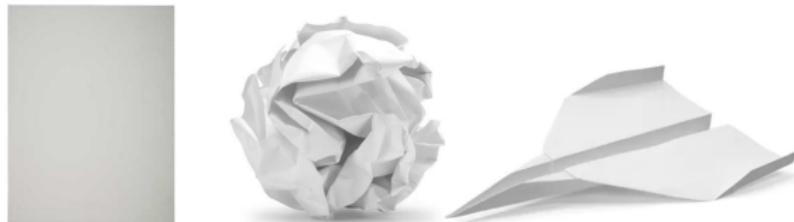
Three different embeddings of the **same** object

Q: what is preserved in all these embeddings?

# Invariances

In general, a given data point admits **many possible embeddings**.

Example: A sheet lives naturally in  $\mathbb{R}^2$ , but is usually embedded in  $\mathbb{R}^3$ .



Three different embeddings of the **same** object

A: **distances** are preserved in all the embeddings

## Invariances

In general, a given data point admits **many possible embeddings**.

Example: A sheet lives naturally in  $\mathbb{R}^2$ , but is usually embedded in  $\mathbb{R}^3$ .



Three different embeddings of the **same** object

A: **distances** are preserved in all the embeddings

Challenge: discover what **intrinsic** properties are preserved; these properties characterize the data.

# Latent features

In the general case:

- Features are not necessarily **localized** in space
- Features are not necessarily **evident** in the embedding

## Latent features

In the general case:

- Features are not necessarily **localized** in space
- Features are not necessarily **evident** in the embedding

We talk about **latent** features. Direct access to the embedding only.

# Latent features

In the general case:

- Features are not necessarily **localized** in space
- Features are not necessarily **evident** in the embedding

We talk about **latent** features. Direct access to the embedding only.

Example



# Latent features

In the general case:

- Features are not necessarily **localized** in space
- Features are not necessarily **evident** in the embedding

We talk about **latent** features. Direct access to the embedding only.

Example



Latent feature: directional illumination

# Latent features

In the general case:

- Features are not necessarily **localized** in space
- Features are not necessarily **evident** in the embedding

We talk about **latent** features. Direct access to the embedding only.

Example



Latent feature: directional illumination

**3** params for the light source position + **1** param for light intensity

# Latent features

In the general case:

- Features are not necessarily **localized** in space
- Features are not necessarily **evident** in the embedding

We talk about **latent** features. Direct access to the embedding only.

Discovering latent features involves discovering:

the “true” embedding space for the data

+

the **transformation** between the two spaces

General idea: find & discard **non-informative** dimensions.

## Optimal dimensionality

Even just discovering the intrinsic **dimensionality** is a challenge by itself.

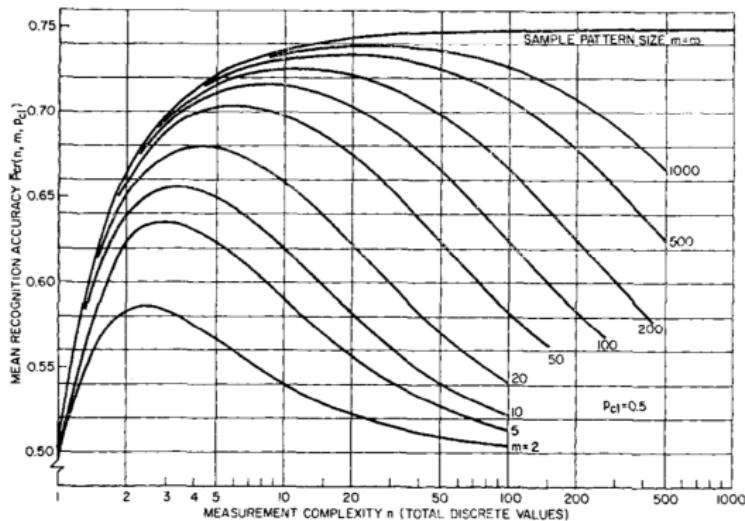
Usually, it is specified by hand by whoever designs the learning model.

# Optimal dimensionality

Even just discovering the intrinsic **dimensionality** is a challenge by itself.

Usually, it is specified by hand by whoever designs the learning model.

The effect of different dimensions, captured by **Hughes phenomenon**:



There is an optimal dimension which maximizes accuracy.

Hughes, "On the mean accuracy of statistical pattern recognizers", IEEE TIT 1968

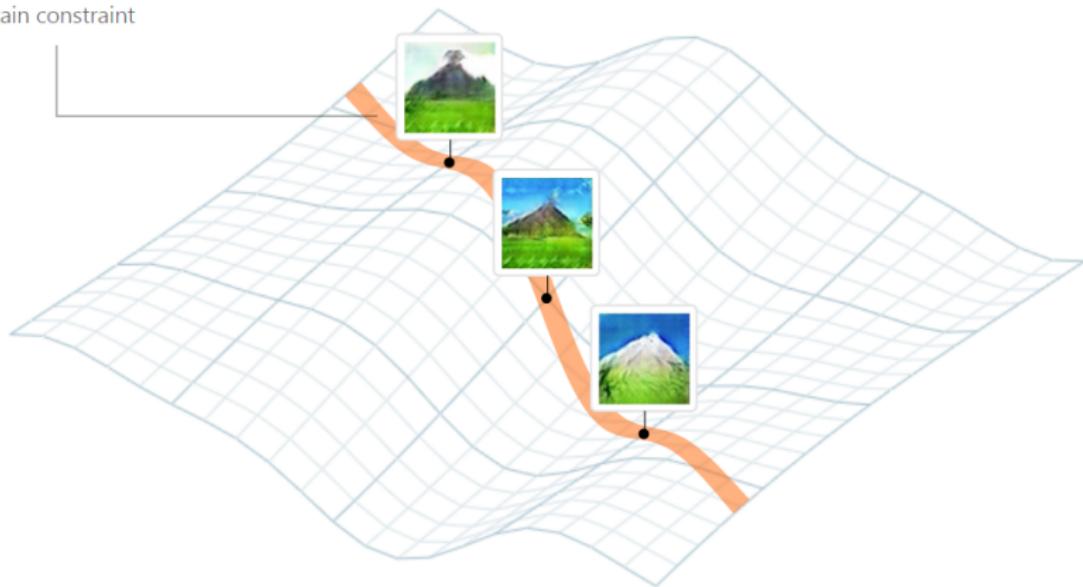
## The manifold hypothesis

Deep learning assumes that the input data lives on some underlying non-Euclidean structure called a [manifold](#).

# The manifold hypothesis

Deep learning assumes that the input data lives on some underlying non-Euclidean structure called a **manifold**.

Subspace of all images  
that satisfy the  
mountain constraint



# Features are task-driven

How are features extracted from given data?

Speaking about features only makes sense if we are given a **task** to solve!

# Features are task-driven

How are features extracted from given data?

Speaking about features only makes sense if we are given a **task** to solve!



Is color important?

# Features are task-driven

How are features extracted from given data?

Speaking about features only makes sense if we are given a [task](#) to solve!



Is color important?

Rank, suit, and color are generic features, but [specific problems](#) determine what features are important for that task.

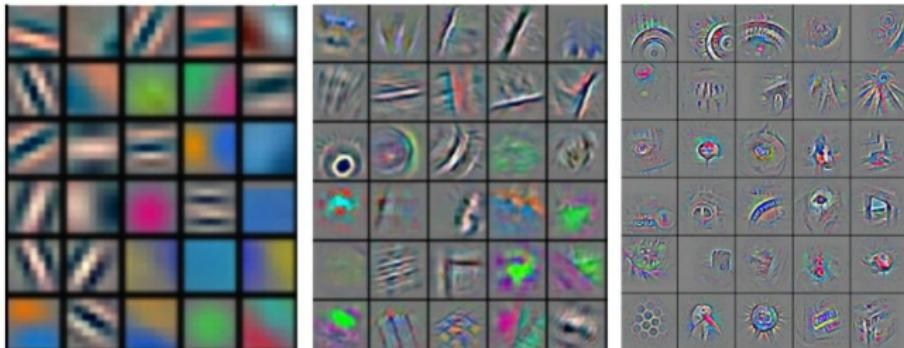
What counts in spades, does not count in poker.

Deep learning is a **task-driven** paradigm to extract patterns and **latent features** from given observations

Deep learning is a **task-driven** paradigm to extract patterns and **latent features** from given observations

However, features are not always the focus of deep learning; rather, they are instrumental for the given task and drive the decision.

Example: Visual classification



## Suggested reading

Blog post on the datasaurus:

<https://www.autodeskresearch.com/publications/samestats>

TED talk on the idea behind imaging the black hole:

<https://www.youtube.com/watch?v=BIvezCVcsYs>

VLBI reconstruction dataset:

<http://vlbiimaging.csail.mit.edu/>

Paper on the black hole imaging technique:

<https://arxiv.org/pdf/1512.01413.pdf>

Tutorial video and slides on ML fairness:

<https://nips.cc/Conferences/2017/Schedule?showEvent=8734>

Distill post on t-SNE:

<https://distill.pub/2016/misread-tsne/>