A (pre)sheaf for distributed learning

David Balduzzi

Max Planck Institute for Intelligent Systems

July 6, 2012

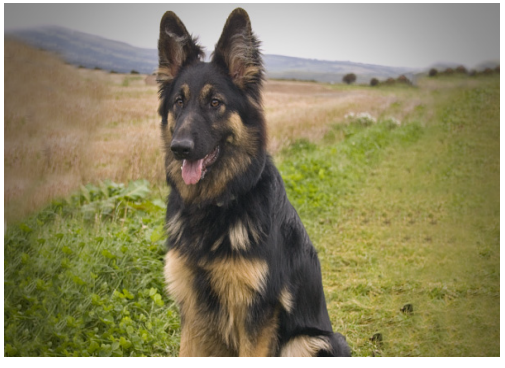
1. Motivation

Learning to classify

x_1 cat

x_2 cat

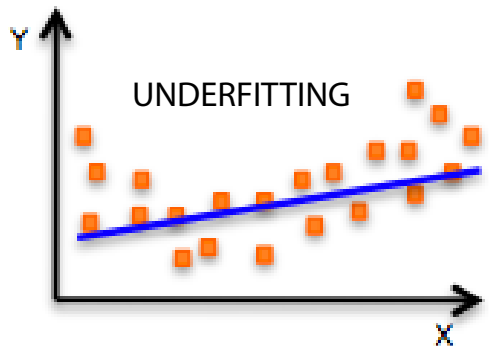
x_3 dog

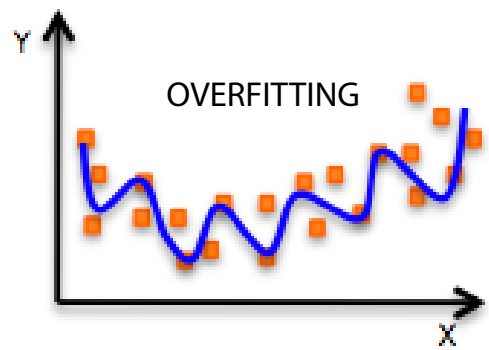


Find a classifier **guaranteed to perform well** on **future data** based on finite training sample.

The bias/variance dilemma

1. **Bias**
   - Simple learning algorithms generalize well (they capture the "gist")
   - if they fit the data (e.g. most data is not linear).
2. **Variance**
   - More complex algorithms perform better on training data (since they look for solutions in a larger search space)
   - but they tend to generalize poorly (memorize noise, see patterns that aren't there).



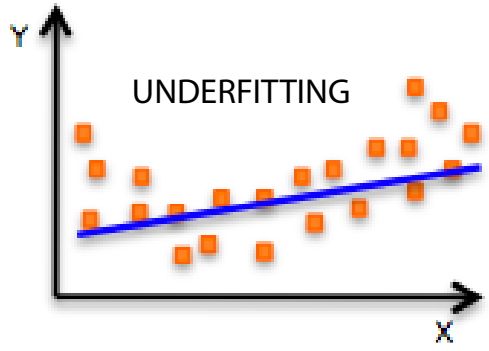


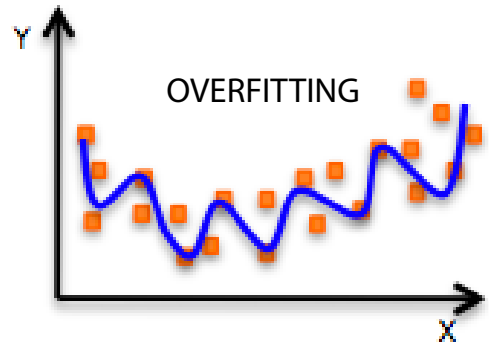
The bias/variance dilemma

Theorem (. . . many variants . . .)

With probability at least $1-\delta$,

$$\underbrace{\mathbf{R}(f)}_{\text{future error}} \leq \underbrace{\hat{\mathbf{R}}(f)}_{\text{training error}} + c_1 \underbrace{\sqrt{\frac{VC\text{-}entropy(\mathcal{F},\mathcal{D})}{\ell}}}_{\text{capacity of } \mathcal{F}} + c_2 \underbrace{\sqrt{\frac{1-\log\delta}{\ell}}}_{\text{confidence}}.$$





Can we find ways to combine learners that increase power, but not complexity?

The human brain

1. consists of $\pm 10^8$ neurons which can
2. learn an enormous variety of tasks
3. (that cannot have been genetically hardwired),
4. largely without overfitting.

It combines low bias (learns many tasks) with low variance (does not overfit).

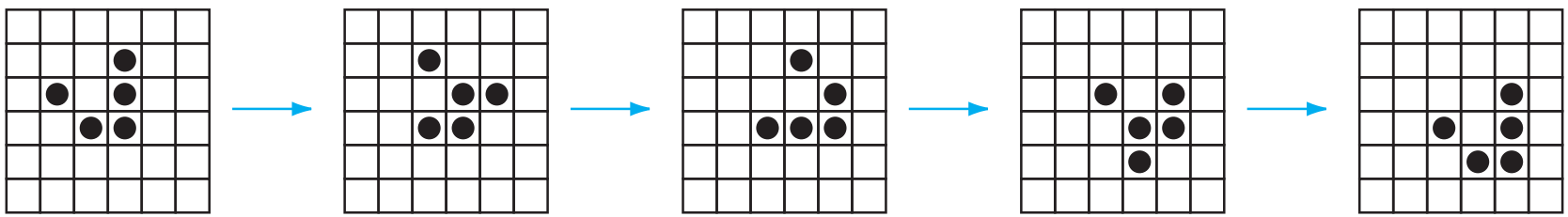
2. Computations and dependencies

Conway's Game of Life

"Atomic physics": A cell outputs 1 at time t iff

- 3 neighbors outputted 1s at $t-1$ or
- the cell and 2 neighbors outputted 1s at $t-1$,

otherwise it outputs 0.

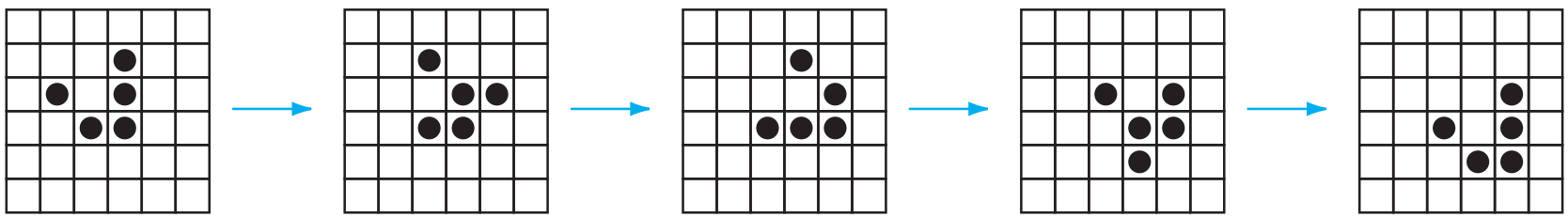


GLIDER

Conway's Game of Life

An infinitely large grid can be initialized to form a universal Turing machine, by hierarchically combining computational processes...

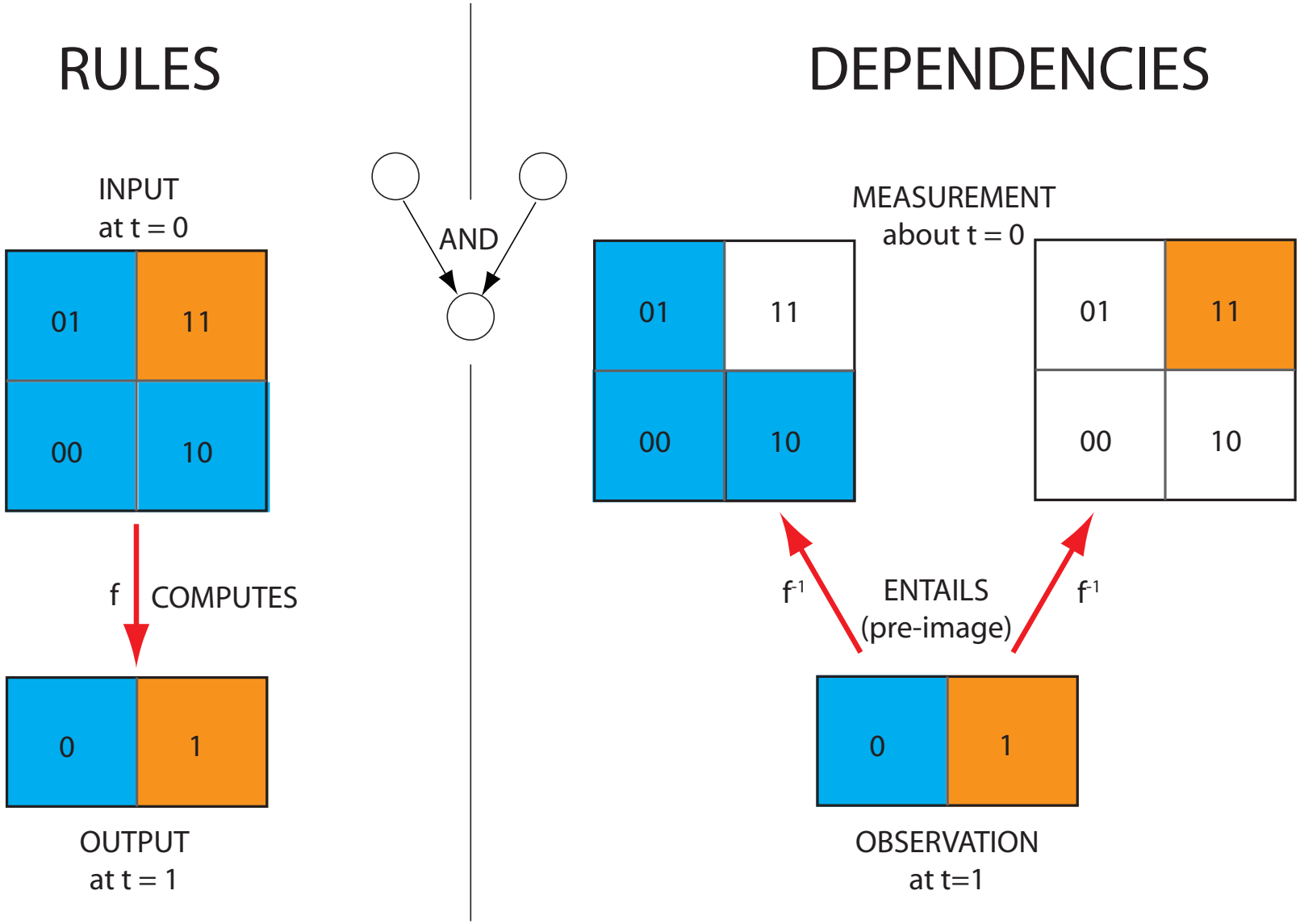
- can build logic gates out of gliders, glider guns, etc.
- and organize them to perform any computation



GLIDER

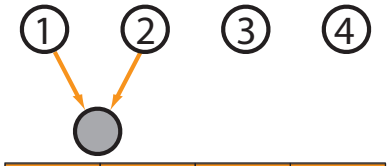
How to distinguish computationally interesting processes (gliders) from computationally boring processes (blank stretches of grid)?

[relation to learning will come later]


RULES
DEPENDENCIES
INPUT
at t = 0
01
11
00
10
f
COMPUTES
0
1
OUTPUT
at t = 1
AND
MEASUREMENT
about t = 0
01
11
00
10
01
11
00
10
f-1
ENTAILS
(pre-image)
f-1
0
1
OBSERVATION
at t=1

Dependencies in a single computation

$$f(x) = \begin{cases} 1 & \text{if } \sum x_i \geq 2 \\ 0 & \text{else.} \end{cases}$$

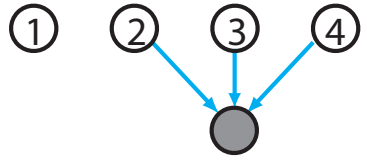


11 00	11 01	11 10	11 11
01 00	01 01	01 10	01 11
10 00	10 01	10 10	10 11
00 00	00 01	00 10	00 11

information

$= \log \left\lceil \frac{16}{4} \right\rceil$

$= 2$ bits



11 00	11 01	11 10	11 11
01 00	01 01	01 10	01 11
10 00	10 01	10 10	10 11
00 00	00 01	00 10	00 11

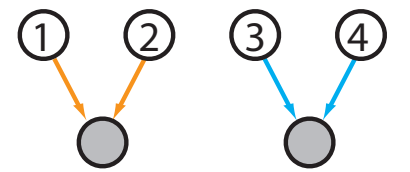
information

$= \log \left\lceil \frac{16}{8} \right\rceil$

$= 1$ bit

Independent, redundant and synergistic computations

$$f(x) = \begin{cases} 1 & \text{if } \sum x_i \geq 2 \\ 0 & \text{else.} \end{cases}$$

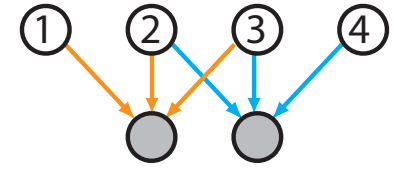


11 00	11 01	11 10	11 11
01 00	01 01	01 10	01 11
10 00	10 01	10 10	10 11
00 00	00 01	00 10	00 11

INDEPENDENT

$4 = 2 + 2$ bits

1/16

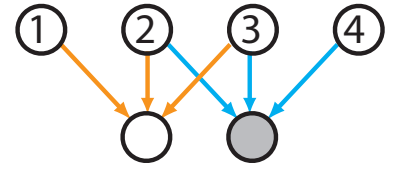


11 00	11 01	11 10	11 11
01 00	01 01	01 10	01 11
10 00	10 01	10 10	10 11
00 00	00 01	00 10	00 11

REDUNDANT

$1.4 < 1 + 1$ bits

6/16



11 00	11 01	11 10	11 11
01 00	01 01	01 10	01 11
10 00	10 01	10 10	10 11
00 00	00 01	00 10	00 11

SYNERGISTIC

$3 > 1 + 1$ bits

2/16

More formally . . .

Stochastic matrices

Given finite set X, let $\mathcal{V}X := \{f : X \to \mathbb{R}\}$ be $|X|$-dim vector space with Dirac basis $\{\delta_x | x \in X\}$.

- Any function $f : X \to Y$ can be written as

$$\mathfrak{m}_f : \mathcal{V}X \to \mathcal{V}Y : \delta_x \mapsto \delta_{f(x)}.$$

- Any conditional probability distribution $p(y|x)$ can be written

$$\mathfrak{m}_p : \mathcal{V}X \to \mathcal{V}Y : \delta_x \mapsto \sum_{y \in Y} p(y|x) \cdot \delta_y.$$

Informal definition:
stochastic matrix is any matrix arising this way.

Probabilistically representing dependencies

Matrix $\mathfrak{m}$ has **stochastic dual** $\mathfrak{m}^\natural$
(= transpose with columns renormalized)

For conditional distributions, **stochastic dual = Bayes' rule**.

	functions on sets	mechanisms on function-spaces
computation	$f : S \to A$	$\mathfrak{m} : \mathcal{V}S \to \mathcal{V}A$
dependencies	$f^{-1} : A \to \mathcal{P}(S)$	$\mathfrak{m}^\natural : (\mathcal{V}A)^* \to (\mathcal{V}S)^*$

Applying the dual yields a probability distribution, $\mathfrak{m}^\natural(\delta_a)$, that generalizes the preimage $f^{-1}(a)$:

$$\left\langle \mathfrak{m}^\natural(\delta_a), \delta_s \right\rangle = \begin{cases} \frac{1}{|f^{-1}(a)|} & \text{if } f(s) = a \\ 0 & \text{else.} \end{cases}$$

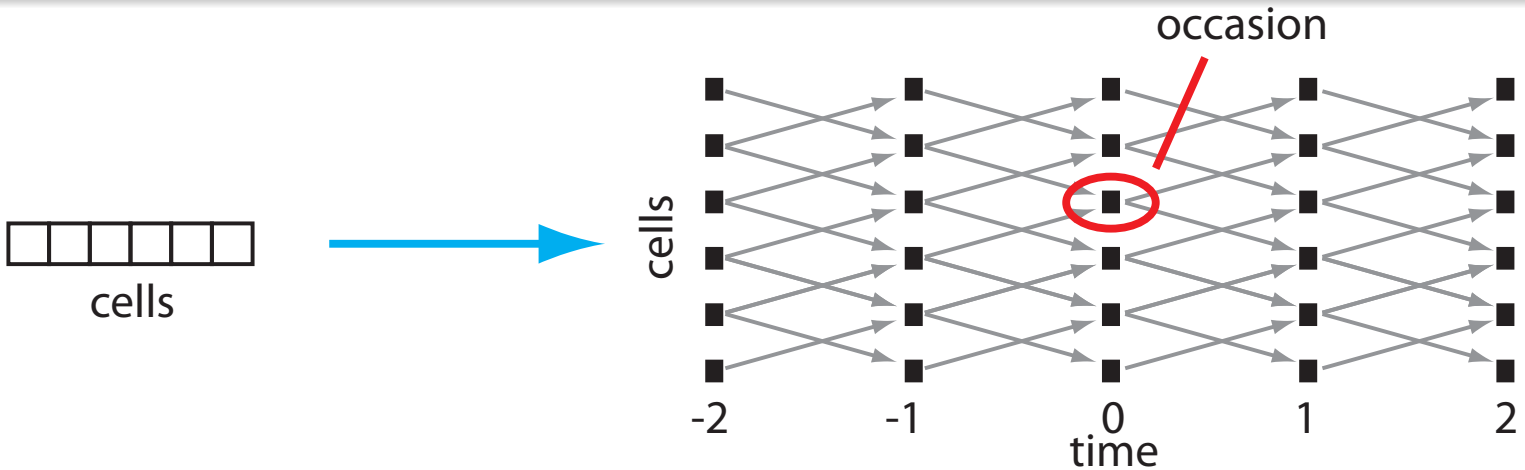
3. The geometry of dependencies

Distributed systems

A **distributed system D** consists of the following data:

1. **Directed graph**. Edges signify functional dependencies. Vertices are **occasions**: spacetime coordinates.
2. Each occasion l has **output alphabet** A_l and **input alphabet** $S_l := \prod_{k \to l} A_l$.
3. Each occasion l is equipped with a **mechanism** (= stochastic matrix):

$$\mathfrak{m}_l : \mathcal{V}S_l \to \mathcal{V}A_l.$$



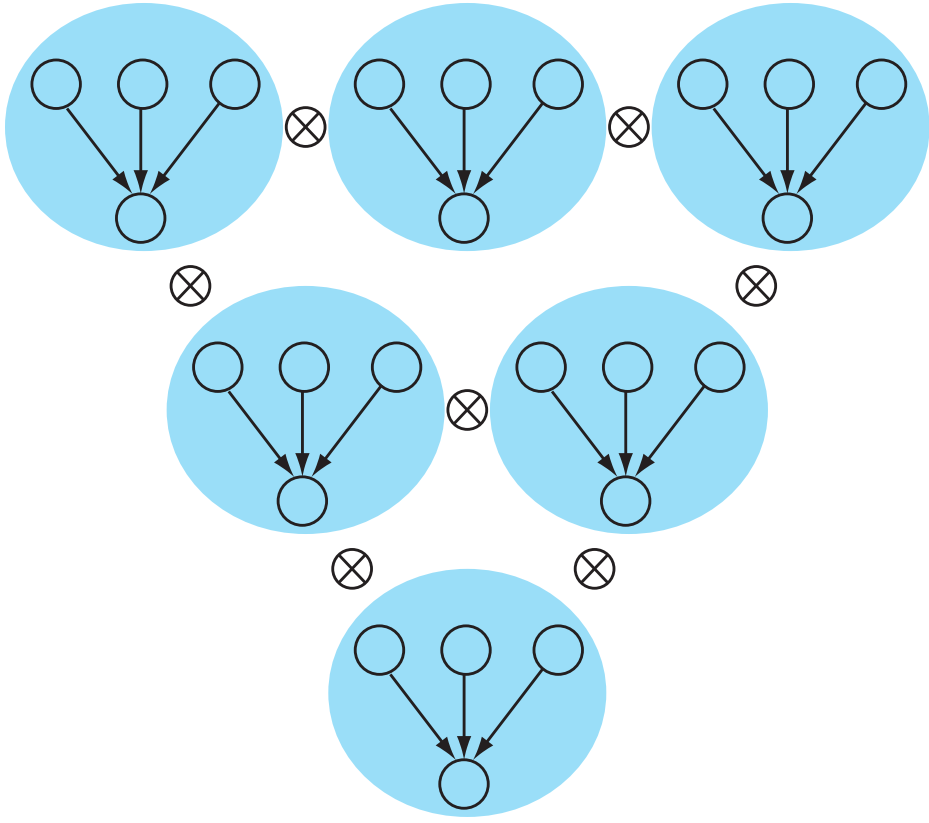
cellular automaton

distributed dynamical system

Gluing stochastic matrices

Tensor together stochastic matrices of individual occasions ...

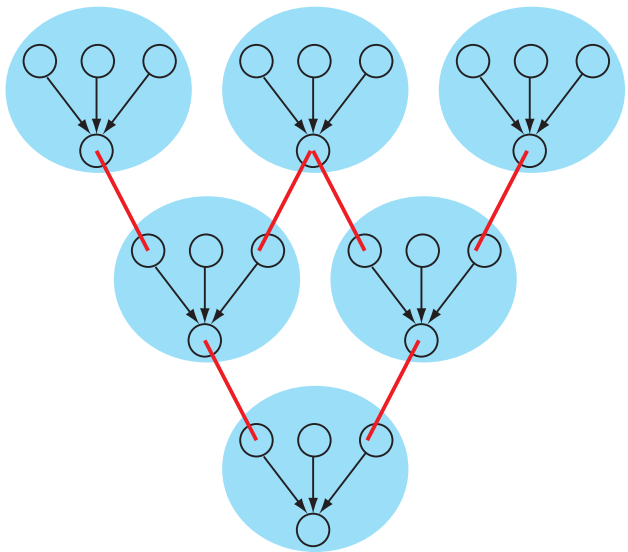
$$\mathcal{V}S^{\mathbf{D}} \xrightarrow{\iota_{\Delta}} \bigotimes_{l \in trg(\mathbf{D})} \mathcal{V}S_l \xrightarrow{\otimes \mathfrak{m}_l} \mathcal{V}A^{\mathbf{D}}$$

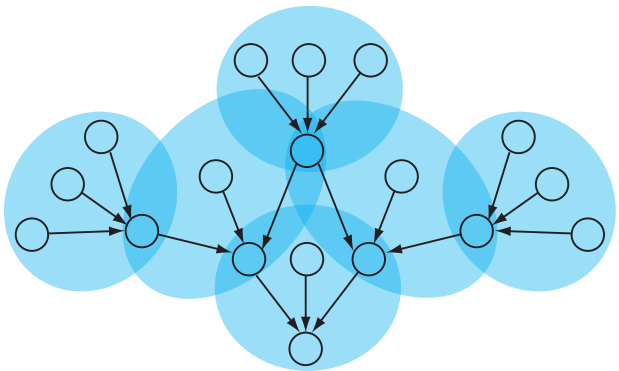


Gluing stochastic matrices

... and precompose with a "generalized diagonal" that encodes the directed graph structure

$$\mathcal{V}S^{\mathbf{D}} \xrightarrow{\iota_\Delta} \bigotimes_{I \in trg(\mathbf{D})} \mathcal{V}S_I \xrightarrow{\otimes \mathfrak{m}_I} \mathcal{V}A^{\mathbf{D}}$$

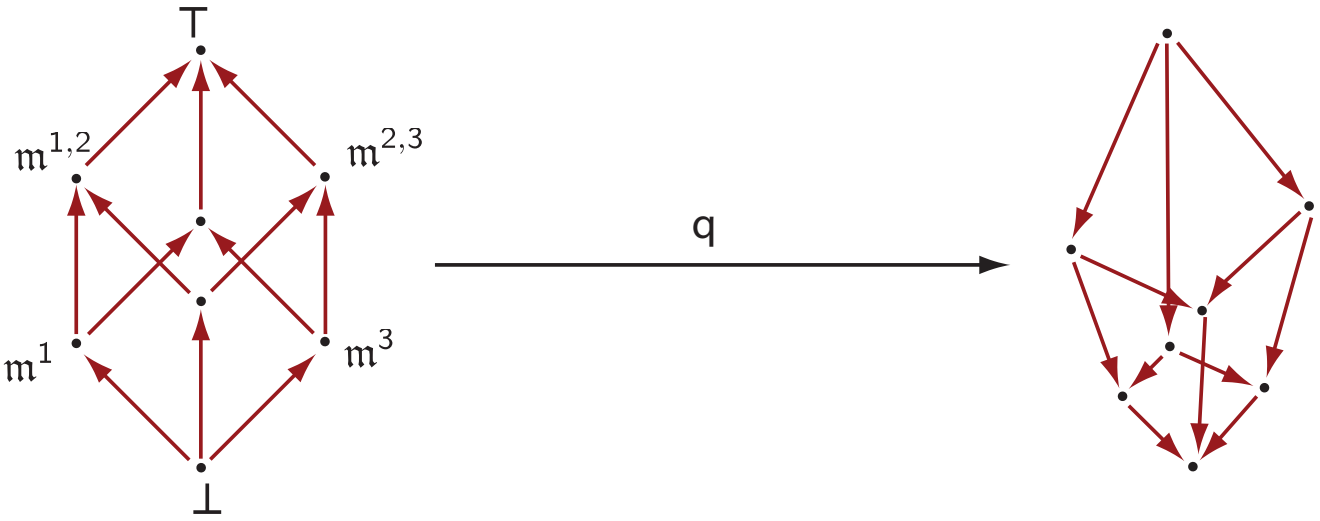




Probabilistic dependencies $\longrightarrow$ sections of a presheaf

Given global output a by system $\mathbf{D}$,

$$q_a : \quad \left\{\text{subsystems } \subset \mathbf{D}\right\} \longrightarrow \left\{\text{dependencies in } \mathcal{V}S^{\mathbf{D}}\right\}$$
$$\mathbf{C} \mapsto \mathfrak{m}^{\natural}_{\mathbf{C}}(\delta_a)$$

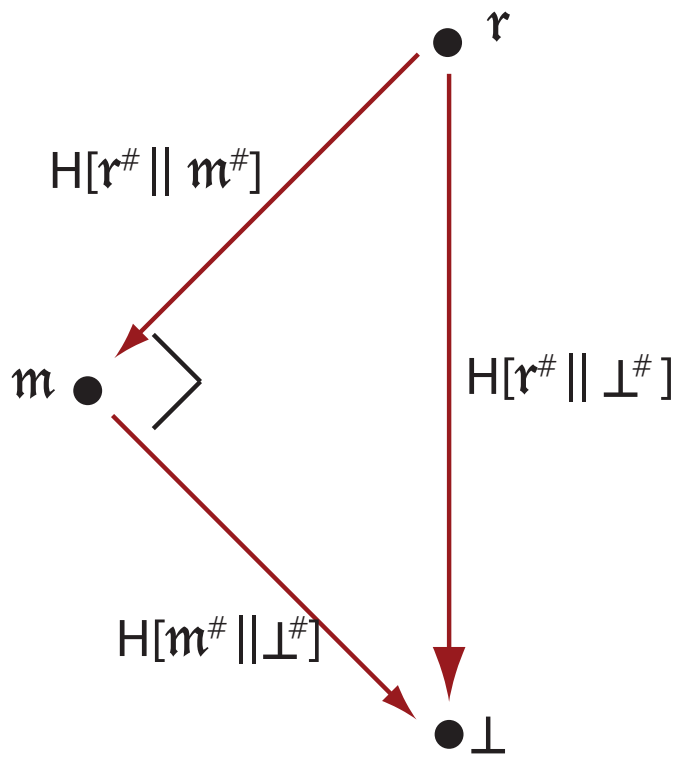


"Lengths" of arrows $\leftrightarrow$ **effective information**:

$$ei(\mathfrak{m} \to \mathfrak{r}, a) := H\left[\mathfrak{r}^{\natural}(\delta_a) \,\middle\|\, \mathfrak{m}^{\natural}(\delta_a)\right],$$

where $H[p\|q] := \sum_i p_i \log \frac{p_i}{q_i}$.

Information-theoretic Pythagoras (Amari and Nagaoka)



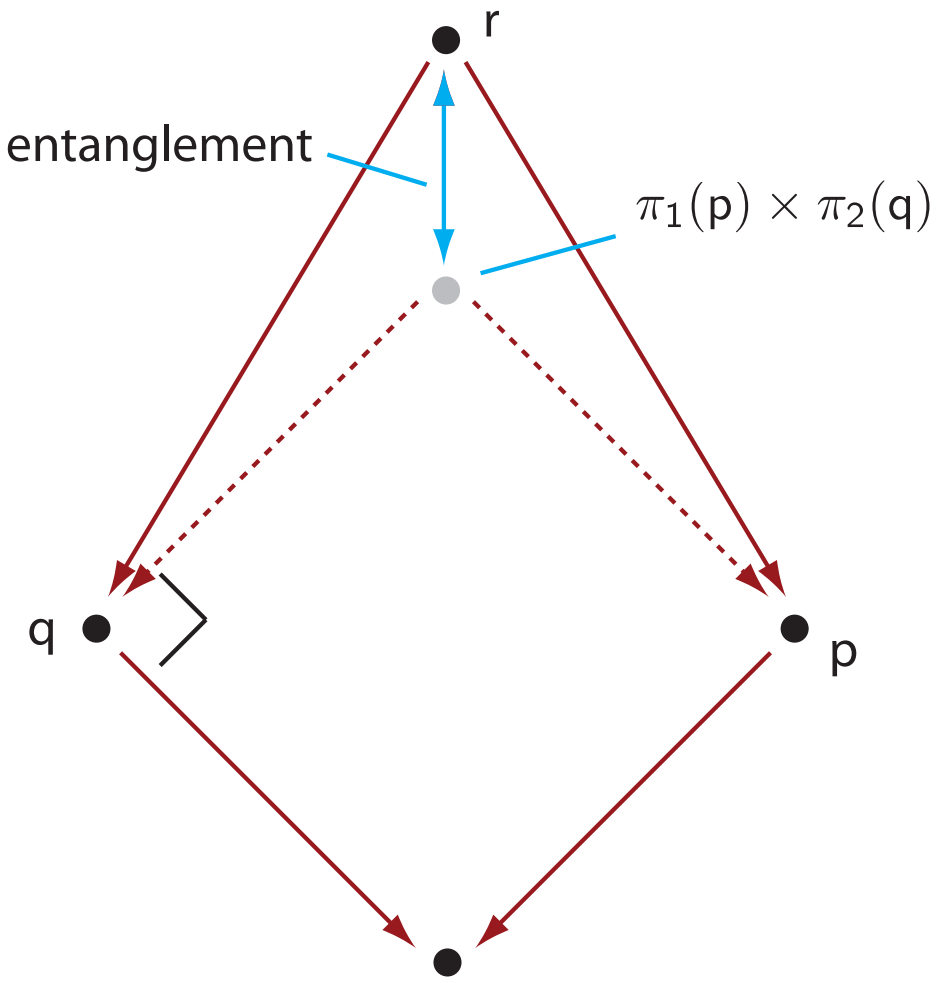
If computations by $\mathfrak{m}$ and $\mathfrak{t}$ are independent then

$$H\left[\mathfrak{r}^{\natural} \,\middle\|\, \perp^{\natural}\right] = H\left[\mathfrak{r}^{\natural} \,\middle\|\, \mathfrak{m}^{\natural}\right] + H\left[\mathfrak{m}^{\natural} \,\middle\|\, \perp^{\natural}\right],$$

where $\mathfrak{r} = \mathfrak{m} \cup \mathfrak{t}$.

Entanglement quantifies obstruction to unique descent in presheaf

[gluing axiom holds]



Theorem ("Gestalt theorem")

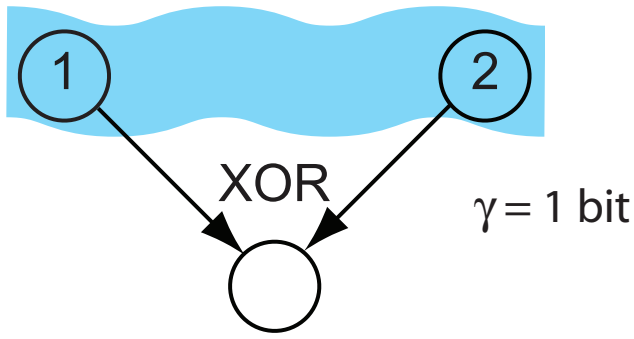
Entanglement is necessary for synergistic dependencies:

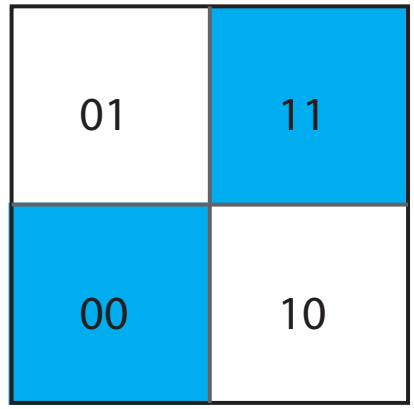
$$ei(\mathfrak{m}_1 \cup \mathfrak{m}_2) > ei(\mathfrak{m}_1) + ei(\mathfrak{m}_2) \implies \gamma > 0.$$

$$\gamma = H\Big[r \,\Big\|\, \pi_1(p) \otimes \pi_2(q)\Big]$$

Example: An entangled XOR-gate

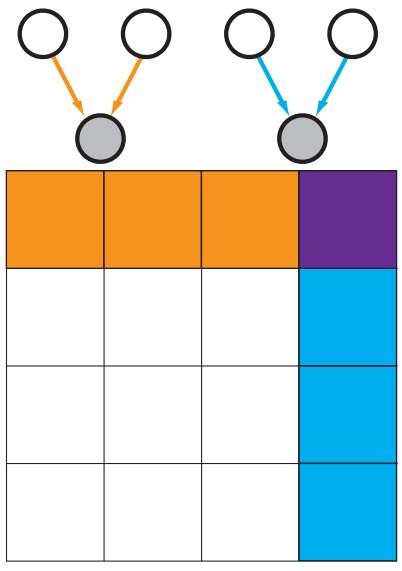
A XOR-gate outputting 0 entails the input was **either** 00 or 11.



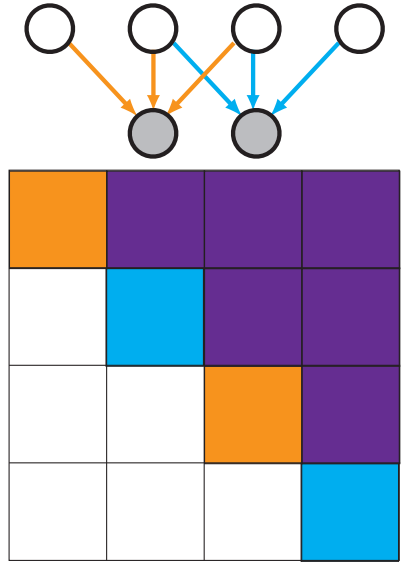


XOR generates 1 bit of information about $\{n_1, n_2\}$,
but zero bits about n_1 and n_2 separately.

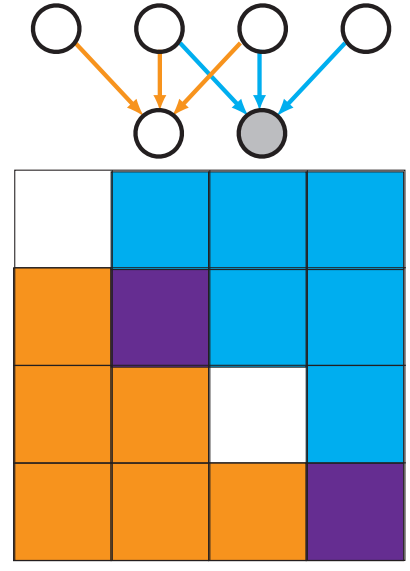
Independent, redundant and synergistic computations



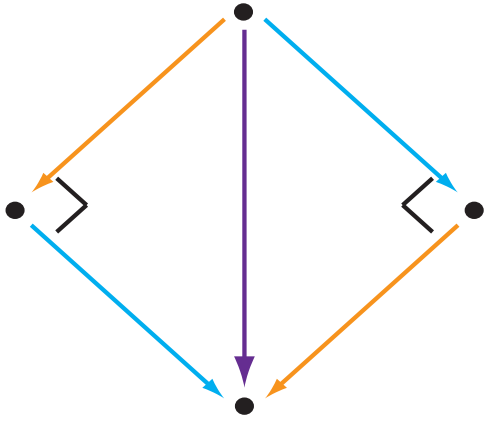
INDEPENDENT

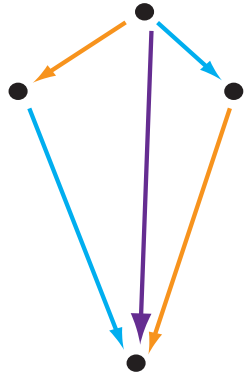


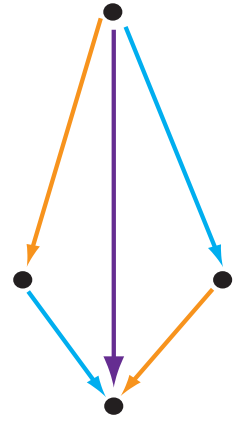
REDUNDANT



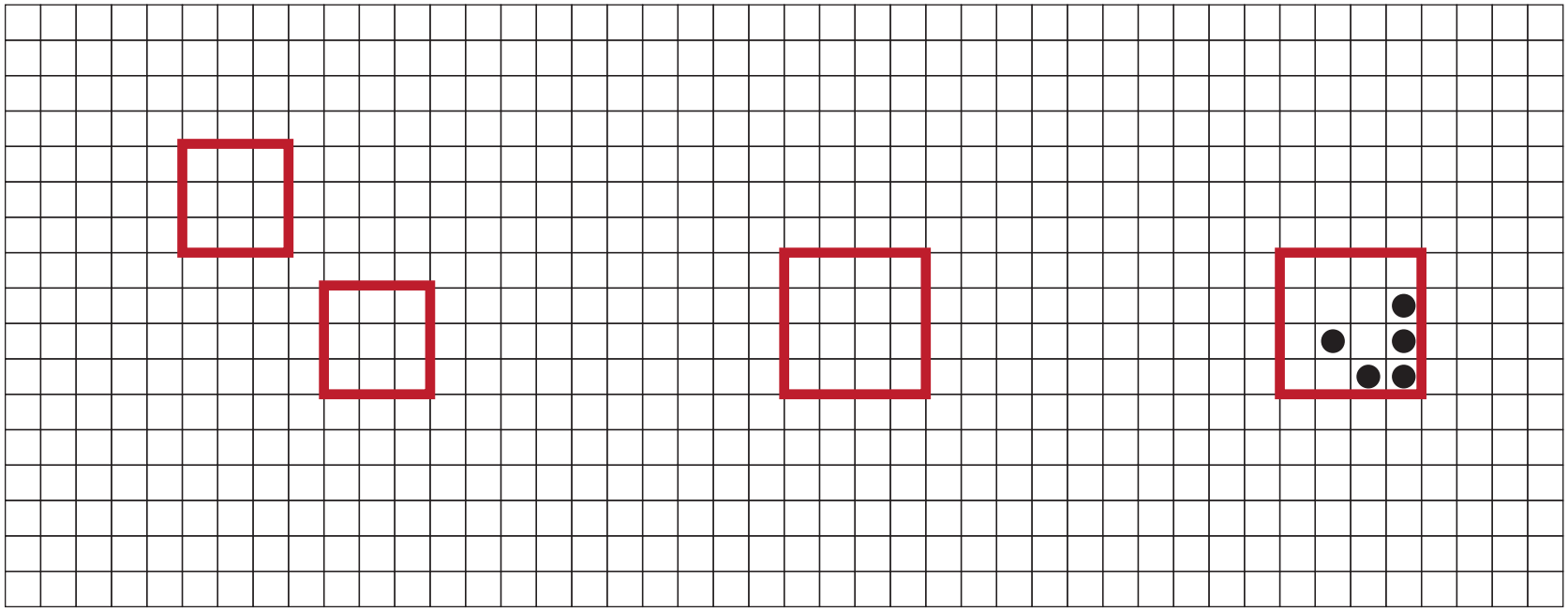
SYNERGISTIC







Dependencies in Conway's Game of Life



4. Statistical learning theory

Setup

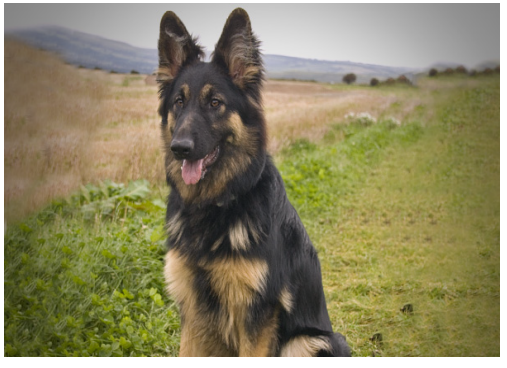
Suppose data $\mathcal{D} = (x_1, \ldots, x_\ell)$ is drawn from unknown probability distribution P_X and labeled as belonging to one of two categories by an unknown supervisor $\sigma : \mathcal{D} \to \{\pm 1\}$.

The learning problem:
Find a classifier **guaranteed to perform well** on **future data** sampled from P_X and labeled by σ.

x_1 cat

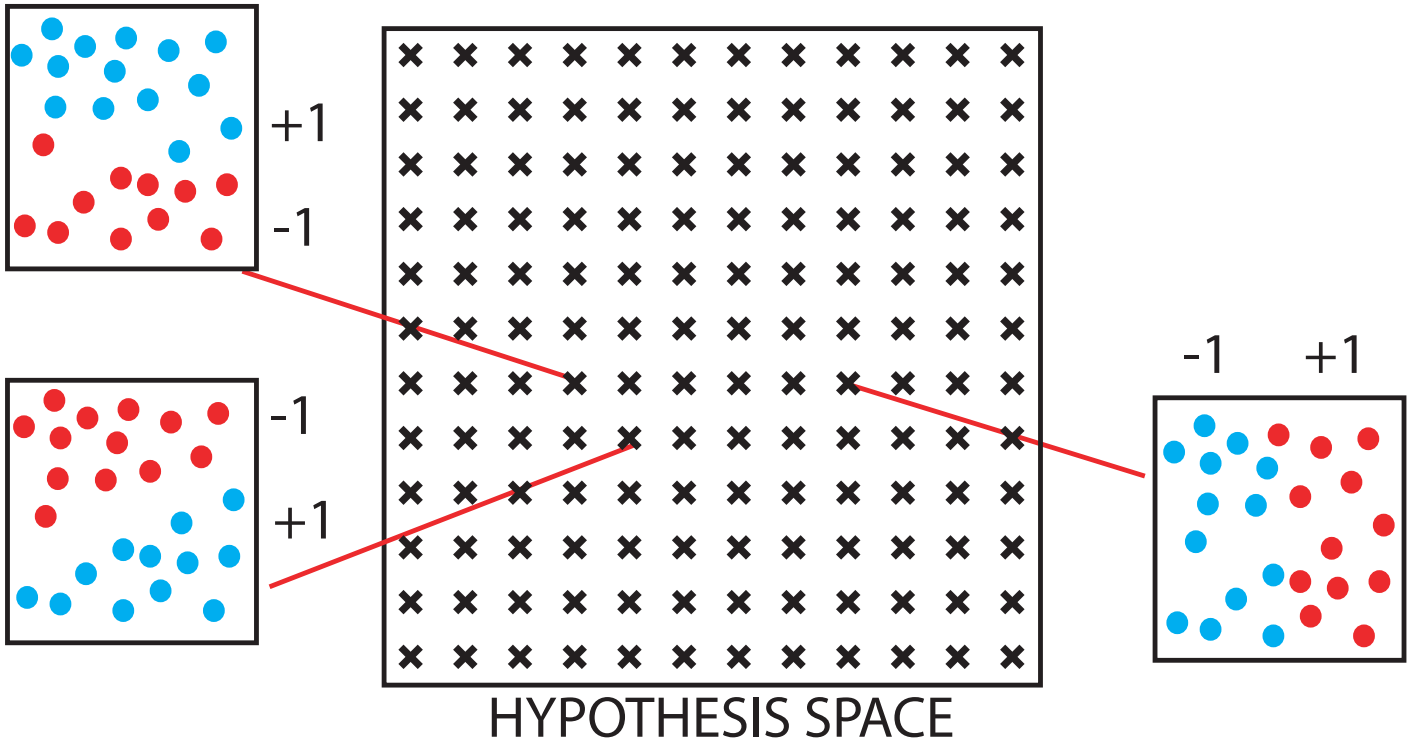
x_2 cat

x_3 dog



Hypothesis space

Given unlabeled data $\mathcal{D} = (x_1, \ldots, x_\ell) \subset X^\ell$, let **hypothesis space** $\Sigma_\mathcal{D} = \left\{\sigma : \mathcal{D} \to \pm 1\right\}$ be the set of all possible labelings.



Empirical risk minimization

Suppose we are given a class $\mathcal{F}$ of functions.

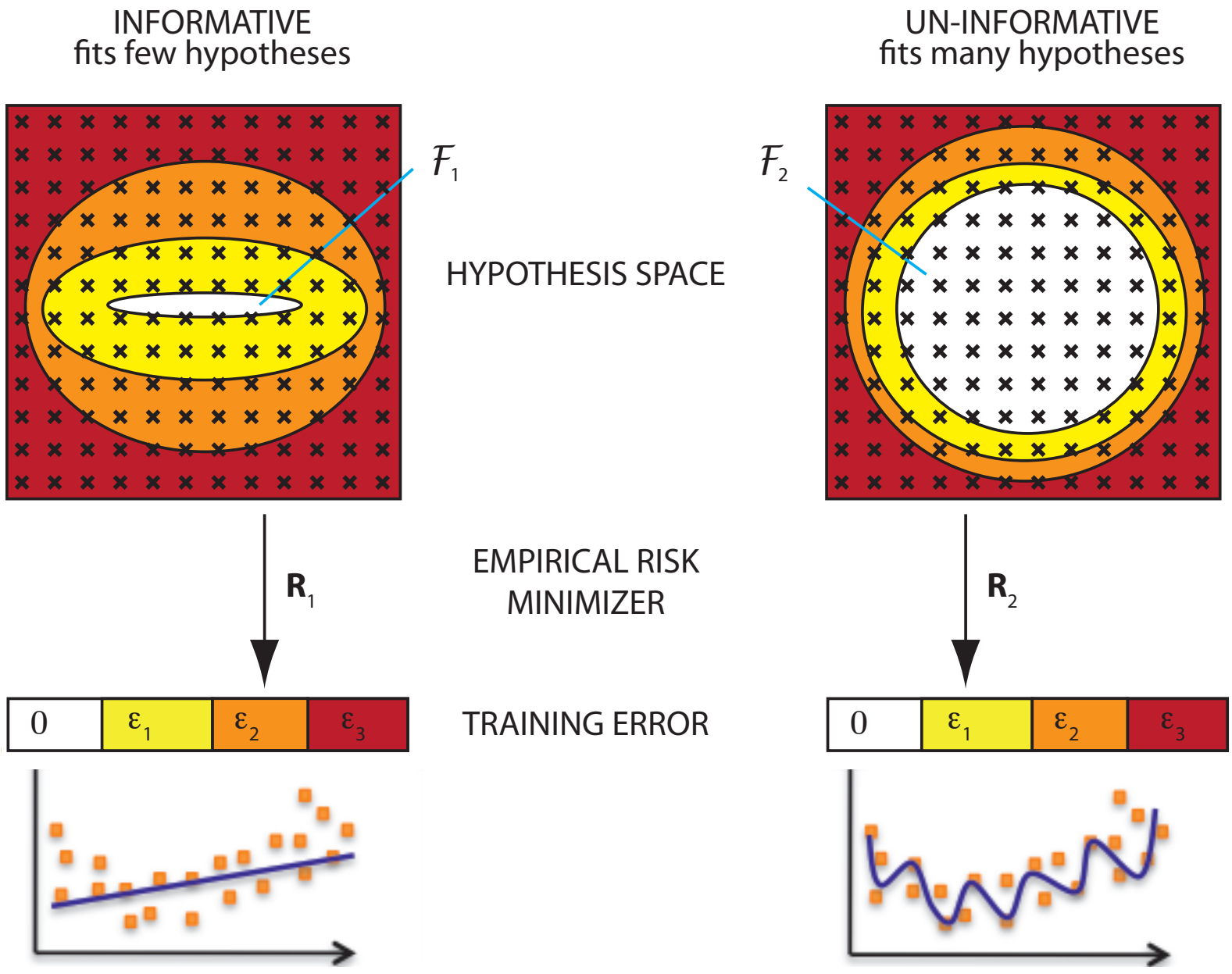
Algorithm:
Given data $\mathcal{D}$ labeled by supervisor σ, find classifier $\hat{f} \in \mathcal{F}$ that minimizes (training) errors on data:

$$\hat{f} := \arg\min_{f \in \mathcal{F}} \frac{1}{\ell} \sum_{i=1}^{\ell} \mathbb{I}_{f(x_i) \neq \sigma(x_i)}$$

Reformulate algorithm as a function between finite sets:

Empirical risk minimization:

$$\begin{array}{lccc} \mathbf{R}_{\mathcal{F},\mathcal{D}}: & \text{HYPOTHESIS SPACE} & \longrightarrow & \text{EMPIRICAL RISK} \\ & \sigma & \mapsto & \min_{f \in \mathcal{F}} \frac{1}{\ell} \sum_{i=1}^{\ell} \mathbb{I}_{f(x_i) \neq \sigma(x_i)} \end{array}$$


INFORMATIVE
fits few hypotheses
UN-INFORMATIVE
fits many hypotheses
$\mathcal{F}_1$
$\mathcal{F}_2$
HYPOTHESIS SPACE
$\mathbf{R}_1$
$\mathbf{R}_2$
EMPIRICAL RISK
MINIMIZER
0
ε_1
ε_2
ε_3
TRAINING ERROR

Effective information $\leftrightarrow$ capacity of learner (VC-entropy)

Corollary

With probability at least $1-\delta$,

$$\underbrace{\mathbf{R}(f)}_{\text{future error}} \leq \underbrace{\widehat{\mathbf{R}}(f)}_{\text{training error}} + \underbrace{c_1\sqrt{1-\frac{ei(\mathfrak{m}_{\mathbf{R}_{\mathcal{F},\mathcal{D}}},0)}{\ell}}}_{\text{information generated by ERM}} + \underbrace{conf(\delta)}_{\text{confidence term}}$$

where ℓ is amount of training data.

Proof: Combine standard learning theory with a result linking capacity to effective information. ■

Conclusion

Learning algorithms generating more effective information,
have better guarantees on their future performance.

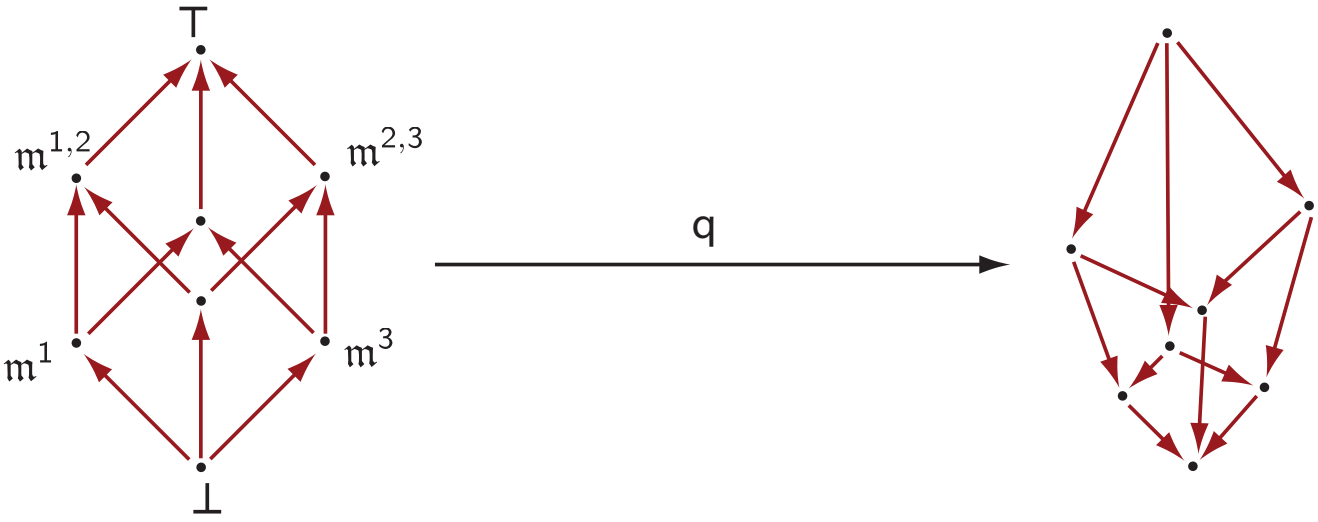
Complex learning algorithms are less informative.

5. Goal

Topological methods for distributed learning

- Have defined presheaf

$$\mathcal{S} : \text{Lattice}^{op} \longrightarrow \text{Probability distributions.}$$



- entanglement = deviation from trivial gluing
  detects synergistic computations which (when computation relates to learning) improve performance guarantees.

Goal: develop cohomology theory (?) suitable for analyzing interacting learning algorithms

Thank you for
your attention

References

1. Effective information and integrated information:
   - in **PLoS Computational Biology (2008)**
2. Geometry and synergy, entanglement:
   - in **PLoS Computational Biology (2009)**
3. Stochastic matrices, presheaf (on arXiv):
   - **"On the information-theoretic structure of distributed measurements"**
4. Relation to statistical learning theory (on arXiv):
   - **"Falsification and future performance"**
   - **"Information, learning and falsification"**