Introduction to
Machine Learning

Mustafa Jarrar

Birzeit University

Watch this lecture
and download the slides

Course Page: http://www.jarrar.info/courses/AI/

More Courses : http://www.jarrar.info/courses

Introduction to
Machine Learning

Mustafa Jarrar

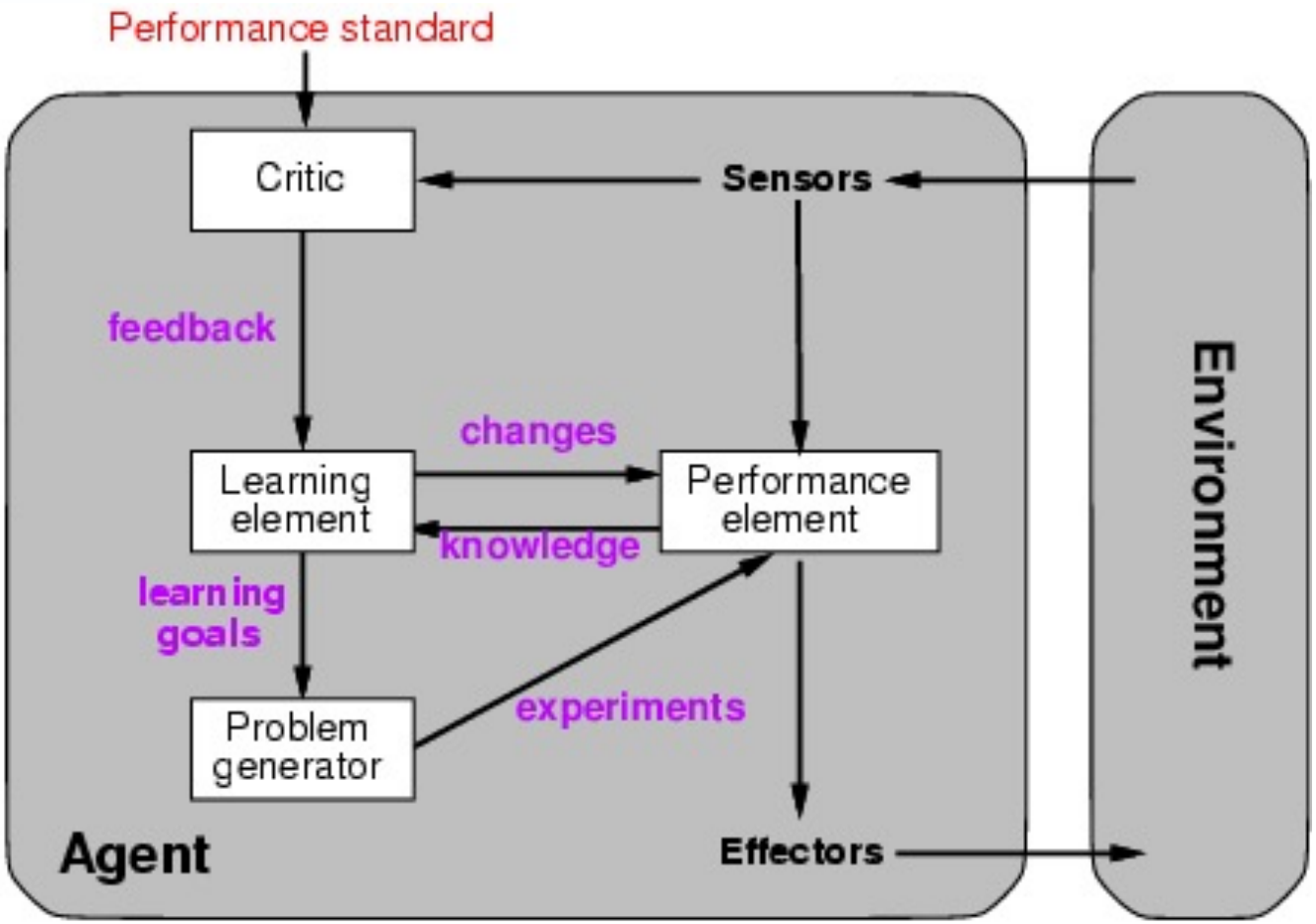
- Part 1: **Introduction and Basics**
- Part 2: **Supervised Learning (Regression, Classification)**
- Part 3: **Unsupervised Learning (Clustering)**
- Part 4: **Reinforcement Learning**

Keywords: Learning, Machine learning, Supervised Learning, unsupervised Learning, Reinforcement learning

Learning Agents

Based on [8]

The agent adapts its action(s) based on feedback (not only sensors).



Introduction

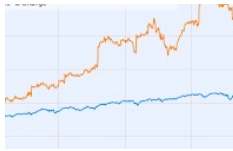
What is Machine Learning?

Field of study that gives computers the ability to learn without being explicitly programed (Arthur Samuel 1959)

Machine Learning is used when [1,2]:

- expertise does not exist (Curiosity Rover)
- we cannot explaining our expertise (Speech Recognition)

- data is too large for us to analyze (Data Mining)



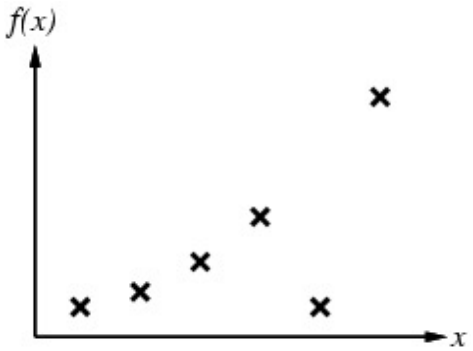
- Prediction of new data (Stock Market Prediction)
- Tasks that are learnt by practicing (Robot Path Planning)

Motivation: Inductive Learning

learn a function from examples (past experience)

f is the **target function**

An **example** is a pair (*x*, *f(x)*)



Problem: find a **hypothesis** *h*
such that $h \approx f$
given a **training set** of examples

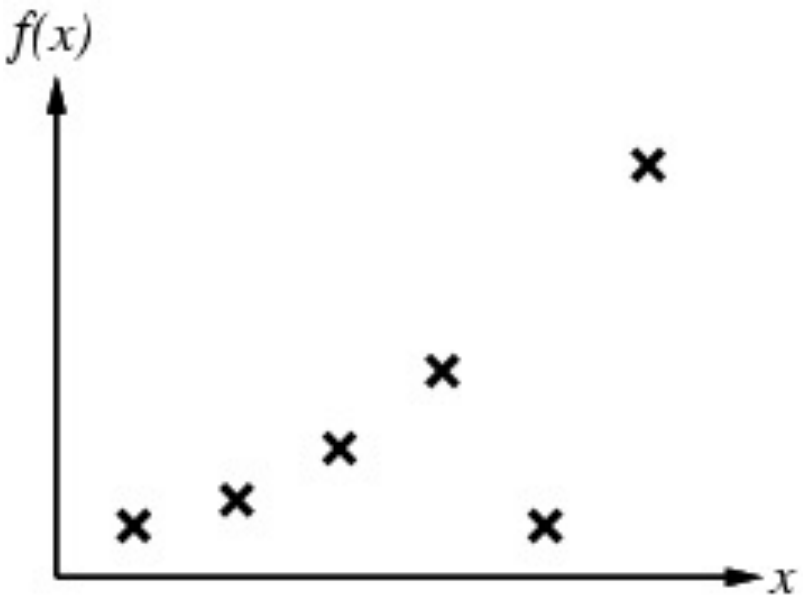
Motivation: Inductive Learning

Based on [8]

Construct/adjust *h* to agree with *f* on training set

(*h* is consistent if it agrees with *f* on all examples)

e.g., curve fitting:



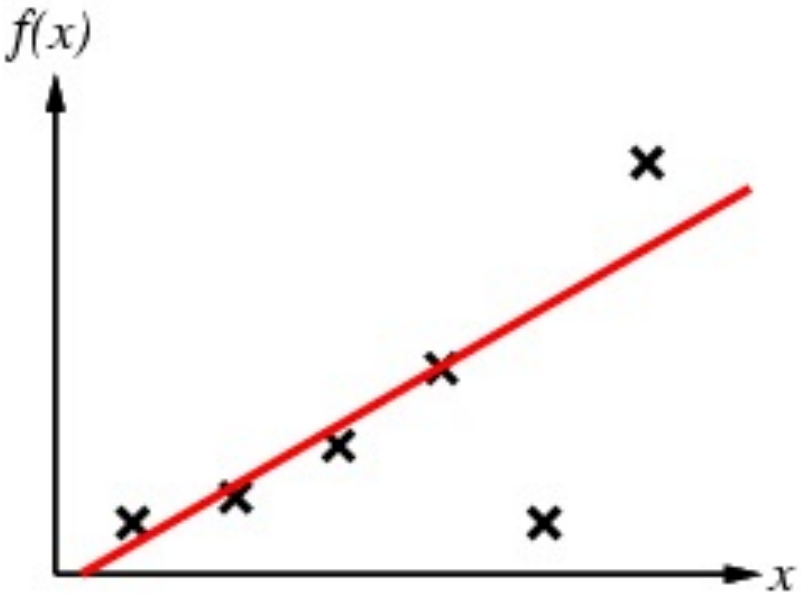
Motivation: Inductive Learning

Based on [8]

Construct/adjust *h* to agree with *f* on training set

(*h* is consistent if it agrees with *f* on all examples)

e.g., curve fitting:



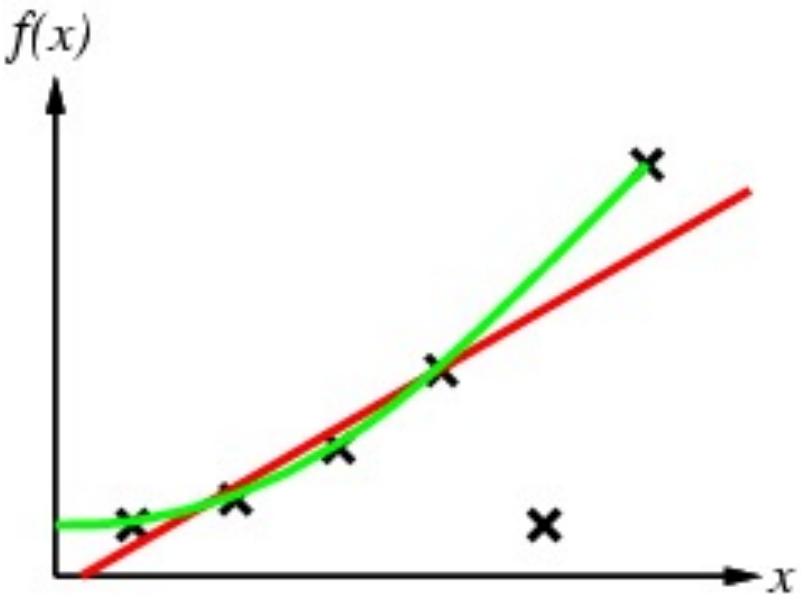
Motivation: Inductive Learning

Based on [8]

Construct/adjust *h* to agree with *f* on training set

(*h* is consistent if it agrees with *f* on all examples)

e.g., curve fitting:



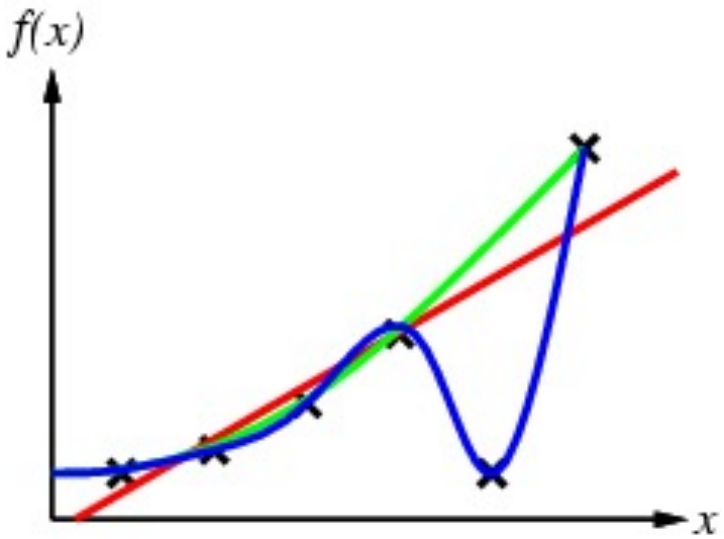
Motivation: Inductive Learning

Based on [8]

Construct/adjust *h* to agree with *f* on training set

(*h* is consistent if it agrees with *f* on all examples)

e.g., curve fitting:



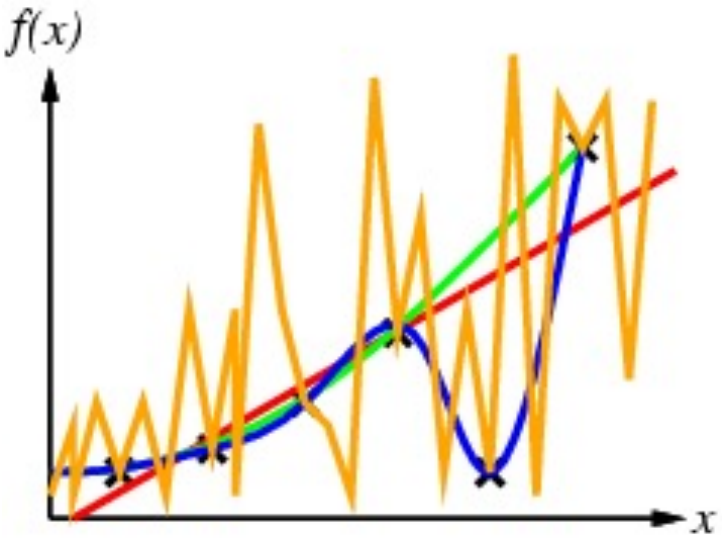
Motivation: Inductive Learning

Based on [8]

Construct/adjust *h* to agree with *f* on training set

(*h* is consistent if it agrees with *f* on all examples)

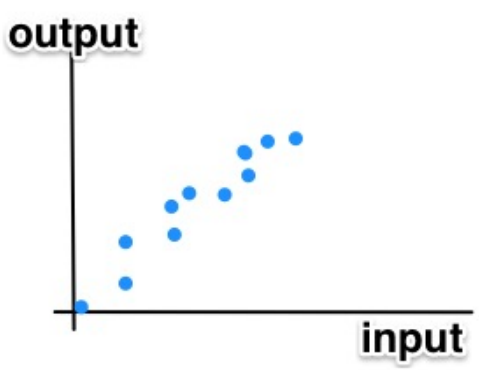
e.g., curve fitting:



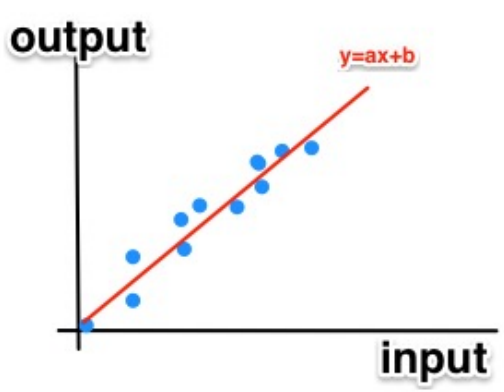
Ockham's razor: prefer the simplest hypothesis consistent with data

What is meant by learning?

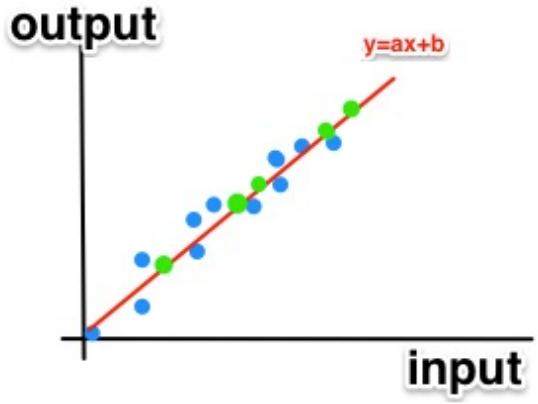
- Writing algorithms that can learn patterns from data.
- The algorithms create a statistical model that is a good approximation of the data.



Data from Past Experiences



Calculating a model



Estimating the output for new input values

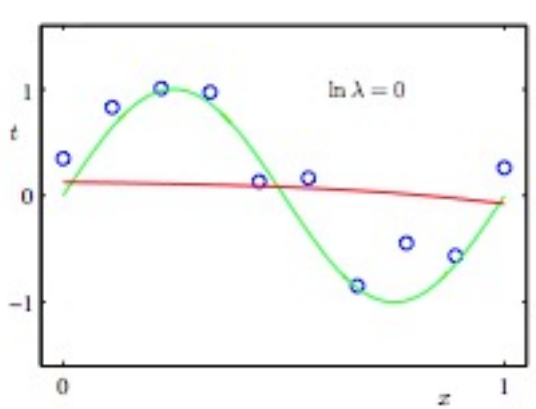
Challenges of Machine Learning

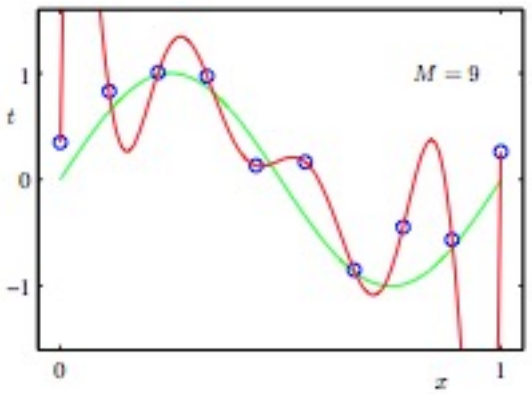
High Dimensionality [3]

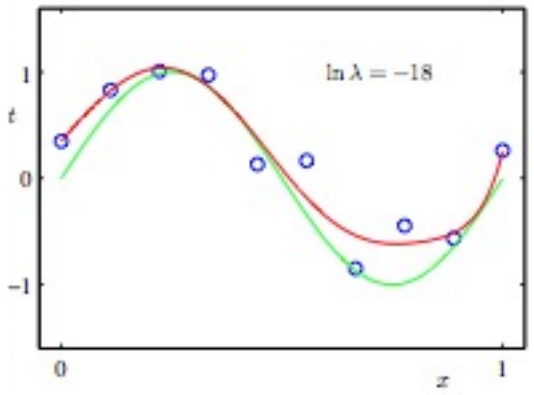
- Complexity of the data requires bigger models.
- Requires bigger of memory and more time to process.
- Might cause over-fitting.

Choice of Statistical Model [4]

- Choosing the correct model and parameters that satisfy the data
- Can cause under-fitting or over-fitting



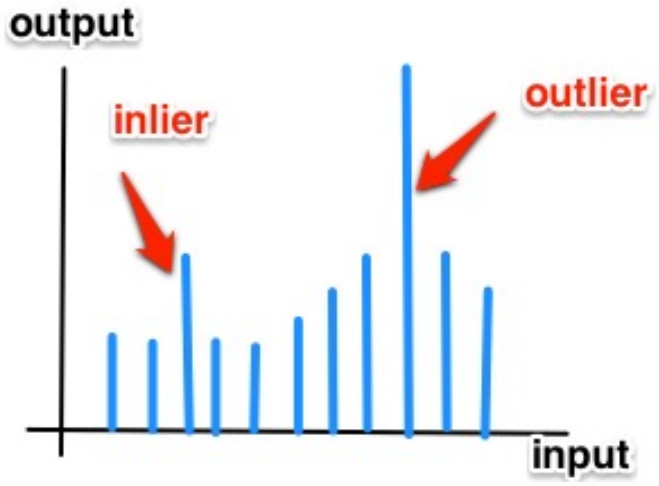




Challenges of Machine Learning

Noise and Errors [5]

- Gaussian Noise: Statistical Noise that has its probability density function equal to normal distribution.
- Outlier: an observation that is distant from the rest of the data.
- Inlier: a local outlier. (see: 2-sigma rule).
- Human Error causing incorrect measurements



Challenges of Machine Learning

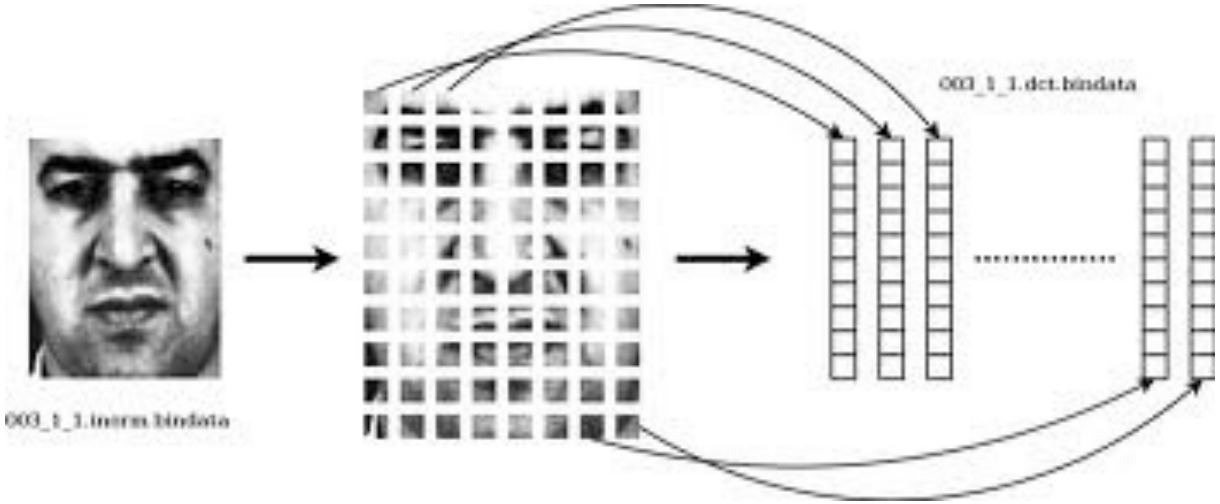
Insufficient Training Data

The amount of data is not sufficient to build a good approximation of the process that generated the data.

Feature Extraction in Patterns

Feature extraction is the process of converting the data to a reduced representation of a set of features.

Image Reference:
Face Verification



Introduction to
Machine Learning

Mustafa Jarrar

- Part 1: **Introduction and Basics**
- Part 2: **Supervised Learning (Regression, Classification)**
- Part 3: **Unsupervised Learning (Clustering)**
- Part 4: **Reinforcement Learning**

Keywords: Learning, Machine learning, Supervised Learning, unsupervised Learning, Reinforcement learning

Supervised Learning (Regression)

Regression aims to estimate a response.

Input: $\{x_1, x_2, \ldots, x_n\}$ *numeric values*, called *features*

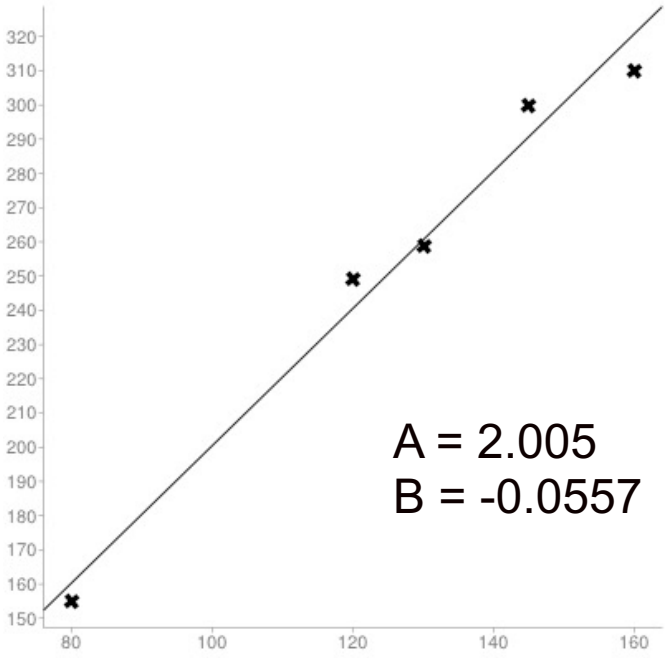
Output: y *numeric values*, called *Target Value*

Toy Problem: areas and prices of apartments (training data).

Find a model to predict prices of unseen cases

features | *Target Value*

Area (m^2)	Price (1000$)
80	155
120	249
130	259
145	300
160	310



➔ 150 ???

Example (Regression)

Suppose we have the following features

x_1	x_2	x_3	x_4	y
Size ft^2	**bedrooms**	**floors**	**Age**	**Price**
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

Linear regression

A hypothesis function $h(x)$ might be:

$$h(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \ldots + \theta_n x_n$$

e.g., $h(x) = 80 + 0.1 \cdot x_1 + 0.01 \cdot x_2 + 3 \cdot x_3 - 2 \cdot x_4$

There are many algorithms to find θs values

One (old) way is called **Normal Equation:** $\boldsymbol{\theta} = (X^T \cdot X)^{-1} \cdot X^T \cdot y$

Example (Regression) –using Normal Equation

x_1	x_2	x_3	x_4	y
Size ft²	bedrooms	floors	Age	Price
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178

$$X = \begin{bmatrix} 1 & 2104 & 5 & 1 & 45 \\ 1 & 1416 & 3 & 2 & 40 \\ 1 & 1534 & 3 & 2 & 30 \\ 1 & 852 & 2 & 1 & 36 \end{bmatrix} \quad Y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$$

In Octave:

```
load X.txt
load y.txt
C= pinv(X'*X)*X' *y
Save θ.txt
```

In R:

```
X = as.matrix(read.table("~/x.txt", header=F, sep=","))
Y = as.matrix(read.table("~/y.txt", header=F, sep=","))
thetas = solve( t(X) %*% X ) %*% t(X) %*% Y
write.table(thetas, file="~/thetas.txt", row.names=F, col.names=F)
```

$$\theta = \begin{bmatrix} 188.4 \\ 0.4 \\ -56 \\ -93 \\ -3.7 \end{bmatrix}$$

These are the values of θs we need to use in our hypothesis function h(x)

$$h(x) = \theta_0 + \theta_1 x + \theta_2 x + \theta_3 x + \theta_4 x$$

$$h(x) = 188.4 + 0.4x - 56x - 93x - 3.7x$$

Supervised Learning (Classification)

Classification aims to identify group membership.

Input: $\{x_1, x_2, \ldots, x_n\}$ *categorical values*, called *features*

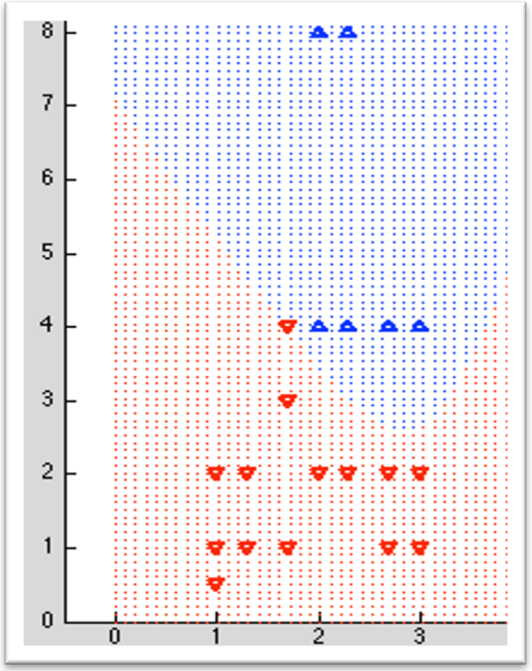
Output: y *categorical values*, called *Target Value*

Toy Problem: data about computers (training data)

Find a model to predict status of unseen cases

features *Target Value*

Processor (GHz)	Memory (GB)	Status
1.0	1.0	Bad
2.3	4.0	Good
2.6	4.0	Good
3.0	8.0	Good
2.0	4.0	Bad
2.6	0.5	Bad
3.0	4.0	???



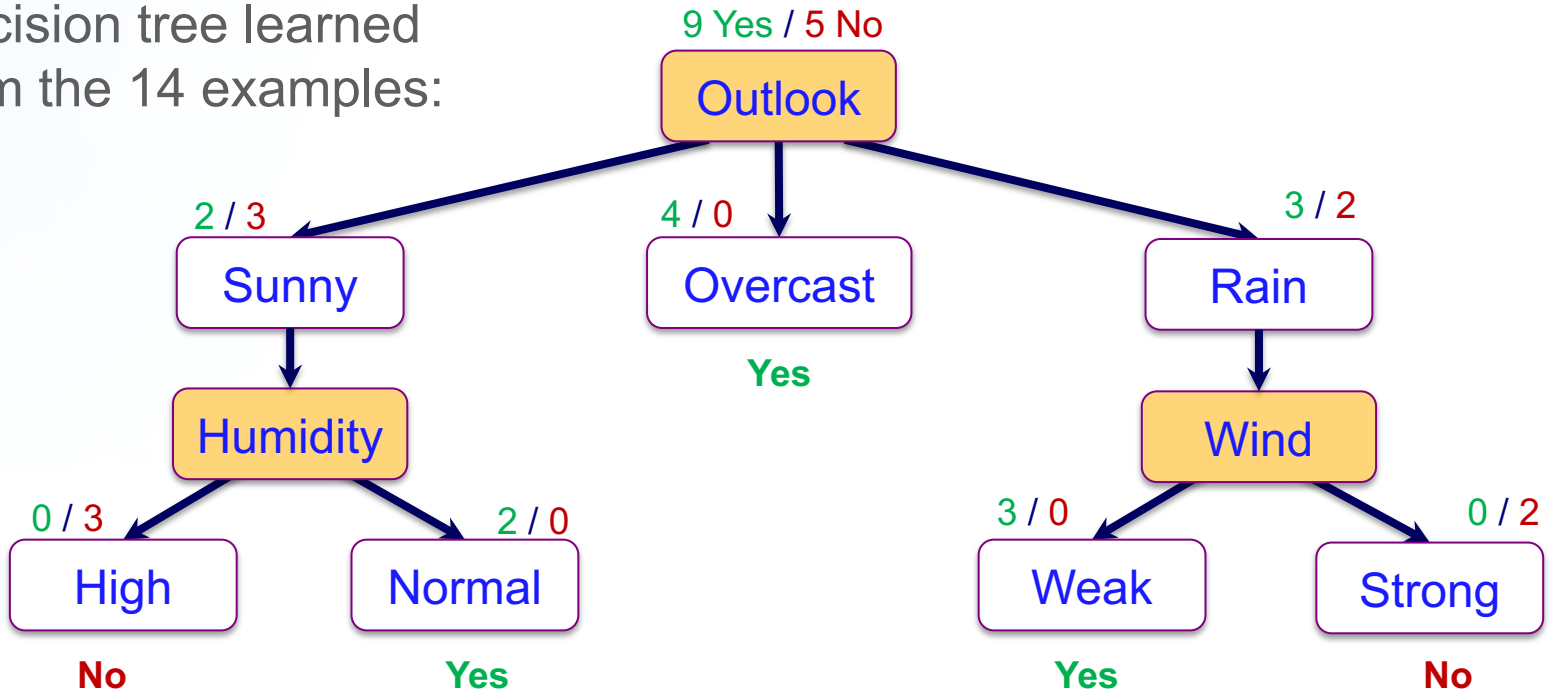
Example (Classification) – using Decision Tree

Given the following training examples, will you play in D15?

Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D3	Overcast	High	Weak	Yes
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No
D7	Overcast	Normal	Strong	Yes
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D10	Rain	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes
D12	Overcast	High	Strong	Yes
D13	Overcast	Normal	Weak	Yes
D14	Rain	High	Strong	No
D15	**Rain**	**High**	**Weak**	**???**

Example (Classification) – using Decision Tree

Decision tree learned from the 14 examples:



Decision Rule:

Yes ⇔ (Outlook=Overcast) ∨
(Outlook=Sunny ∧ Humidity=Normal)∨
(Outlook=Rain ∧ Wind=Weak)

Day	Outlook	Humid	Wind	
D15	Rain	High	Weak	???

➔ Play

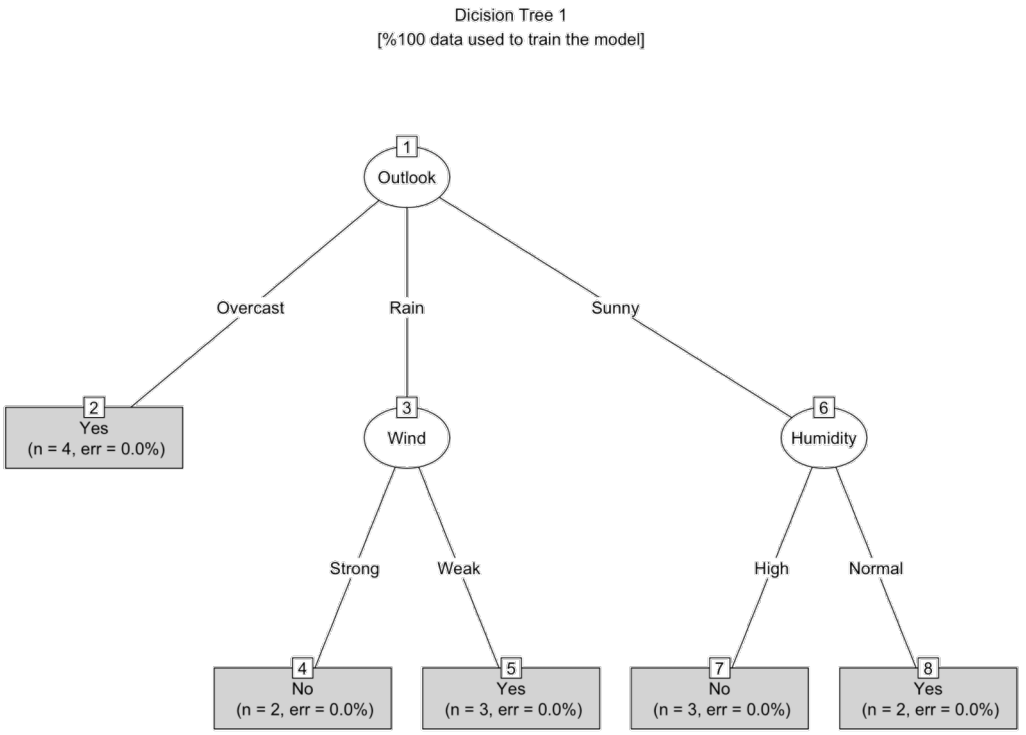
Using R to learn Decision Trees

Input.csv

```
Day,Outlook,Humidity,Wind,Play
D1,Sunny,High,Weak,No
D2,Sunny,High,Strong,No
D3,Overcast,High,Weak,Yes
D4,Rain,High,Weak,Yes
D5,Rain,Normal,Weak,Yes
D6,Rain,Normal,Strong,No
D7,Overcast,Normal,Strong,Yes
D8,Sunny,High,Weak,No
D9,Sunny,Normal,Weak,Yes
D10,Rain,Normal,Weak,Yes
D11,Sunny,Normal,Strong,Yes
D12,Overcast,High,Strong,Yes
D13,Overcast,Normal,Weak,Yes
D14,Rain,High,Strong,No
```

DT_example.R

```
require(C50)
require(gmodels)
dataset = read.table(file.choose(), header = T, sep=",")
dataset = dataset[,-1]
model = C5.0(dataset[, -4], dataset[, 4])
```



Introduction to
Machine Learning

Mustafa Jarrar

- Part 1: **Introduction and Basics**
- Part 2: **Supervised Learning (Regression, Classification)**
- Part 3: **Unsupervised Learning (Clustering)**
- Part 4: **Reinforcement Learning**

Keywords: Learning, Machine learning, Supervised Learning, unsupervised Learning, Reinforcement learning

Unsupervised Learning

- Training data contain only the input vectors (No target value)
- Definition of training data: $\{x_1, x_2, \ldots, x_n\}$ $x \in R^A$
- Goal: Learn some structures in the inputs.
- Can be divided to two categories: Clustering and Dimensionality Reduction

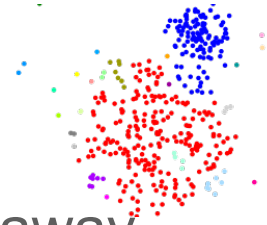
Unsupervised Learning (Clustering)

Clustering aims to group input based on the similarities.

Types of clustering:

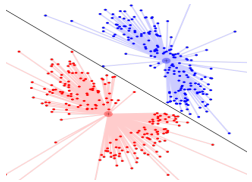
- *Connectivity based clustering*

  objects related to nearby objects than to objects farther away



- *Centroid based clustering*

  cluster points according to a set of given centers

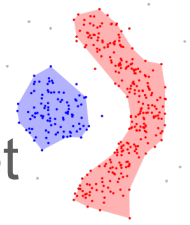


- *Distribution based clustering*

  objects belonging most likely to the same distribution

- *Density based clustering*

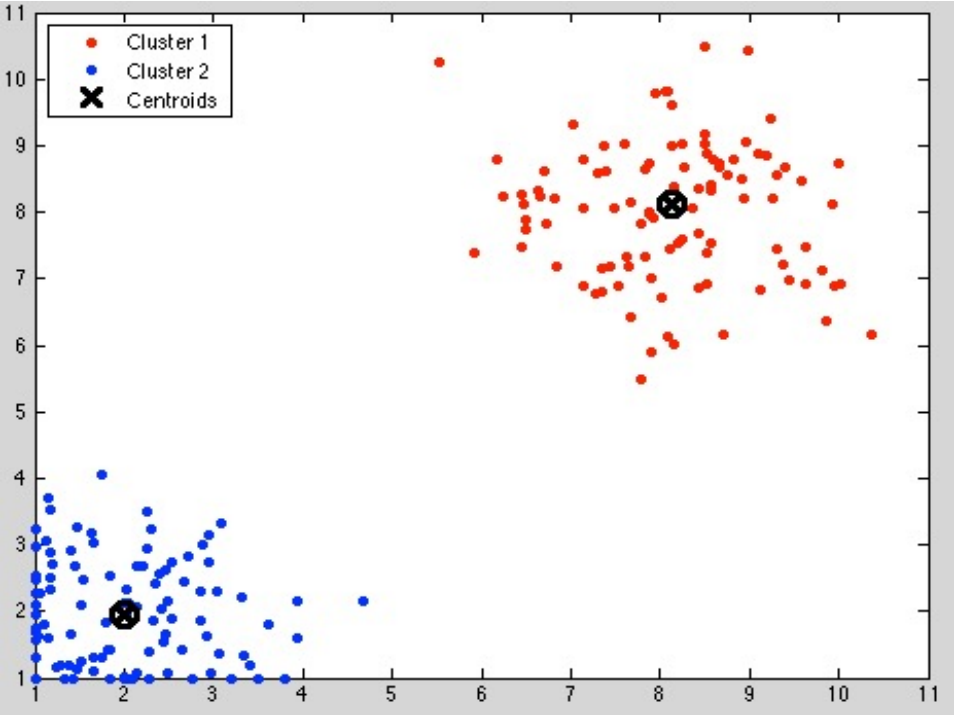
  areas of higher density than the remainder of the data set



Unsupervised Learning (Clustering)

Toy Example: A survey that with questions on a scale 1-10:

- How much do you like shopping?
- How much are you willing to spend on shopping?



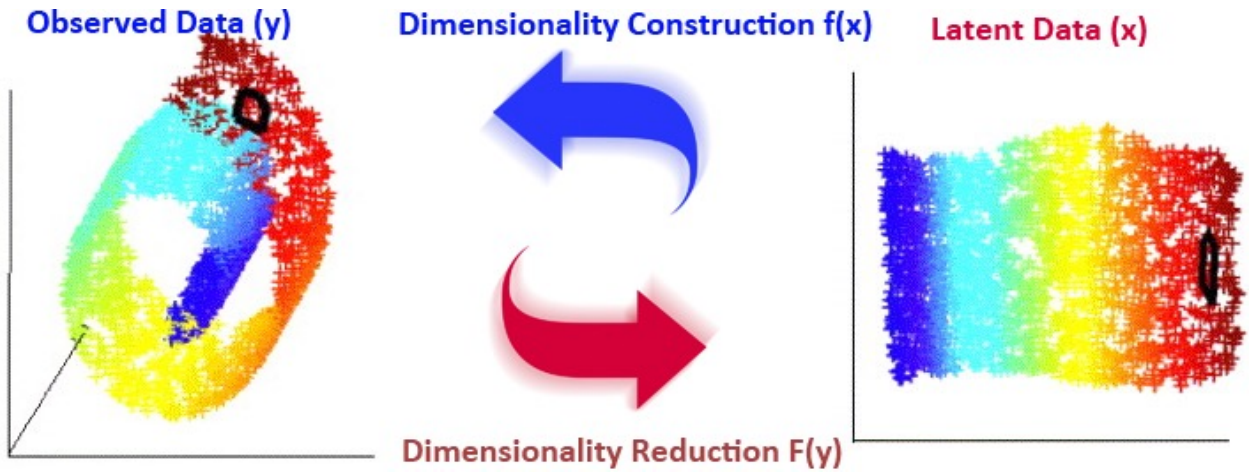
Cluster 1 can refer to people who are addicted to shopping

Cluster 2 can refer to people who rarely go shopping

Unsupervised Learning

Dimensionality Reduction [7]

- Convert high dimensional data to lower order dimension
- Motivation:
  - High Dimensional Data Analysis
  - Visualization of high-dimensional data
  - Feature Extraction



Introduction to
Machine Learning

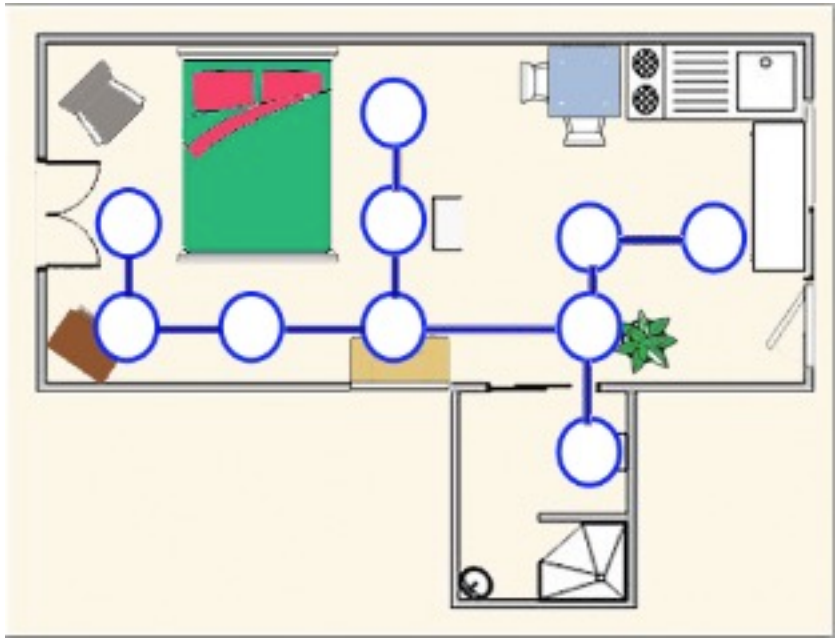
Mustafa Jarrar

- Part 1: **Introduction and Basics**
- Part 2: **Supervised Learning (Regression, Classification)**
- Part 3: **Unsupervised Learning (Clustering)**
- Part 4: **Reinforcement Learning**

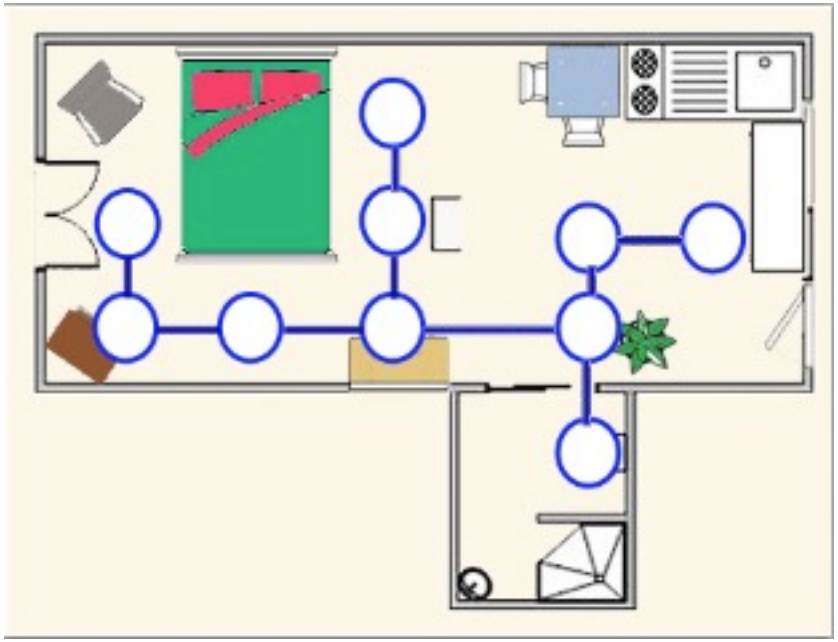
Keywords: Learning, Machine learning, Supervised Learning, unsupervised Learning, Reinforcement learning

Reinforcement Learning

- Learning a policy: a sequence of outputs [1].
- Delayed reward instead of supervised output.
- Toy Example: A robot wants to move from the outer door of an apartment to the bathroom to clean it.

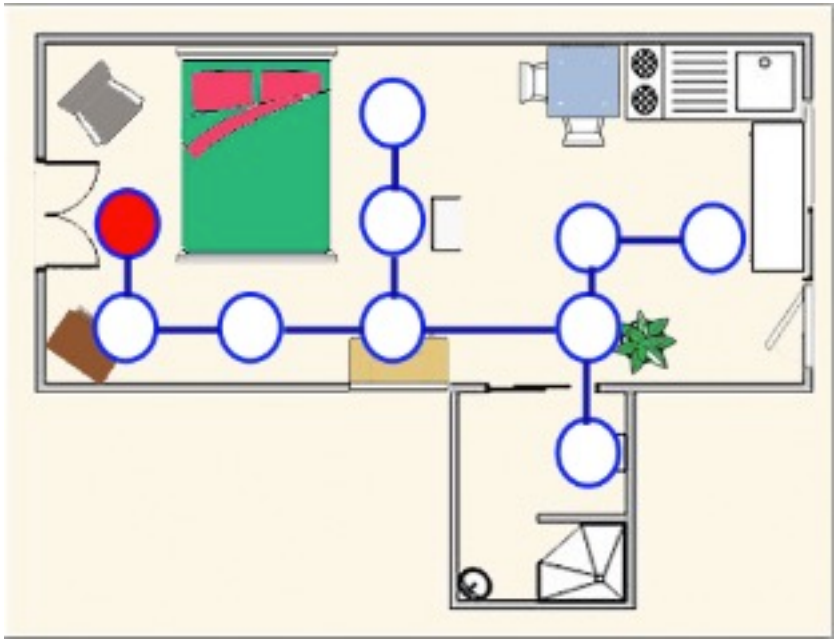


Reinforcement Learning

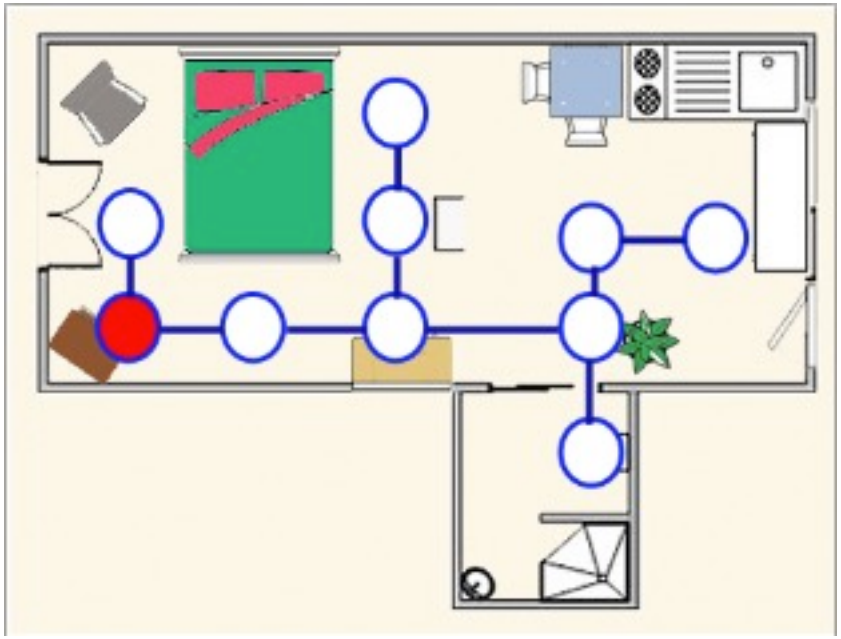


All weights are equal at the first try. Choice of next state is randomly chosen if the weights are equal

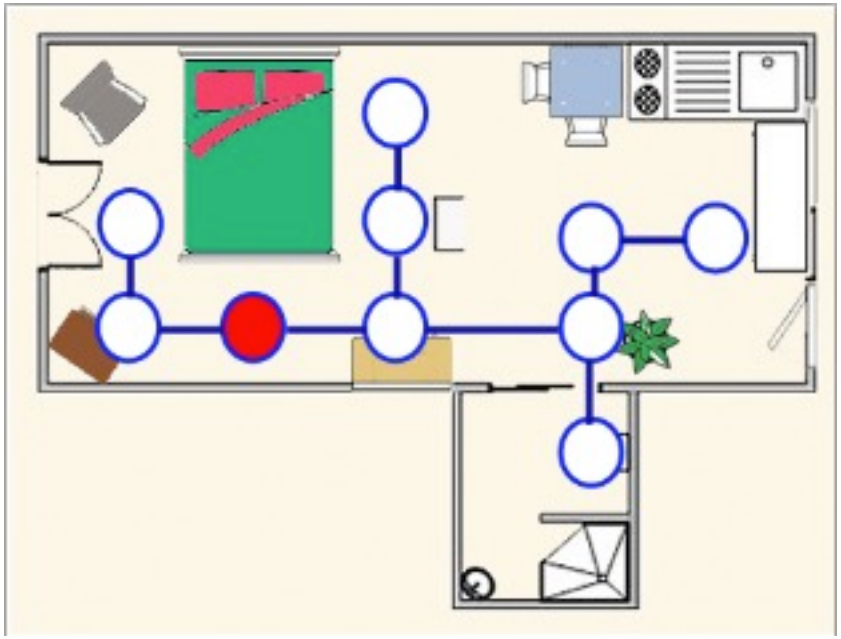
Reinforcement Learning



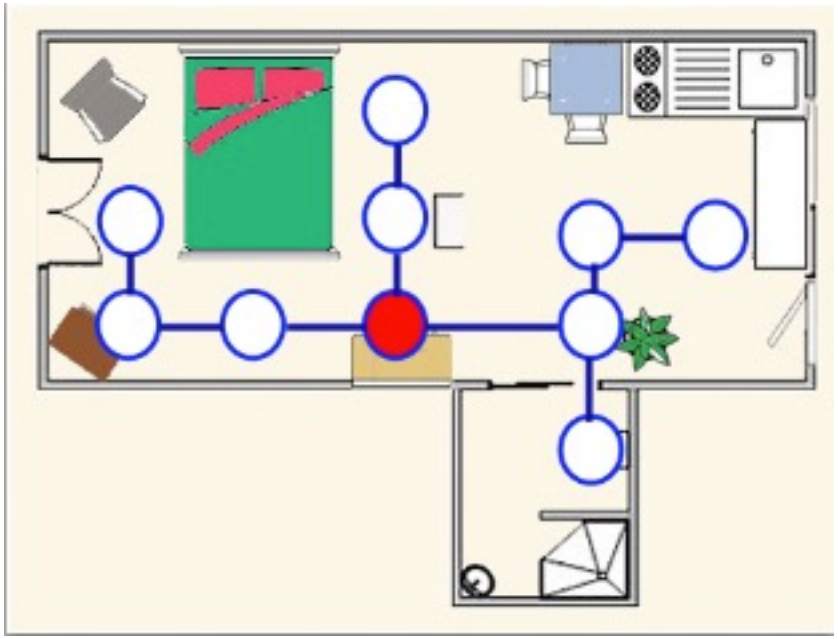
Reinforcement Learning



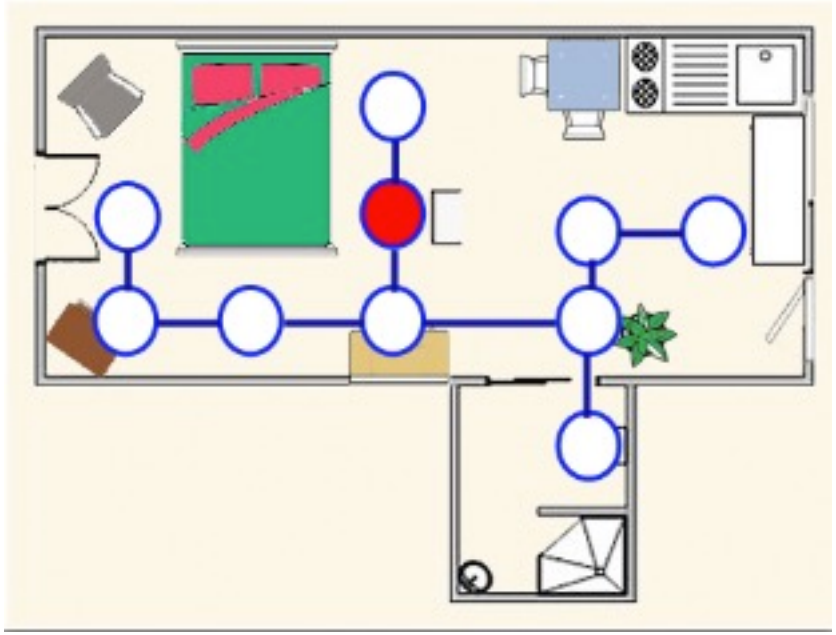
Reinforcement Learning



Reinforcement Learning

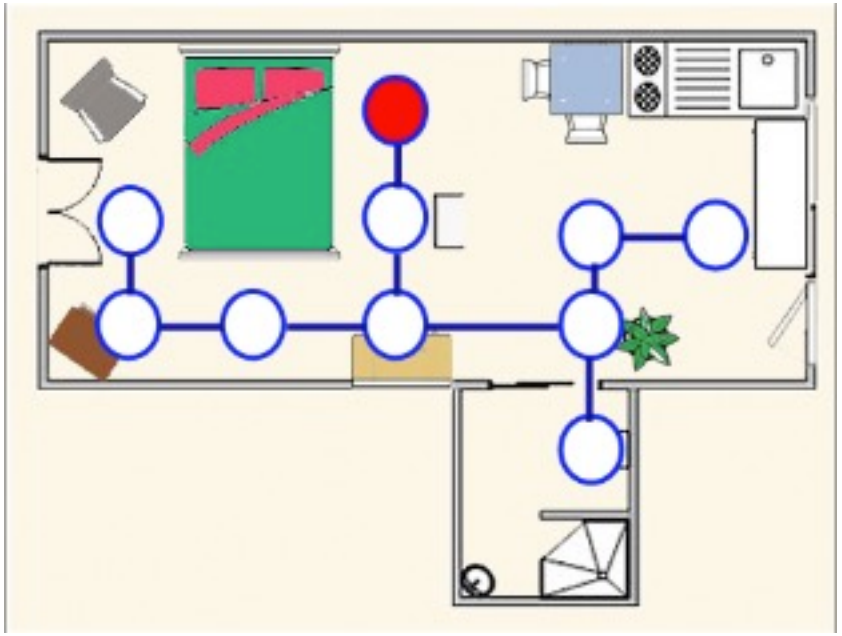


Reinforcement Learning



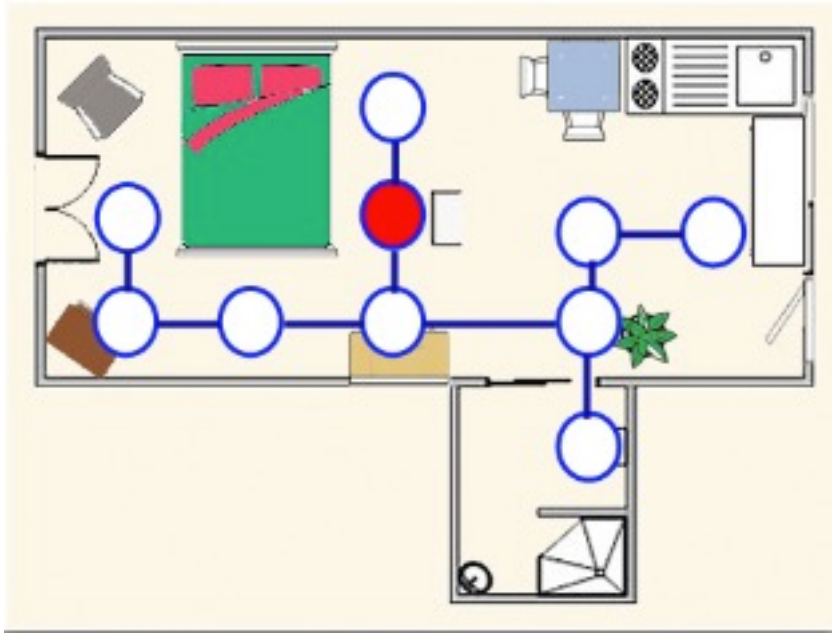
Left is chosen randomly since the weights are equal

Reinforcement Learning

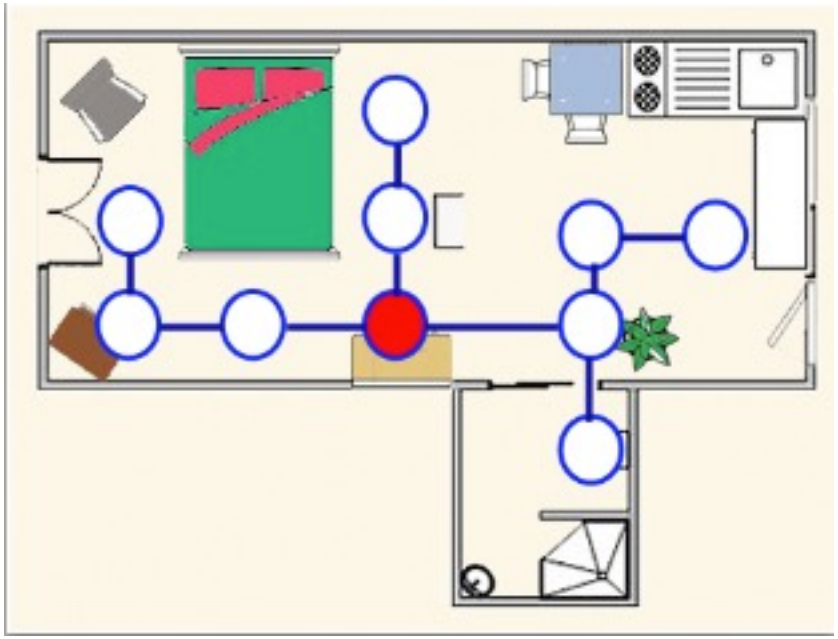


Wrong Destination.
Return by backtracking

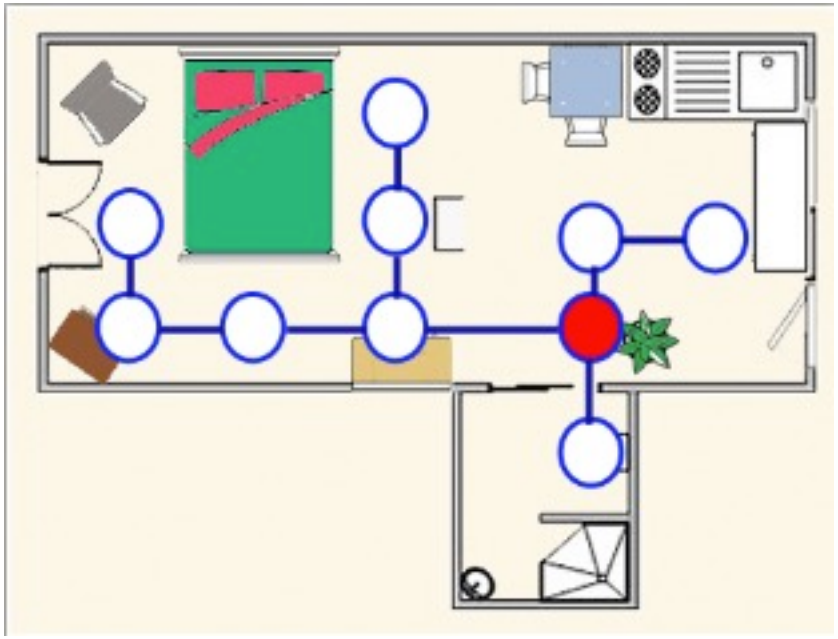
Reinforcement Learning



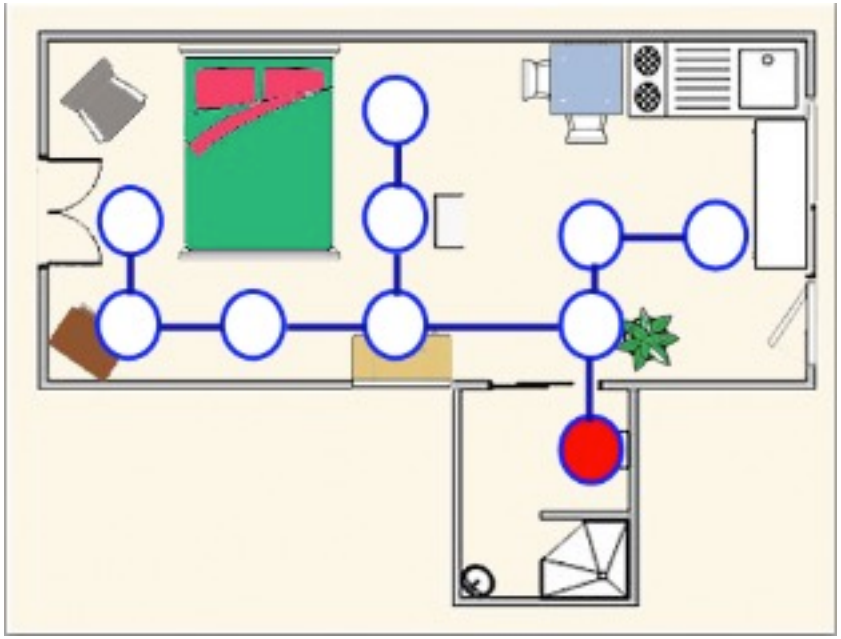
Reinforcement Learning



Reinforcement Learning

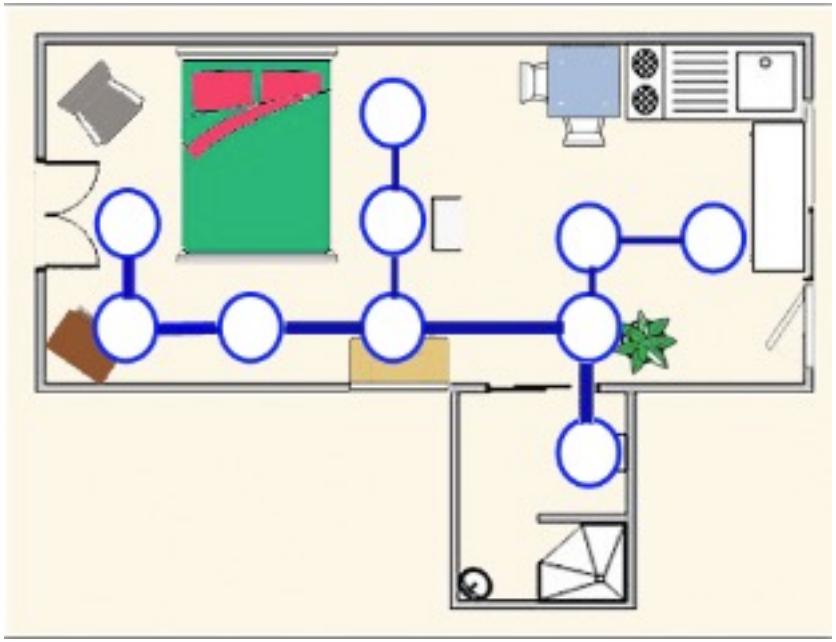


Reinforcement Learning



Reached the destination. Give a reward to the chosen paths by increasing the weight.

Reinforcement Learning



Adjusted weights after reinforcement learning.

Other Learning Paradigms

- Semi-Supervised Learning (Wikipedia)
- Active Learning (Wikipedia)
- Inductive Transfer/Learning (Wikipedia)

Real World Examples

Machine Learning in Real-World Examples: [6]

- Spam Filter
- Signature Recognition
- Credit Card Fraud Detection
- Face Recognition
- Text Recognition
- Speech Recognition
- Speaker Recognition
- Weather Prediction
- Stock Market Analysis
- Advertisement Targeting
- Language Translation
- Recommendation Systems
- Classifying DNA Sequences
- Automatic vehicle Navigation
- Object Detection
- Medical Diagnosis

Online Courses and Material

- Interactive Course with Stanford University Professor
  - Website: https://www.coursera.org/course/ml

- Stanford University Class
  - Playlist: http://www.youtube.com/view_play_list?p=A89DCFA6ADACE599
  - Material: http://cs229.stanford.edu/

References

1. E. Alpaydin, Introduction to Machine Learning. Cambridge, MA: MIT Press, 2004.
2. T. Mitchell, Machine Learning. McGraw Hill, 1997.
3. Liu H, Han J, Xin D, Shao Z (2006) Mining frequent patterns on very high dimensional data: a top-down row enumeration approach. In: Proceeding of the 2006 SIAM international conference on data mining (SDM'06), Bethesda, MD, pp 280–291.
4. C. Bishop. Pattern Recognition and Machine Learning. Springer, 2006.
5. T. Runkler. Data Analytics. Springer, 2012.
6. http://www.cs.utah.edu/~piyush/teaching/23-8-slides.pdf
7. Carreira-Perpinan, M., Lu, Z.: Dimensionality Reduction by Unsupervised Learning Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference, 1895 -1902 13 June 2010 .
8. S. Russell and P. Norvig: *Artificial Intelligence: A Modern Approach* Prentice Hall, *2003, Second Edition*
9. Sami Ghawi, Mustafa Jarrar: Lecture Notes on Introduction to Machine Learning, Birzeit University, 2018
10. Mustafa Jarrar: Lecture Notes on Decision Tree Machine Learning, Birzeit University, 2018
11. Mustafa Jarrar: Lecture Notes on Linear Regression Machine Learning, Birzeit University, 2018