# DD2437 – Artificial Neural Networks and Deep Architectures (annda)

## Lecture 10: **Representation learning**

Pawel Herman

Computational Science and Technology (CST)

KTH Royal Institute of Technology

# Why do we think deep learning is powerful?

- What is the motivation & inspiration for deep network architectures?
  - expressive power (*expressibility*) and compactness (*efficiency*) – exponential gain
  - *"cheap learning"*, *"no-flattening"* theorem, hierarchical struct (Lin & Tegmark)
  - hierarchical brain (cortex) organisation
  - multiple levels of abstraction

# Why do we think deep learning is powerful?

- What is the motivation & inspiration for deep network architectures?
  - expressive power (*expressibility*) and compactness (*efficiency*) – exponential gain
  - *"cheap learning"*, *"no-flattening"* theorem, hierarchical struct (Lin & Tegmark)
  - hierarchical brain (cortex) organisation
  - multiple levels of abstraction
- Algorithmic development
  - advanced regularisation techniques
  - hyperparameter selection schemes
- Computational resources and increasing availability of large data sets

# Learning representations as a hallmark of DL

Learning features as part of DL modus operandi – deep neural networks extract representations from data)

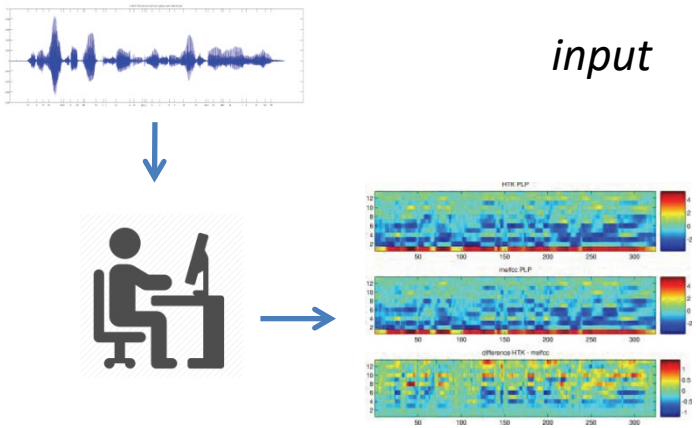
# Learning representations as a hallmark of DL

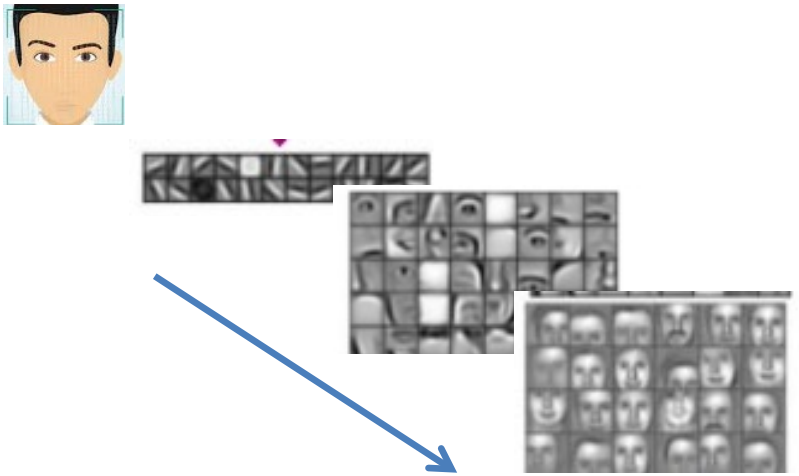
Learning features as part of DL modus operandi

- Traditional pattern recognition *VS* deep neural network approach

Hand-engineered features in a traditional pattern recognition approach

End-to-end networks with learned features spaces, data representations

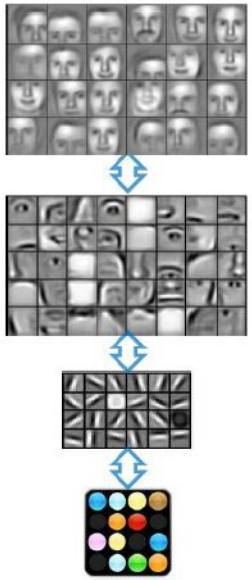


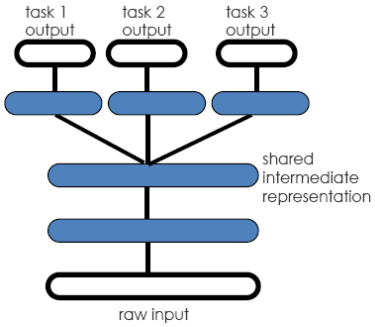
# Learning representations as a hallmark of DL

Learning features as part of DL modus operandi with many implications.....

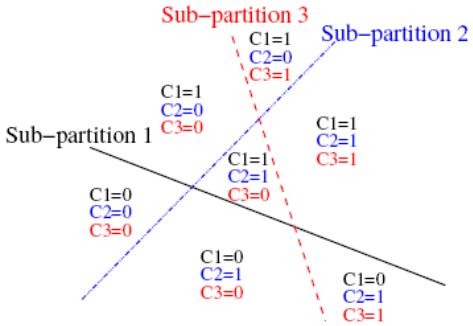
hierarchy of abstraction levels



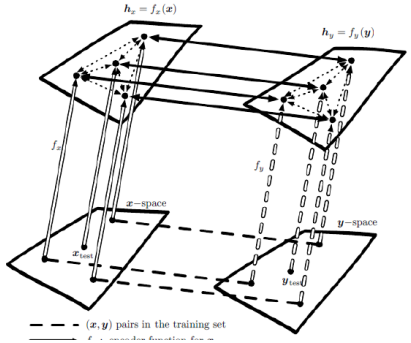

multi-tasking and transfer learning




multi-clustering

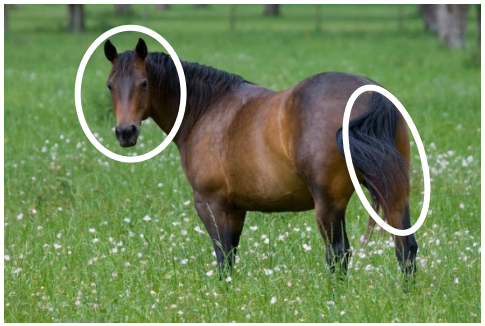



zero-shot learning



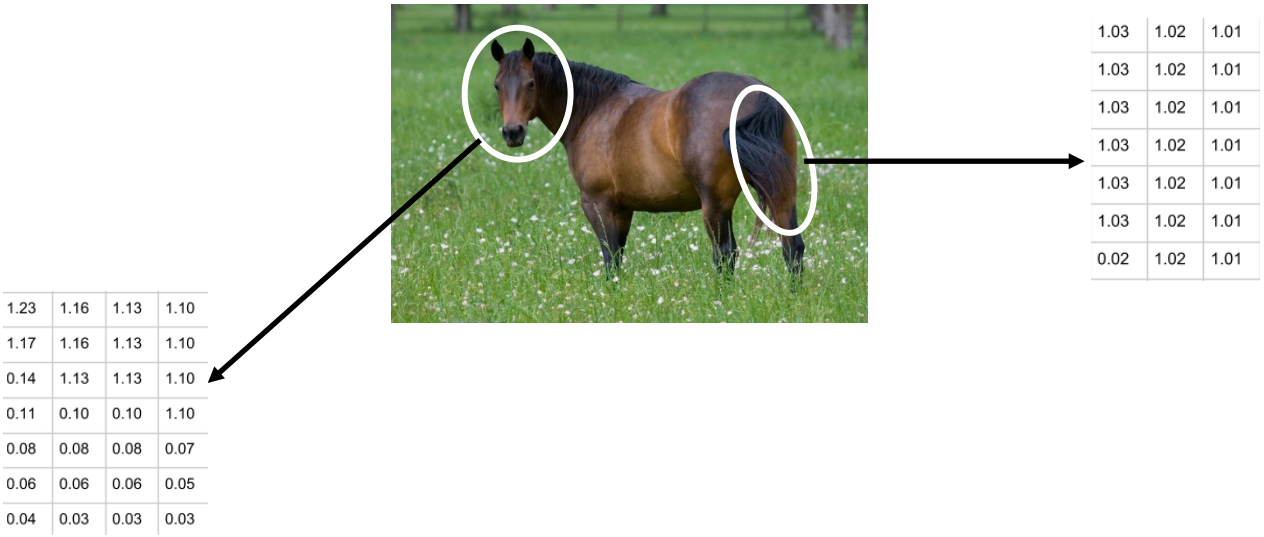
# Data representations

- Multiple ways of representing information – what is the difference? Why should we care?



# Data representations

- Multiple ways of representing information – what is the difference? Why should we care?

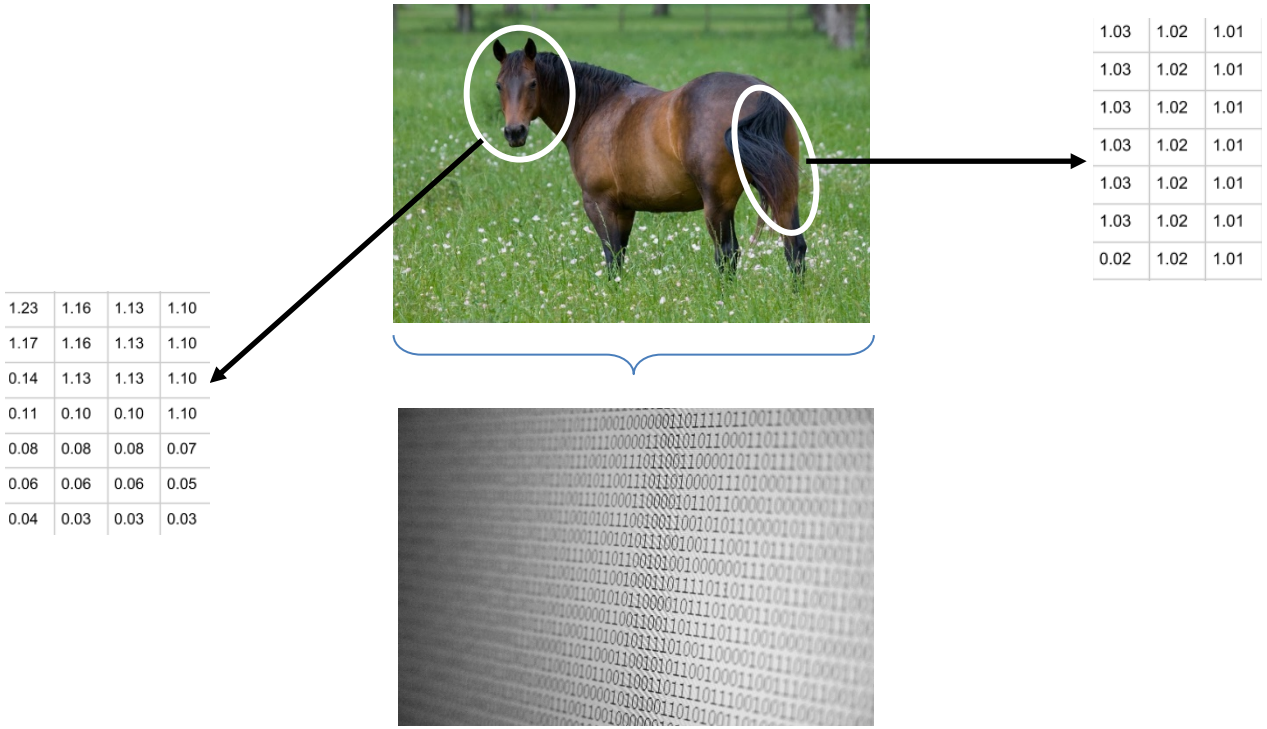


| | | | |
|---|---|---|---|
| 1.23 | 1.16 | 1.13 | 1.10 |
| 1.17 | 1.16 | 1.13 | 1.10 |
| 0.14 | 1.13 | 1.13 | 1.10 |
| 0.11 | 0.10 | 0.10 | 1.10 |
| 0.08 | 0.08 | 0.08 | 0.07 |
| 0.06 | 0.06 | 0.06 | 0.05 |
| 0.04 | 0.03 | 0.03 | 0.03 |

| | | |
|---|---|---|
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 0.02 | 1.02 | 1.01 |

# Data representations

- Multiple ways of representing information – what is the difference? Why should we care?

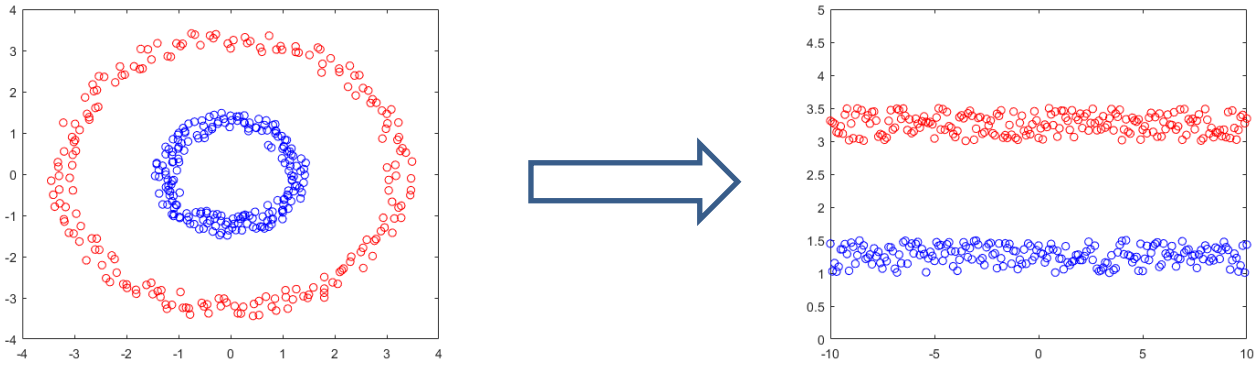


| | | | |
|---|---|---|---|
| 1.23 | 1.16 | 1.13 | 1.10 |
| 1.17 | 1.16 | 1.13 | 1.10 |
| 0.14 | 1.13 | 1.13 | 1.10 |
| 0.11 | 0.10 | 0.10 | 1.10 |
| 0.08 | 0.08 | 0.08 | 0.07 |
| 0.06 | 0.06 | 0.06 | 0.05 |
| 0.04 | 0.03 | 0.03 | 0.03 |

| | | |
|---|---|---|
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 1.03 | 1.02 | 1.01 |
| 0.02 | 1.02 | 1.01 |

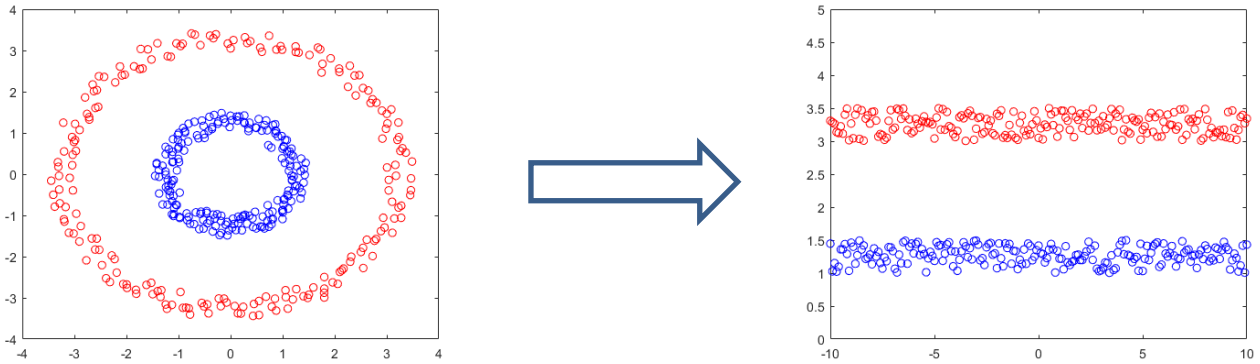
# Data representations

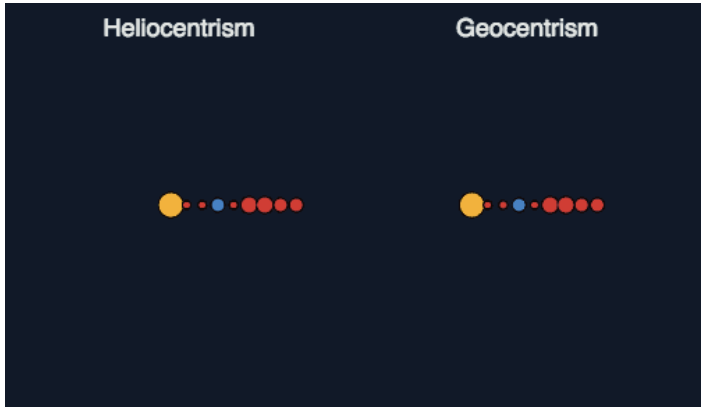
- Multiple ways of representing information – what is the difference? Why should we care?



# Data representations

- Multiple ways of representing information – what is the difference? Why should we care?





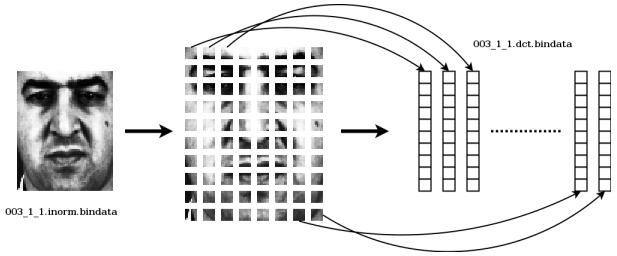
# Data representations

- Multiple ways of representing information – what is the difference?

  Different *data parameterisations*

12102 → 101111101000110 (bin)

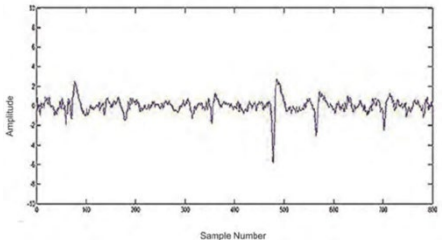
12102 → 2f46 (hex)

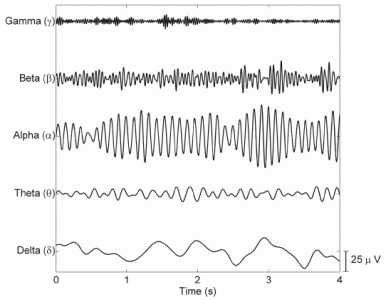


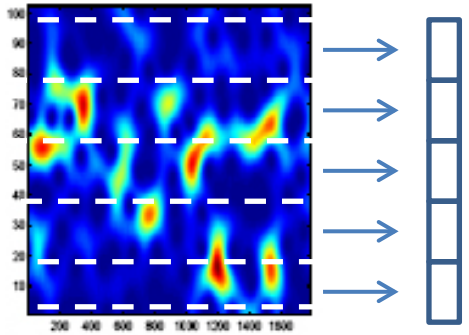
# Data representations

- Multiple ways of representing information – features

  From low-level data description to higher-order representations



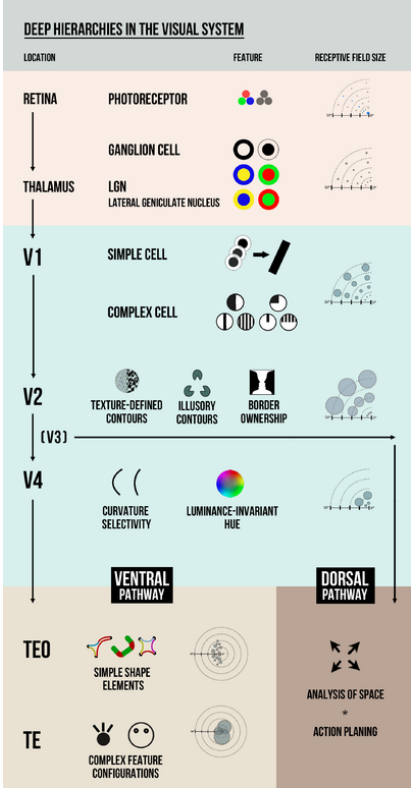



feature vector

How are attributes/features determined, extracted?

# Representations in the brain as an inspiration

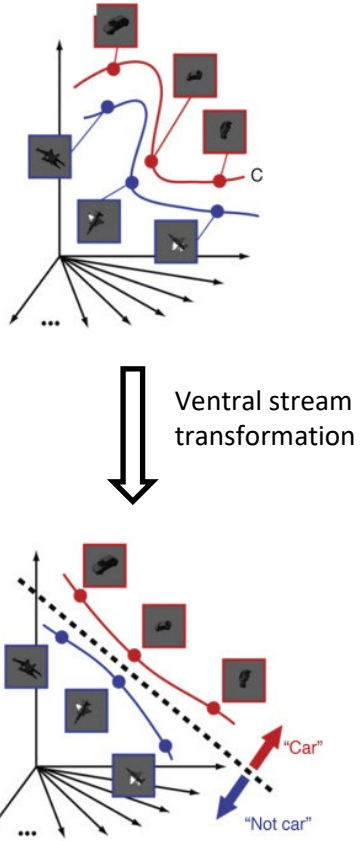
- Sensory information represented by neural activity
  - neurons with different response properties (selectivity, tuning curves)
  - *distributed* nature of neural representations in populations vs grandmother cell concept
  - *sparseness* (energy efficient), redundancy
- Hierarchical representations
  - sensory pathways are organised into *hierarchies*
  - hierarchy of *abstraction* levels




Wikibooks

# Representations in the brain as an inspiration

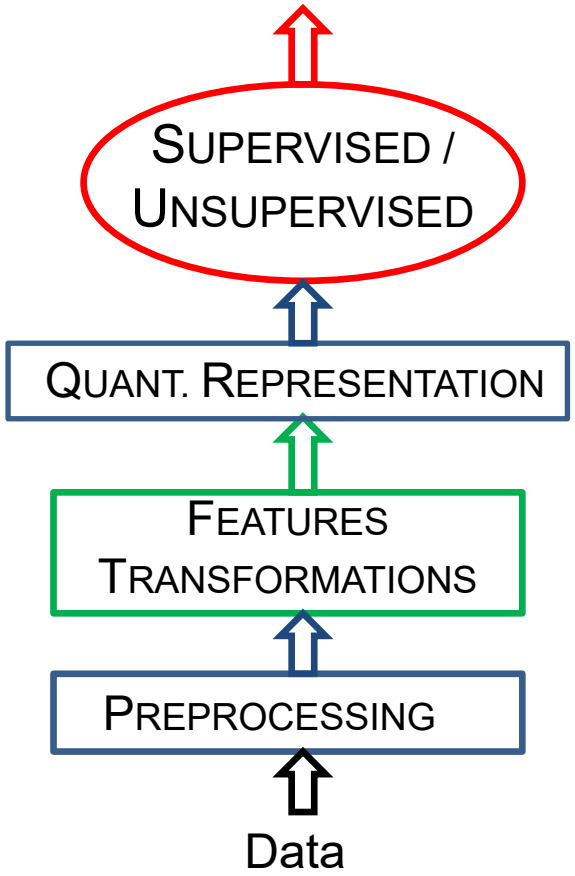
- Sensory information represented by neural activity
  - neurons with different response properties (selectivity, tuning curves)
  - *distributed* nature of neural representations in populations vs grandmother cell concept
  - *sparseness* (energy efficient), redundancy
- Hierarchical representations
  - sensory pathways are organised into *hierarchies*
  - hierarchy of *abstraction* levels
  - ***"untangling"* property of the ventral stream**




adapted from DiCarlo et al., 2012

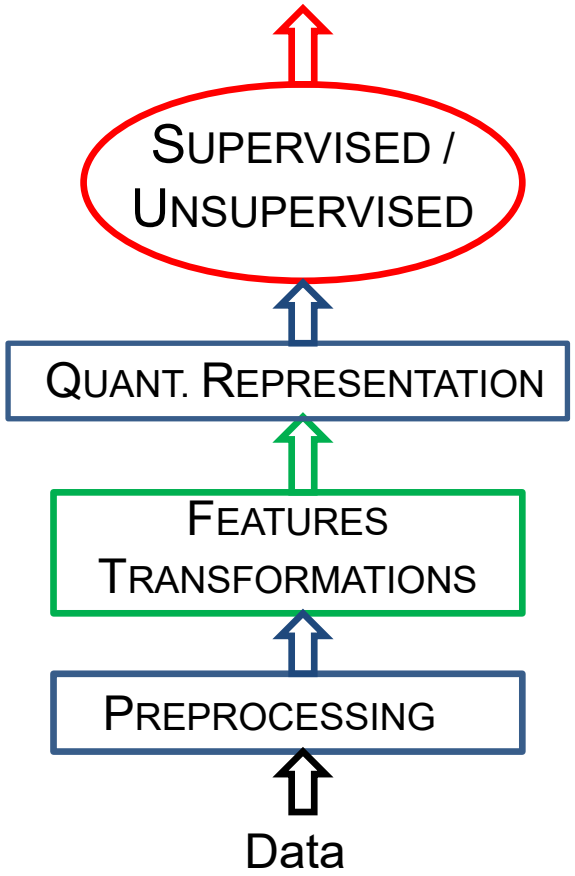
# Representation learning problem

- Importance in the machine learning or pattern recognition context
  - belief that AI can be built on representation learning & complex reasoning (which we still lack)



# Representation learning problem

- Importance in the machine learning or pattern recognition context
- Trade-off between minimising “information” loss and obtaining “nice” properties



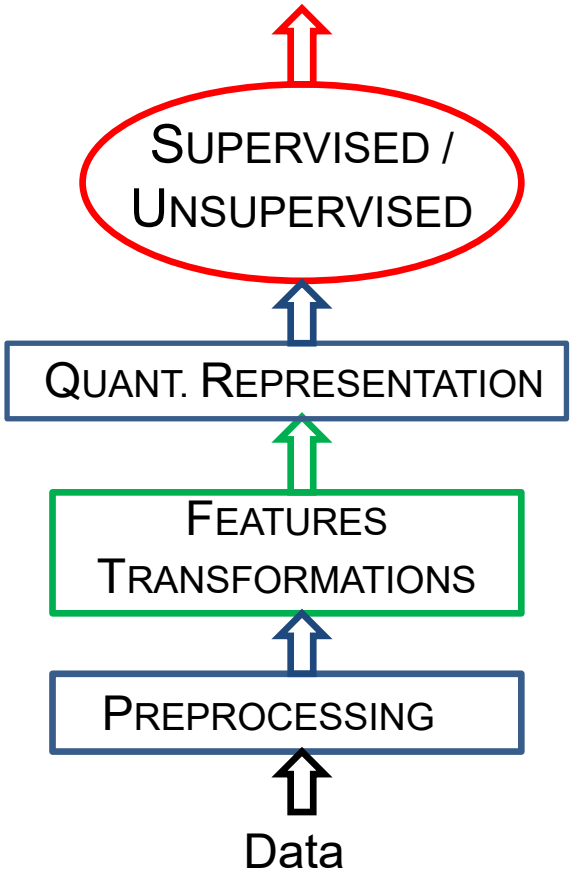
# Representation learning problem

- Importance in the machine learning or pattern recognition context
- Trade-off between minimising "information" loss and obtaining "nice" properties
- What makes representation good? What is desirable/useful information?

factors of variation, causal variables

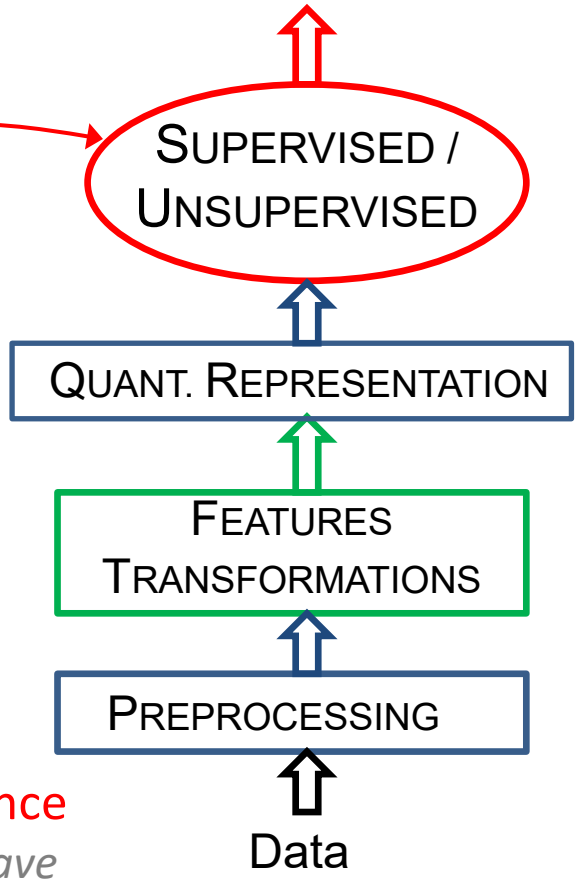
information about key probabilistic features of data distribution

information needed to solve a given task



# Representation learning problem

- Importance in the machine learning or pattern recognition context
- Trade-off between minimising "information" loss and obtaining "nice" properties
- What makes representation good? What is desirable/useful information?

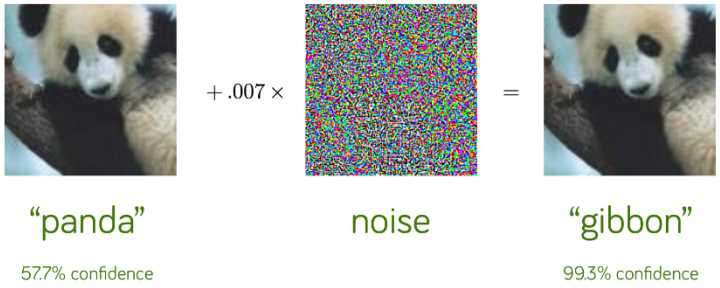



Facilitate the subsequent learning task, maximise performance

(*easiest to define for supervised learning problems but does not have to lead to "good" representations*)

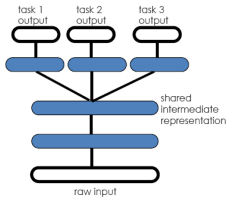
# Supervised vs unsupervised approach

- Supervised – distilling information relevant to a concrete task defined by labels
  - very useful when solving particular tasks
  - strongly relying on the abundance of labelled data and prone to excessive information bottleneck



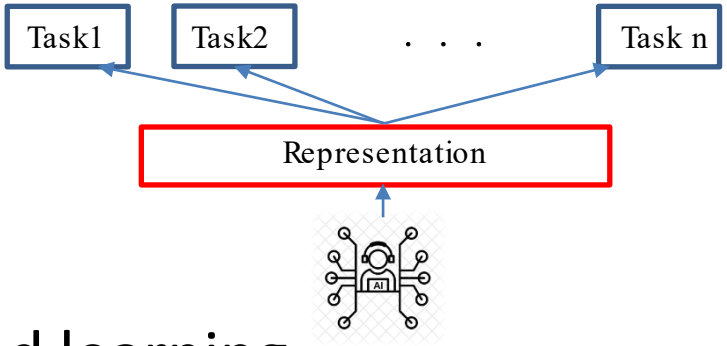

  - lacking “common sense”

  - different ways to “improve” representations



# Supervised vs unsupervised approach

- Supervised – distilling information relevant to a concrete task defined by labels
- Unsupervised learning
  - "The idea of learning to represent the world before learning a task – this is what babies do" (*LeCun*)
  - It appears as a more generic approach less susceptible to inf. Loss




- Semi-supervised learning

# Representation learning

- Computational perspective: disentangling unknown factors causing relevant variation in the data (*factors of variation*)
  - *causes explain* the observed data (discriminative context, both unsupervised and supervised aspects)

# Representation learning

- Computational perspective: disentangling unknown factors causing relevant variation in the data (*factors of variation*)
  - *causes explain* the observed data (discriminative context, both unsupervised and supervised aspects)

Representation learning should strive towards uncovering latent factors, $\boldsymbol{h}$, which capture underlying causes in $\boldsymbol{x}$.

Then, if $\boldsymbol{y}$ is one of them, i.e. $\boldsymbol{y}=h_i$, it should be easy to learn to predict $\boldsymbol{y}$ from this representation.

$$p(\boldsymbol{h}|\boldsymbol{x})=p(\boldsymbol{x}|\boldsymbol{h})\,p(\boldsymbol{h}) \qquad \text{Ideally: } p(\boldsymbol{h}) = \prod_i p(h_i)$$

# Representation learning

- Computational perspective: disentangling unknown factors causing relevant variation in the data (*factors of variation*)
  - *causes explain* the observed data (discriminative context, both unsupervised and supervised aspects)
  - factors in combination can be used to generate data (generative context)
- Probabilistic perspective
  - Classical approach: *density estimation* – learn probability distribution for data with the use of latent variables (PCA, ICA, GMM etc.) -> explain data
  - P(*data*|*latent var*) for generation and P(*latent var*|*data*) for recognition
  - full probabilistic model advocated by generative models, P(*data*,…)

# Representation learning

- Computational perspective: disentangling unknown factors causing relevant variation in the data (*factors of variation*)
  - *causes explain* the observed data (discriminative context, both unsupervised and supervised aspects)
  - factors in co… …erative context)
- Probabilistic…
  - Classical a… …istribution for data with … -> explain data
  - P(*data*|*lat*… …r recognition
  - full probabilistic model advocated by generative models, P(*data*,…)

Can we implicitly guide the u*nsupervised* learning to discover features corresponding to underlying/causal factors?

# Strategies to guide discovery of salient/causal factors

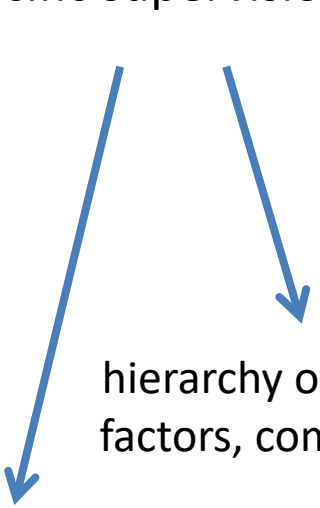
How can the discovery/identification of the underlying causal factors of variation that generate the data be further supported?

(beyond application specific supervision signal)

# Generic regularisation strategies (Bengio et al., 2013)

How can the discovery/identification of the underlying causal factors of variation that generate the data be further supported?

(beyond application specific supervision signal)



hierarchy of explanatory factors, compositionality

multiple explanatory (and ideally, *causal*) factors

(semisupervised/unsupervised pretraining + distributed representations to untangle separate factors of variation in the representation space)

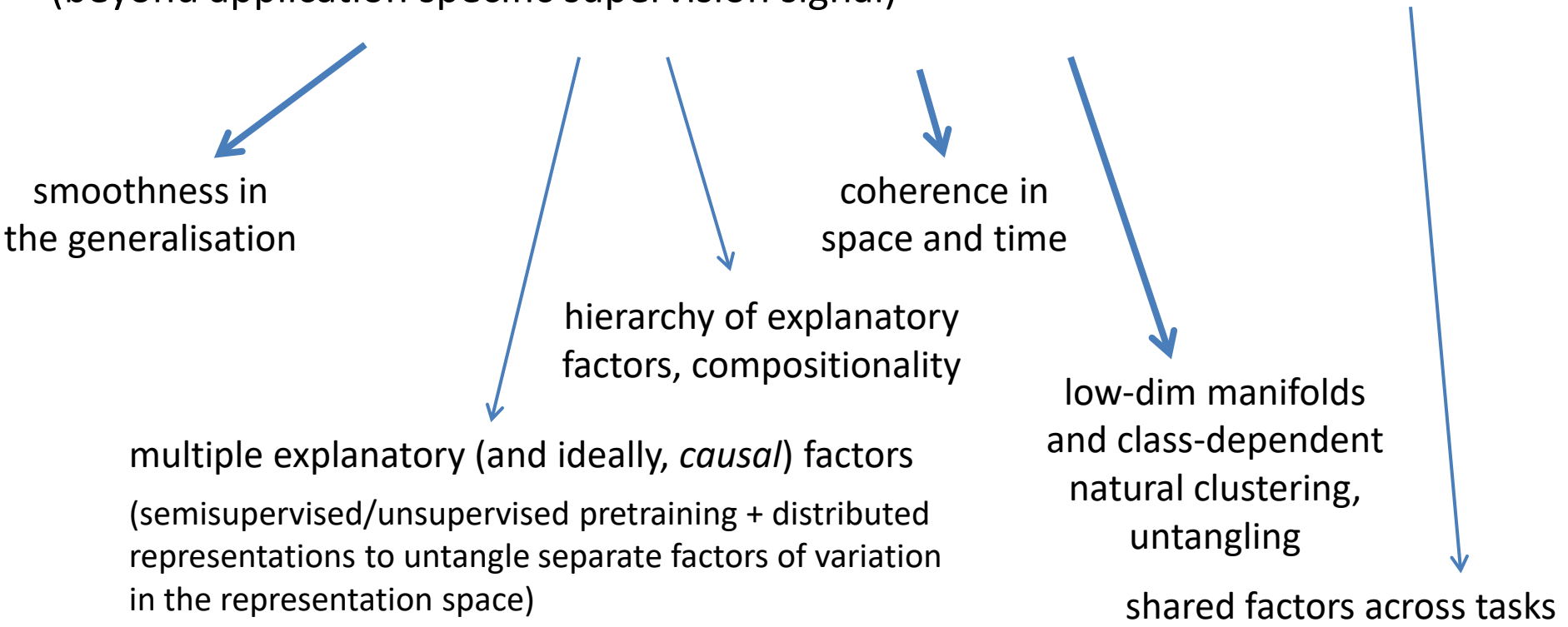
Transfer learning, multi-task learning

shared factors across tasks

# Generic regularisation strategies (Bengio et al., 2013)

How can the discovery/identification of the underlying causal factors of variation that generate the data be further supported?

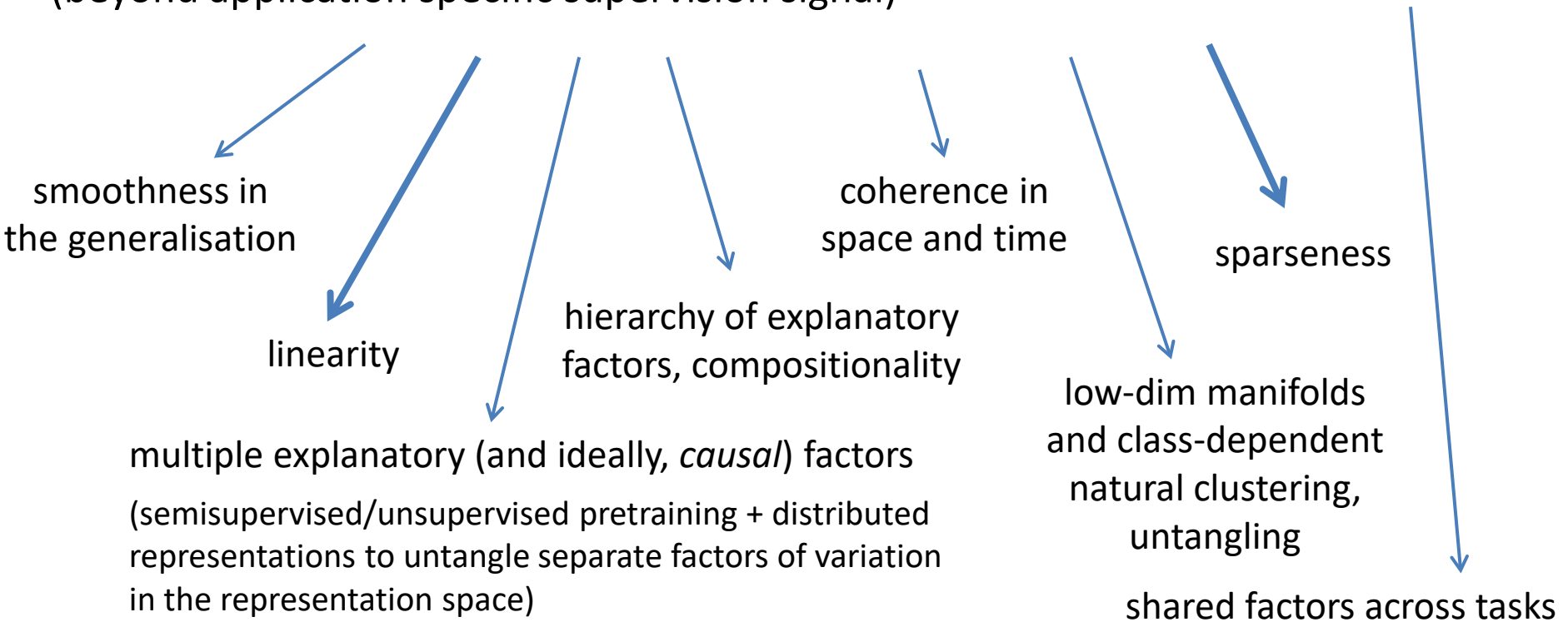
(beyond application specific supervision signal)



# Generic regularisation strategies (Bengio et al., 2013)

How can the discovery/identification of the underlying causal factors of variation that generate the data be further supported?

(beyond application specific supervision signal)



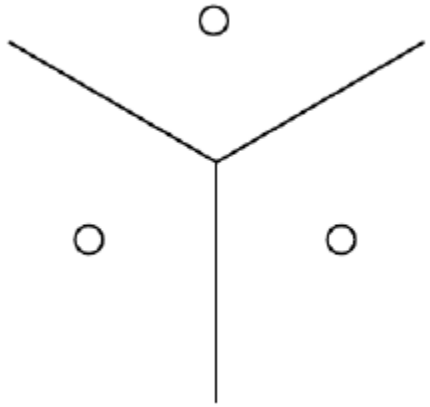
# Distributed representations

Information is distributed across many units that account for information about features that are not mutually exclusive…..

# Distributed vs local (prototype) representations

Information is distributed across many units that account for information about features that are not mutually exclusive…..

… unlike in clustering (cluster centres act as prototypes) with distinct regions where *local generalisation* is observed.

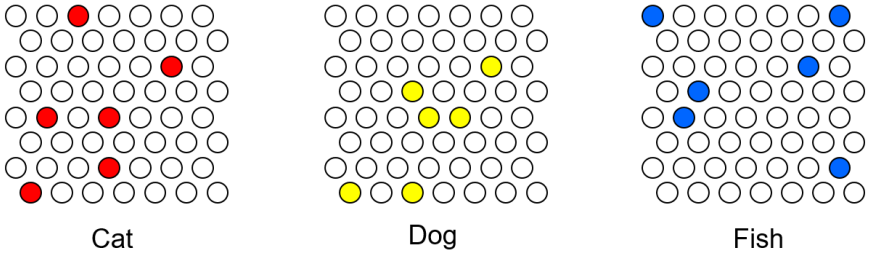


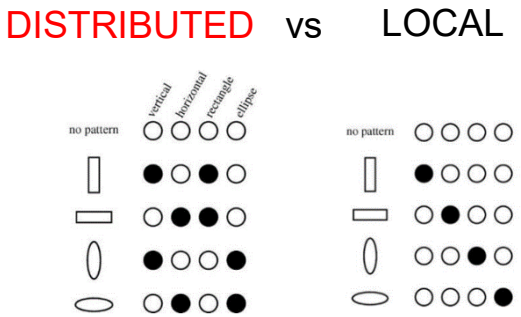
# Distributed vs local representations

Information is distributed across many units that account for information about features that are not mutually exclusive…..

… unlike in clustering (cluster centres act as prototypes) with distinct regions where *local generalisation* is observed.

Locality in input space implies different behaviour of the learned function in different regions of data space (local or symbolic representations).




Generalisation due to shared attributes and semantic proximity.

# Distributed vs local representations

Information is distributed across many units that account for information about features that are not mutually exclusive.....

But learning distributed representations can be global (e.g. PCA, ICA) or local (many manifold learning algorithms)

different regions of data space (local or symbolic representations).







Generalisation due to shared attributes and semantic proximity.

# Distributed vs local representations

Information is distributed across many units that account for information about features that are not mutually exclusive.....

Philosophically different approach from standard symbolic representations used in early AI approaches

There are many advantages and functional implications of *distributed* (and *sparse*) representations despite their typically slow learning





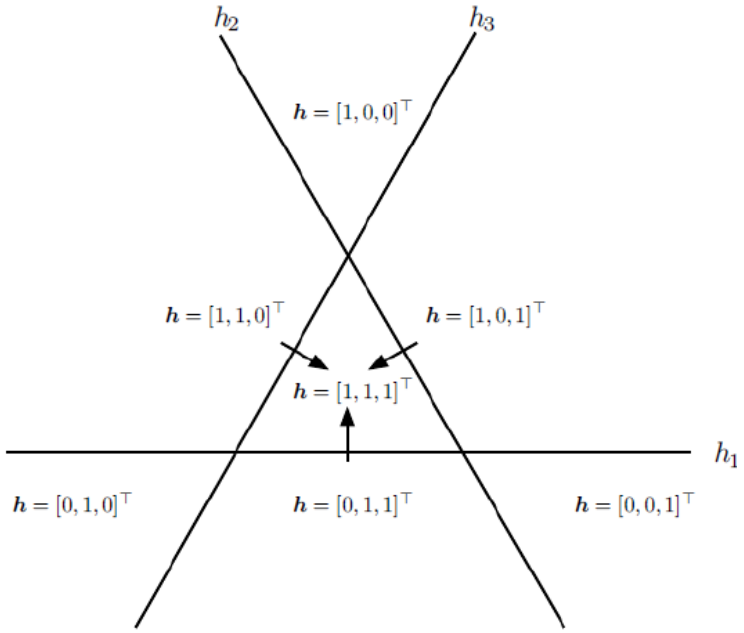

Generalisation due to shared attributes and semantic proximity.

# The power of distributed representations

Desirable characteristics, advantageous properties

- expressiveness (*n* features with *k* values each can describe $k^n$ concepts)
- the combination of powerful distributed representations with weak classifiers could be a strong regulariser

fault tolerance



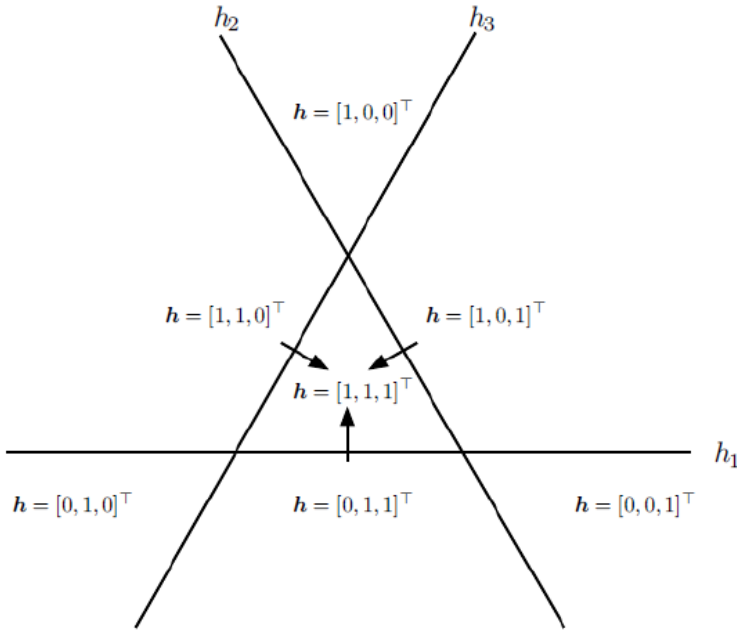

Goodfellow et al.
Bengio et al., 2009

# The power of distributed representations

Desirable characteristics, advantageous properties

- expressiveness (*n* features with *k* values each can describe $k^n$ concepts)
- the combination of powerful distributed representations with weak classifiers could be a strong regulariser
- similarity (topological) space with a distributed code – semantically close objects are close in distance
- generalisation due to shared attributes

content addressability

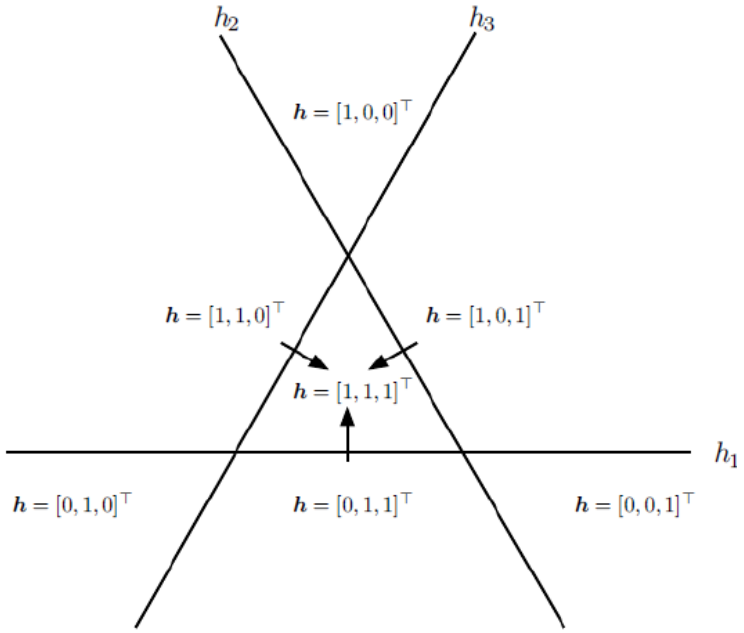



Goodfellow et al.
Bengio et al., 2009

# The power of distributed representations

Desirable characteristics, advantageous properties

- expressiveness (*n* features with *k* values each can describe $k^n$ concepts)
- the combination of powerful distributed representations with weak classifiers could be a strong regulariser
- similarity (topological) space with a distributed code – semantically close objects are close in distance
- generalisation due to shared attributes
- in line with the idea that hidden units can learn to represent the underlying causal factors as different variables (here: directions in the representation space)

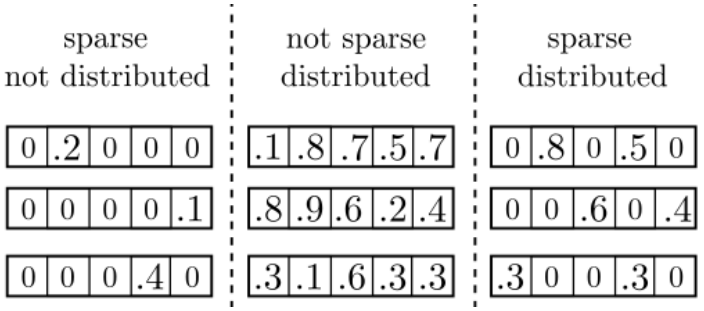



Goodfellow et al.
Bengio et al., 2009

# Sparse vs dense representations

## Sparse representations

- orthogonalisation/decorrelation – more separable in high-dimensional spaces
- "metabolic" efficiency
- neural selectivity (vs coarse coding with broad tuning)
- balance between sparse local representations that suffer from the *curse of dimensionality* and dense representations that *entangle* factors and can be hard to interpret



# Discussion

1) How could we functionally leverage distributed representations?
2) Design a DL system that forms or classifies multi-modal perceptual memories, e.g. images and sounds. What could be a computational basis of multi-modal processing?
3) Why would you expect a DNN trained to map the same inputs to different sets of labels (multi-tasking) to learn “better” representations?
4) What is your intuitive take on the quality of representations? What characteristics a “good” code should have?