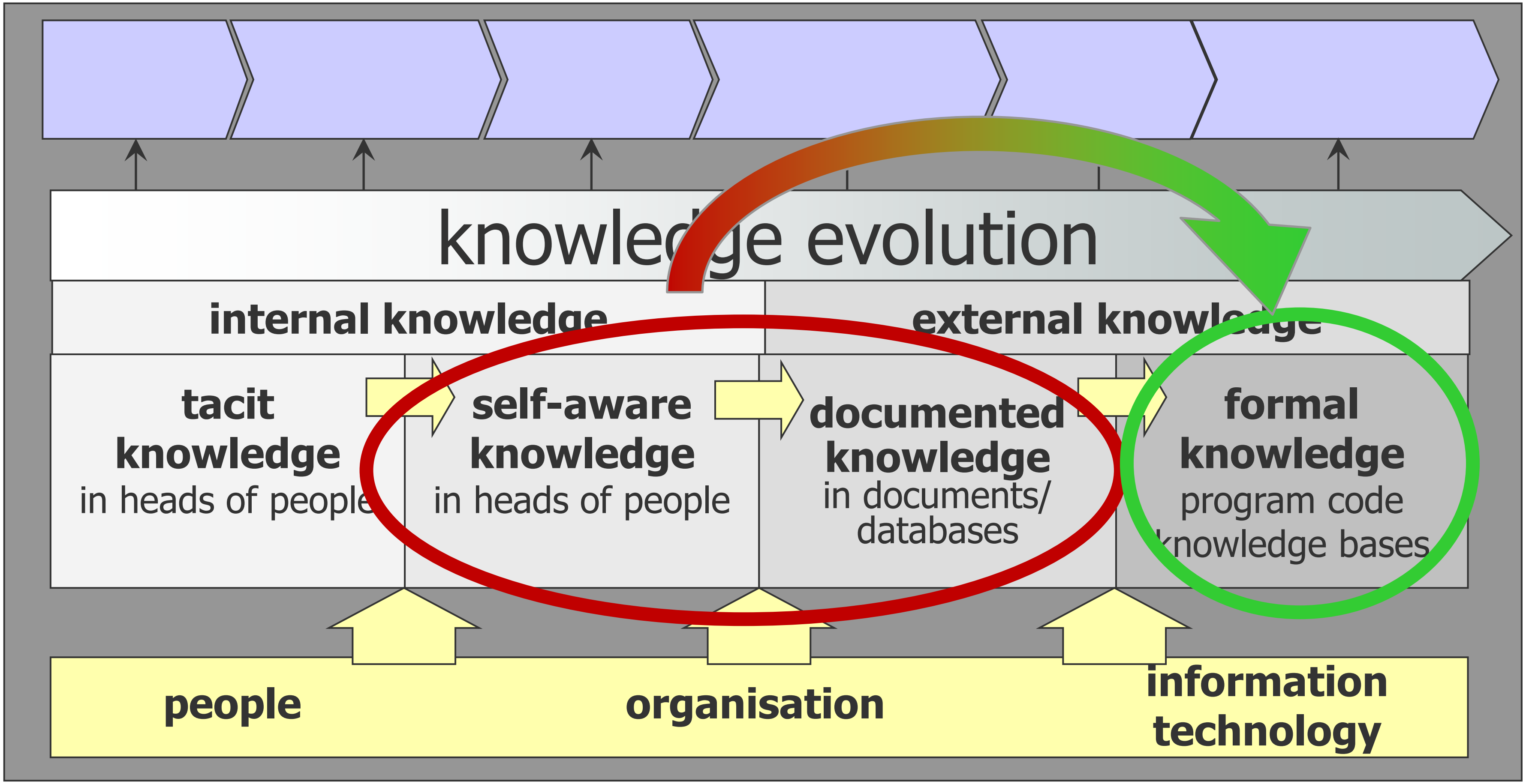
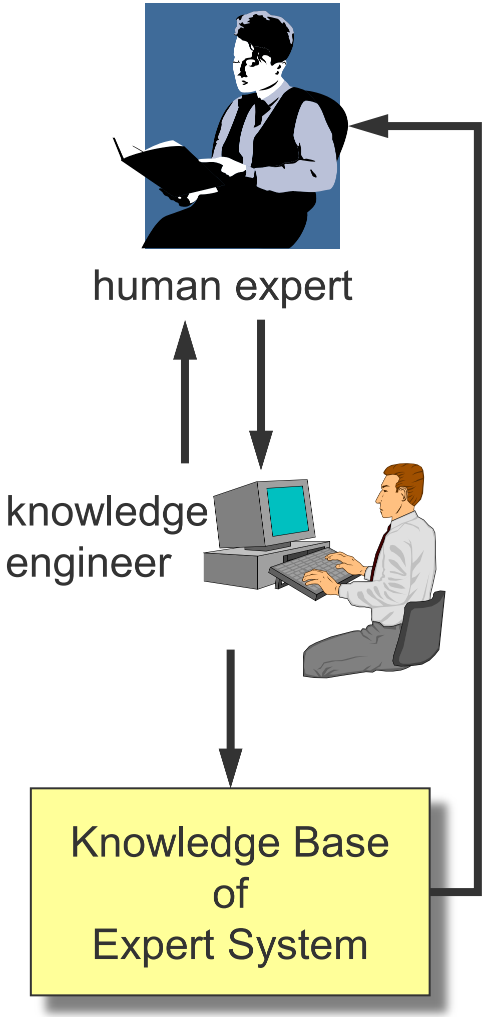
Machine Learning - An Introduction

Knut Hinkelmann

Knowledge Engineering



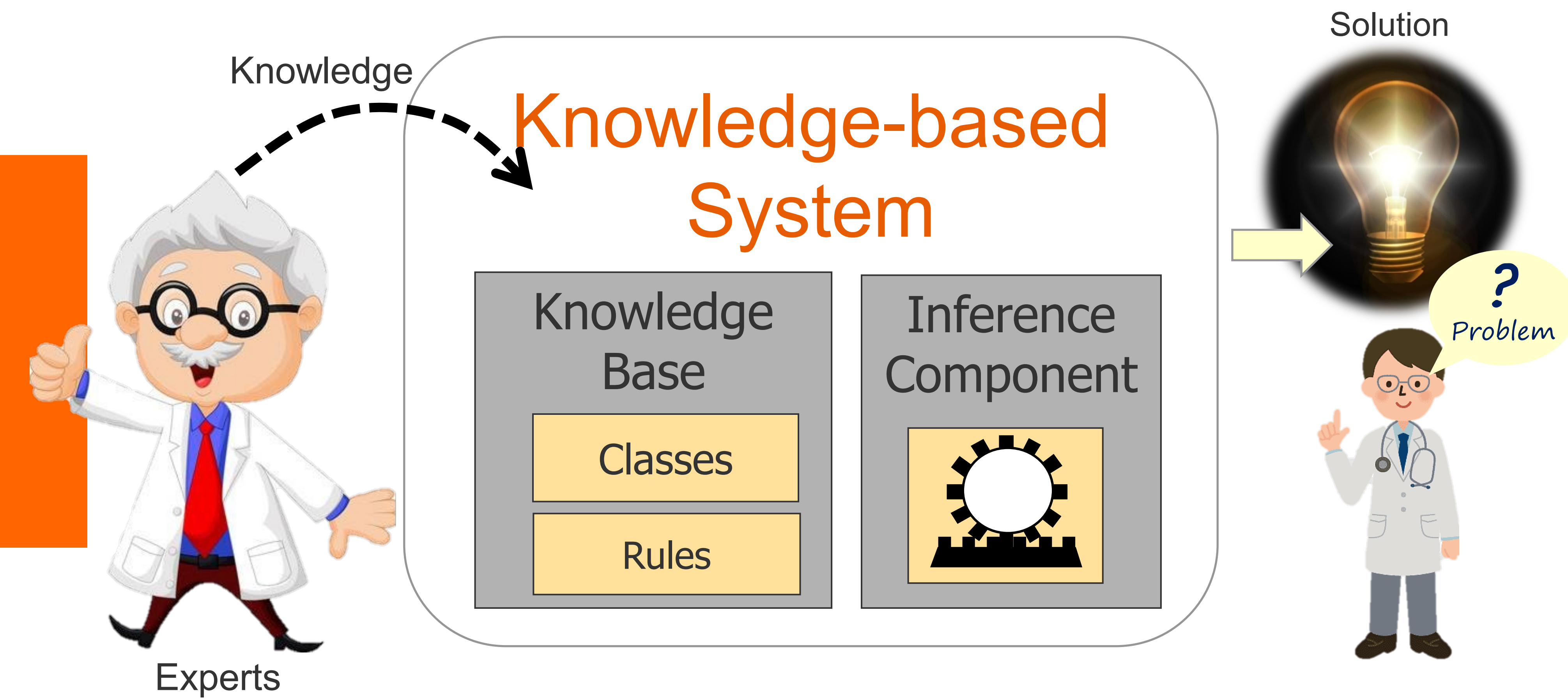
Knowledge Engineering



- Knowledge Engineering is the process of
  - building and
  - maintaining

  knowledge-based systems or intelligent agents
- *"Knowledge Engineering is an engineering discipline that involves integrating knowledge into computer systems in order to solve complex problems normally requiring a high level of human expertise."*[1)]
- Sources of knowledge
  - Human experts
  - Documentation

1) Feigenbaum, E., and P. McCorduck. (1983). The Fifth Generation. Reading, MA: Addison-Wesley

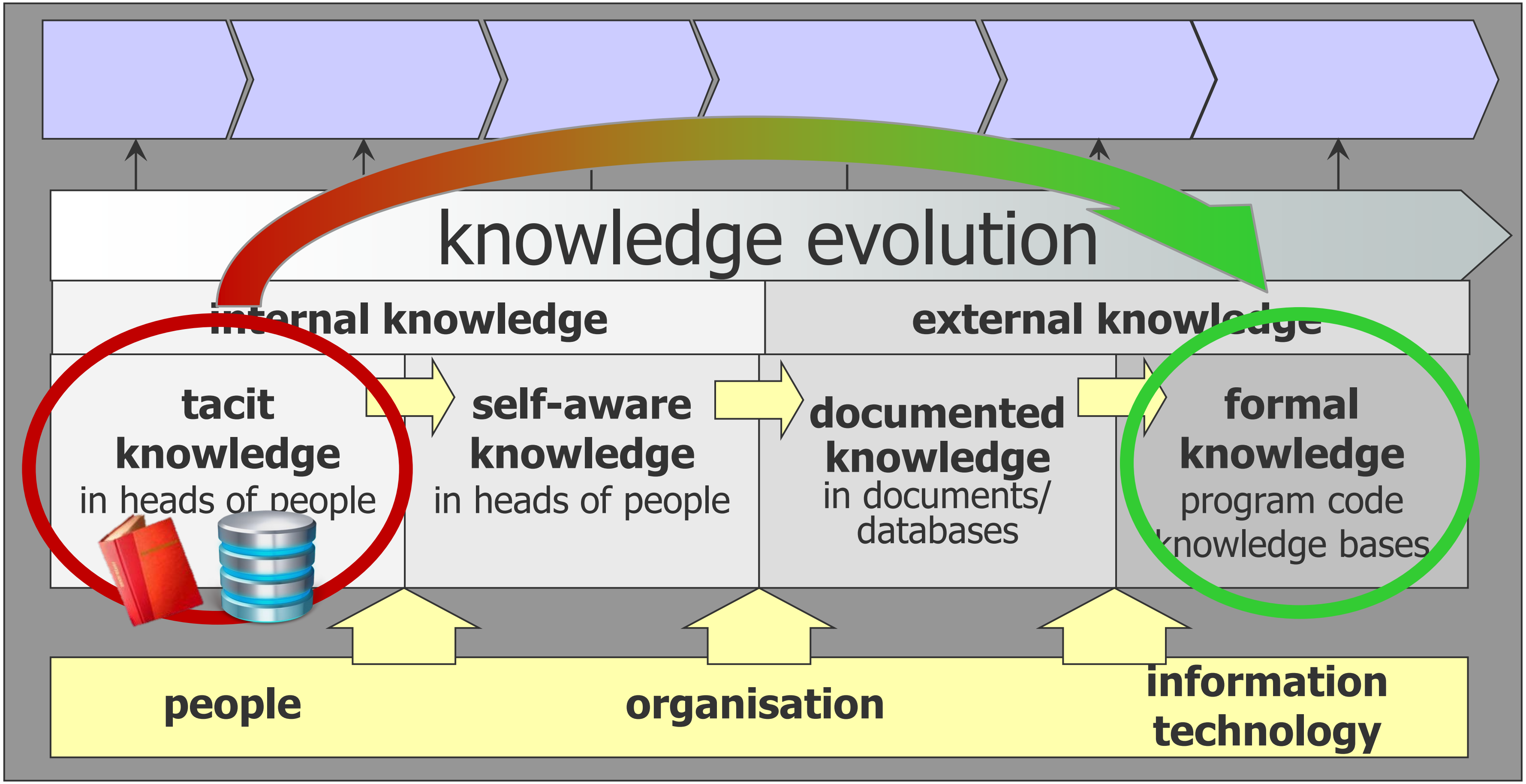


Glühbirne von Gerd Altmann auf Pixabay

Drawbacks of Knowledge Engineering

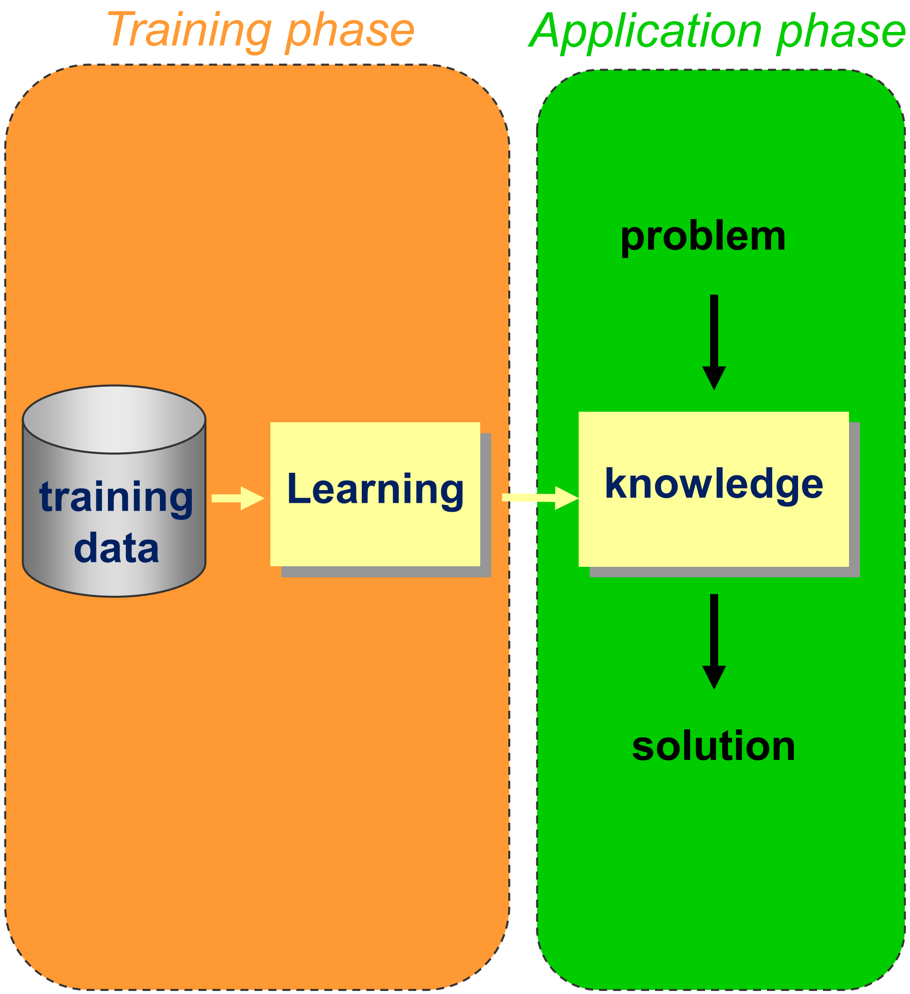
- Effort to …
  - … build the knowledge base
  - … maintain the knowledge base
- Availability of knowledge
- Awareness of knowledge

Machine Learning:
Make Knowledge explicit with the Use of Data



From data (texts or structured data) it is possible to learn tacit knowledge and new knowledge

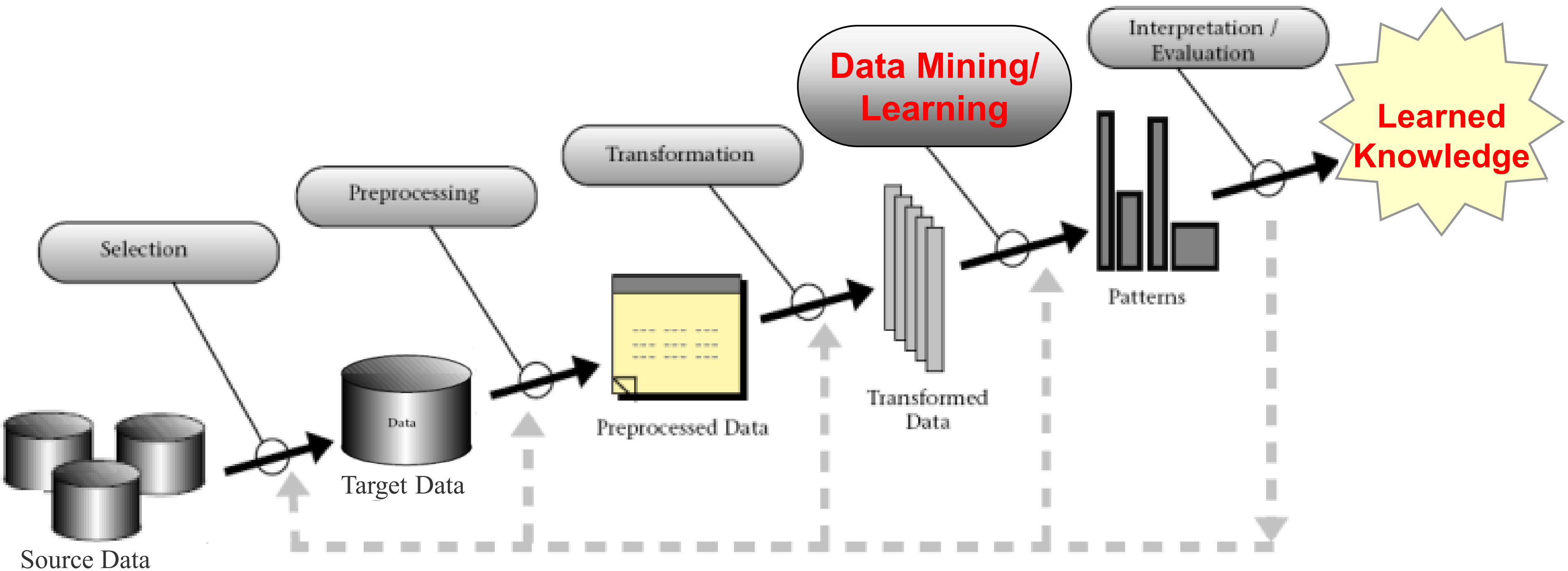
Machine Learning: General Idea



- **Learning/Training**
  - Collect data for the problem
  - Use the data to learn how to solve the type of problem
  - Result: Knowledge
- **Application**
  - Use the learned knowledge for new problems

Machine Learning in Context

- Machine Learning (Data Mining) is a step to discover knowledge in data



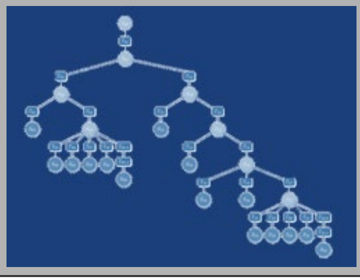
(Fayyad et al., 1996)

Learned Knowledge can then be applied to solve problems, make decisions.

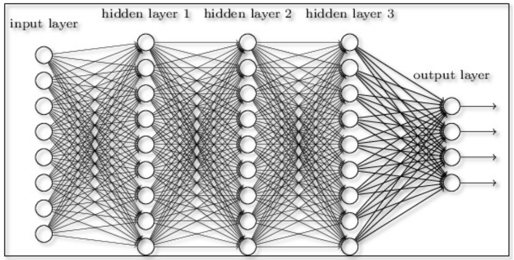
Symbolic vs Subsymbolic Learning

Learning Rules and Classes

Knowledge Base



Training a Neural Network



Types of Learning

- The learning method depends on the kind of data that we have at our disposal
  - The data contains sets of inputs and corresponding outputs: (i,o)
  - No prior knowledge: The data contains only the inputs i: output has to be determined
  - The data contains sets of inputs without corresponding «correct» output, but we can get some measure of the quality of an output o for input i.
    Rewards for good output quality.

→ **Supervised Learning**

→ **Unsupervised Learning**

→ **Reinforcement Learning**

(Lison, 2012)

Supervised Learning: Application Examples

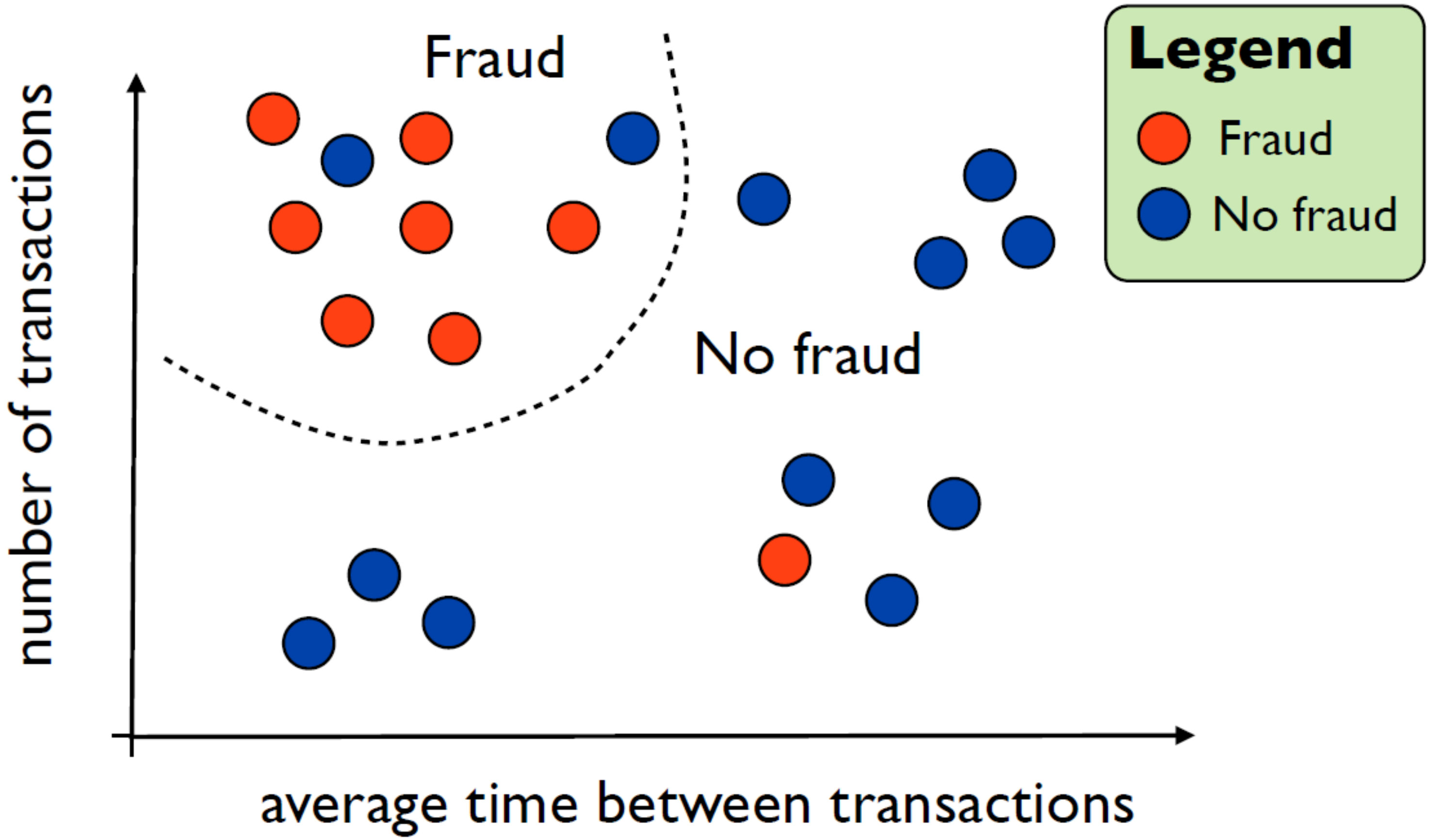
	Input *i*	Output **o**
Spam filtering	An email	{spam, non-spam}
Fraud detection	A financial transaction	{fraud, non-fraud}
Face recognition	An image	Identified faces
Machine translation	A sentence in language A	A sentence in language B

Classification

(Lison, 2012)

Supervised Learning

Example: Classification



(Lison, 2012)

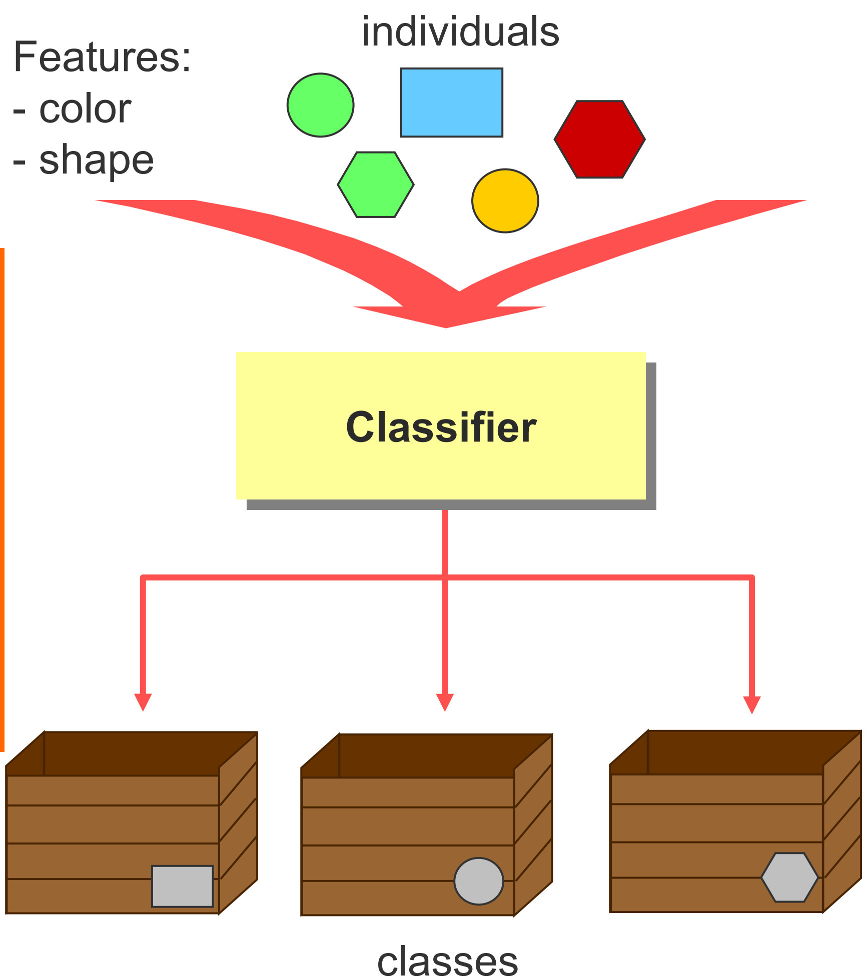
Classification

- Assign objects (input) to known classes (output)
- Examples:
  - credit assessment
    Input: customers of a bank
    Classes: credit worthy
    not credit worthy
  - Spam filtering
    Input: email
    Classes: spam
    non-spam
  - optical character recognition (OCR)
    Input: scanned pixel image
    Classes: ASCII characters

Spam Filter

"It's not the most sophisticated Spam blocker I've tried, but it's the only one that works!"

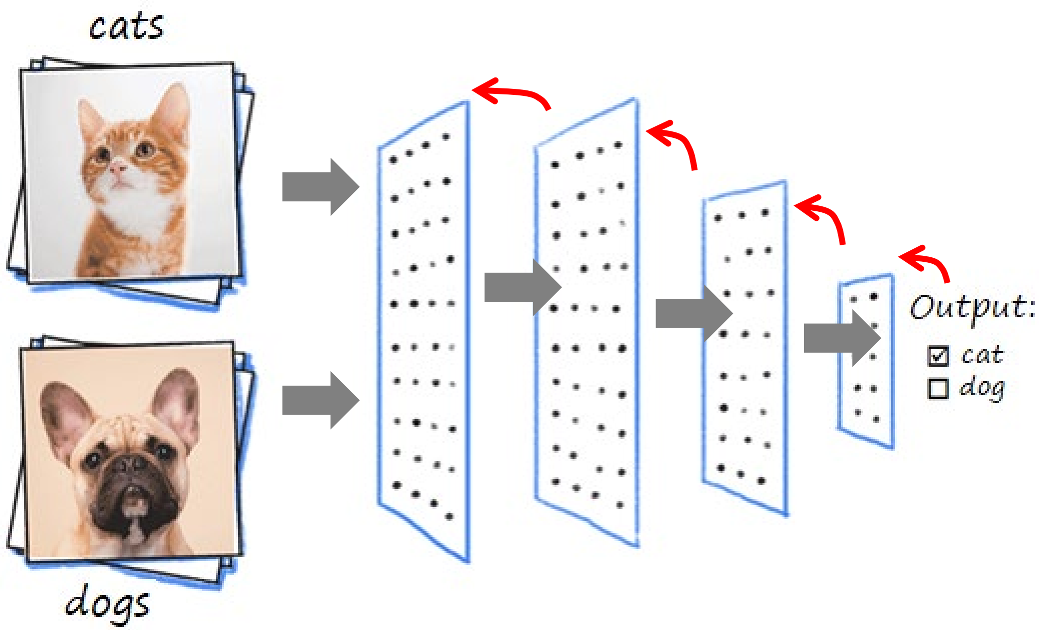
Supervised Learning: Classification Criteria



- The classifier decides, which individual belongs to which class
- Problem:
  - Input has different features
  - The criteria for the decision are not always obvious
- Supervised Learning:
  - Learn the classification criteria from known examples
  - Criteria = relevant features and their valures

Example for Supervised Subsymbolic Learning

Training with large sets of data



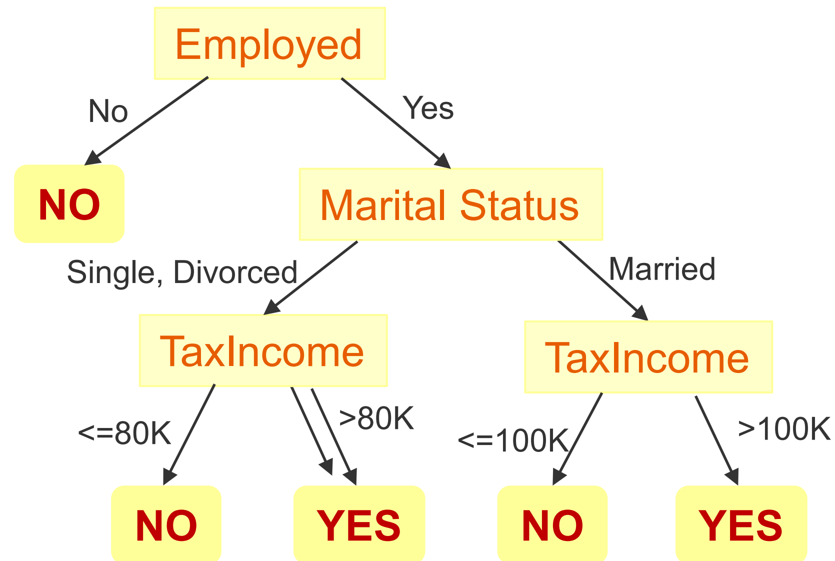
Application: cat or dog?

Example for Supervised Symbolic Learning

Problem: When to give credit

Tid	**Employed**	**Marital Status**	**Taxable Income**	**accept**
1	No	Single	125K	**No**
2	Yes	Married	160K	**Yes**
3	Yes	Single	70K	**No**
4	No	Married	120K	**No**
5	Yes	Divorced	95K	**Yes**
6	Yes	Married	60K	**No**
7	No	Divorced	220K	**No**
8	Yes	Single	85K	**Yes**
9	Yes	Married	95K	**No**
10	Yes	Single	90K	**Yes**

Training Data



Credit Worthiness

	Employed	Marital Status	Taxable Income	Accept
	Yes, No	*Single, Divorced, Married*	*Integer*	*Yes, No*
1	No			No
2	Yes	Single	> 80K	Yes
3	Yes	Divorced	> 80K	Yes
4	Yes	Single	≤ 80K	No
5	Yes	Divorced	≤ 80K	No
6	Yes	Married	> 100K	Yes
7	Yes	Married	≤ 100K	No

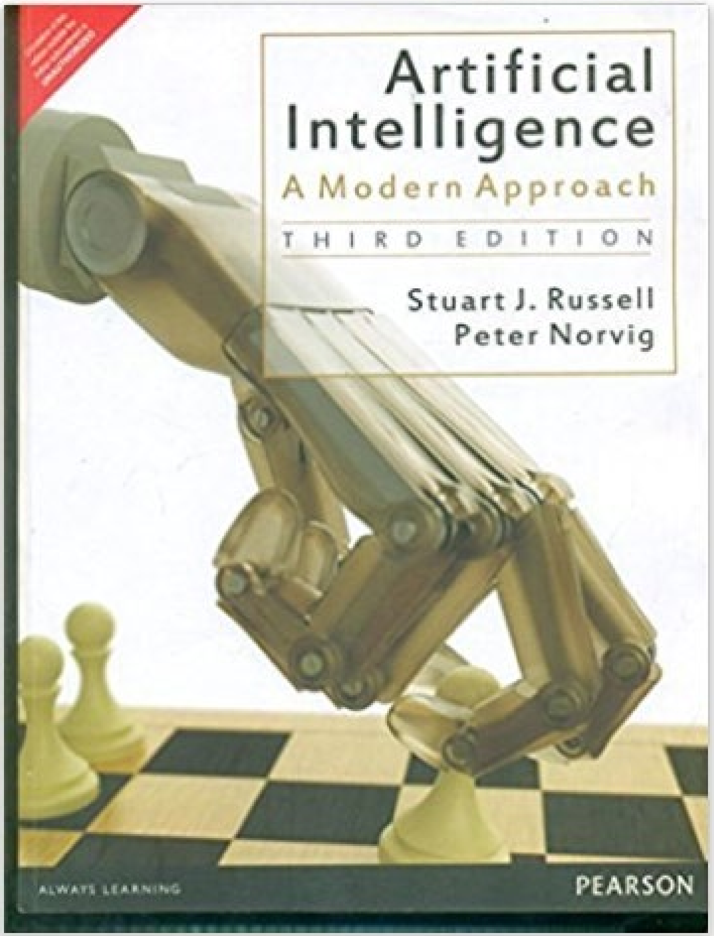
Knowledge Base: Decision Tree , Decision Table

Unsupervised Learning

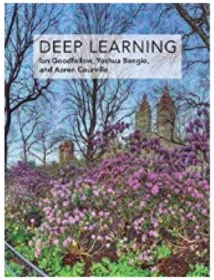
- Sometimes, we don’t have access to any output value **o**, we simply have a collection of input examples **i**
- Input: data sets without corresponding output values.
- Objective: learn the underlying patterns of our data
  - Are there any *correlations* between features?
  - Can we *cluster* our data set in groups which behave similarly?

(Lison, 2012)

Example: Recommender Systems

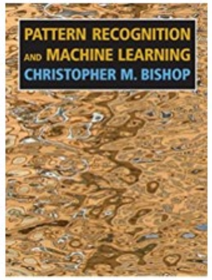


Customers who bought this item also bought



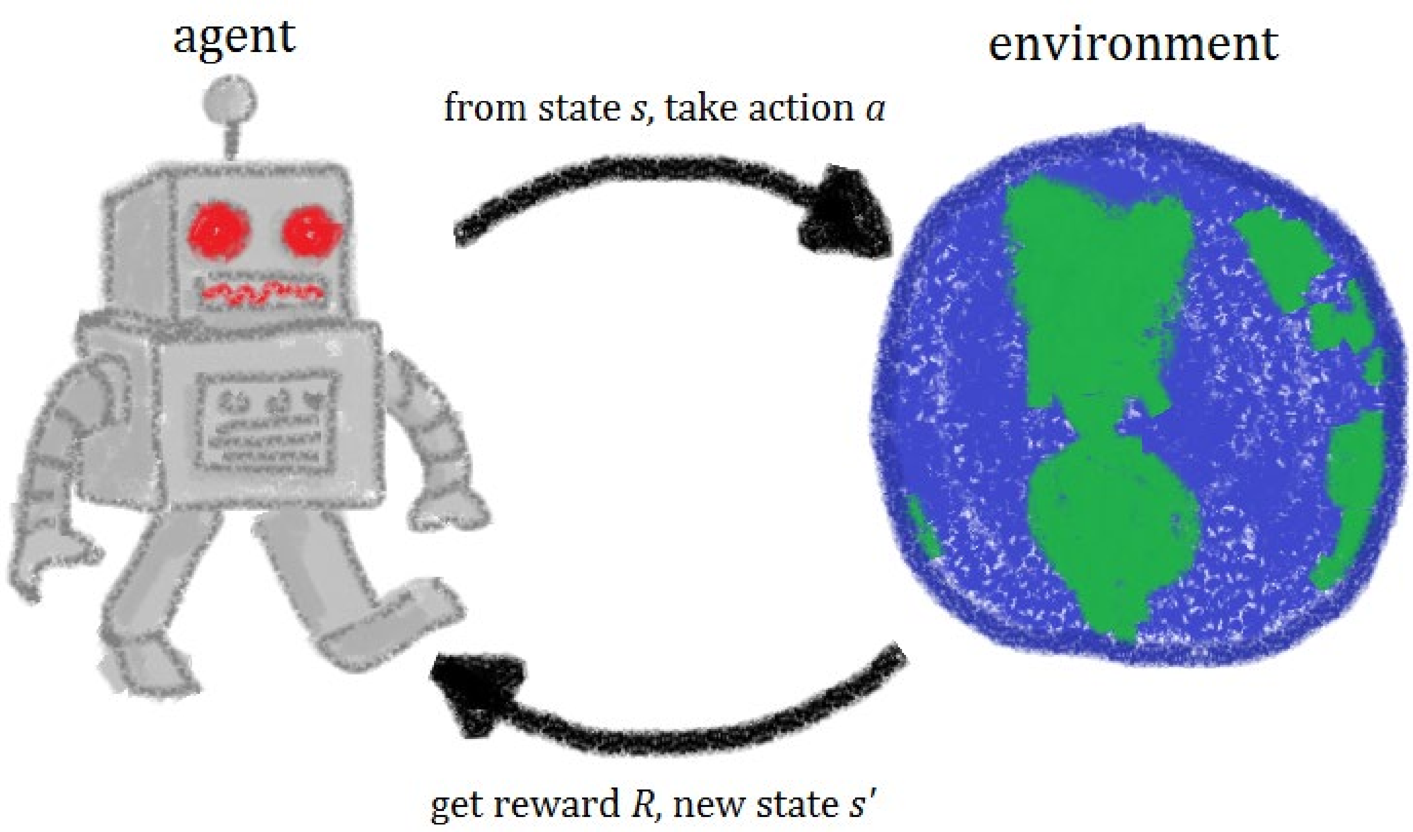
Deep Learning (Adaptive Computation and Machine Learning series)
› Ian Goodfellow

Hands-On Machine Learning with Scikit-Learn and TensorFlow:...
› Aurélien Géron



Pattern Recognition and Machine Learning (Information Science...
› Christopher M. Bishop

Reinforcement Learning



Reinforcement Learning

- Sometimes we don't have direct access to «the» correct output ***o*** for an input ***i***
- But we can get a measure of «how good/bad» an output is
  - Often called the *reward* (can be negative or positive)
- The goal of the agent is to learn the behaviour that maximises its expected cumulative reward over time
  - To learn how to flip pancakes, the reward could for instance be +3 if the pancake is flipped, -1 if the pancake stays in the pan, and -5 if it falls