# COMP 472: Artificial Intelligence
# Machine Learning part 2
# Introduction video #1

- Russell & Norvig: Sections 19.1 - 19.2

# Next Set of Videos

1. Introduction to ML video #1
2. Naïve Bayes Classification
   a. Application to Spam Filtering
3. Decision Trees
4. ( Evaluation
5. Unsupervised Learning )
6. Neural Networks
   a. Perceptrons
   b. Multi Layered Neural Networks

https://medium.com/machine-learning-for-humans/neural-networks-deep-learning-cdad8aeae49b

# Today

1. Introduction to ML 

2. Naïve Bayes Classification
   a. Application to Spam Filtering
3. Decision Trees
4. ( Evaluation
5. Unsupervised Learning )
6. Neural Networks
   a. Perceptrons
   b. Multi Layered Neural Networks

https://medium.com/machine-learning-for-humans/neural-networks-deep-learning-cdad8aeae49b

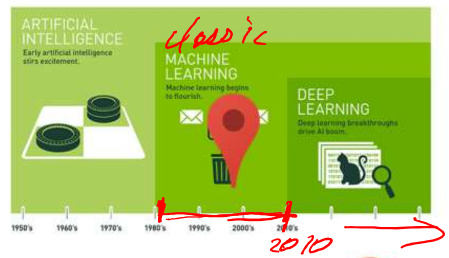
# Remember this slide…

## History of AI

- 1980s-2010
- The rise of Machine Learning
  - More powerful CPUs-> usable implementation of neural networks
  - Big data -> Huge data sets are available
    - document repositories for NLP (e.g. emails)
    - billions on images for image retrieval
    - billions of genomic sequences, …

😃 Rules are now learned automatically !

but Domain Experts still help us determine which features might be relevant



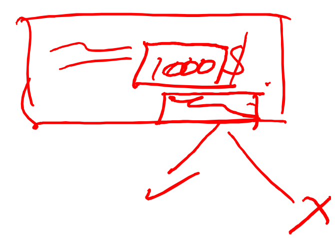


2011: Watson wins at Jeapardy!

40

# Motivation

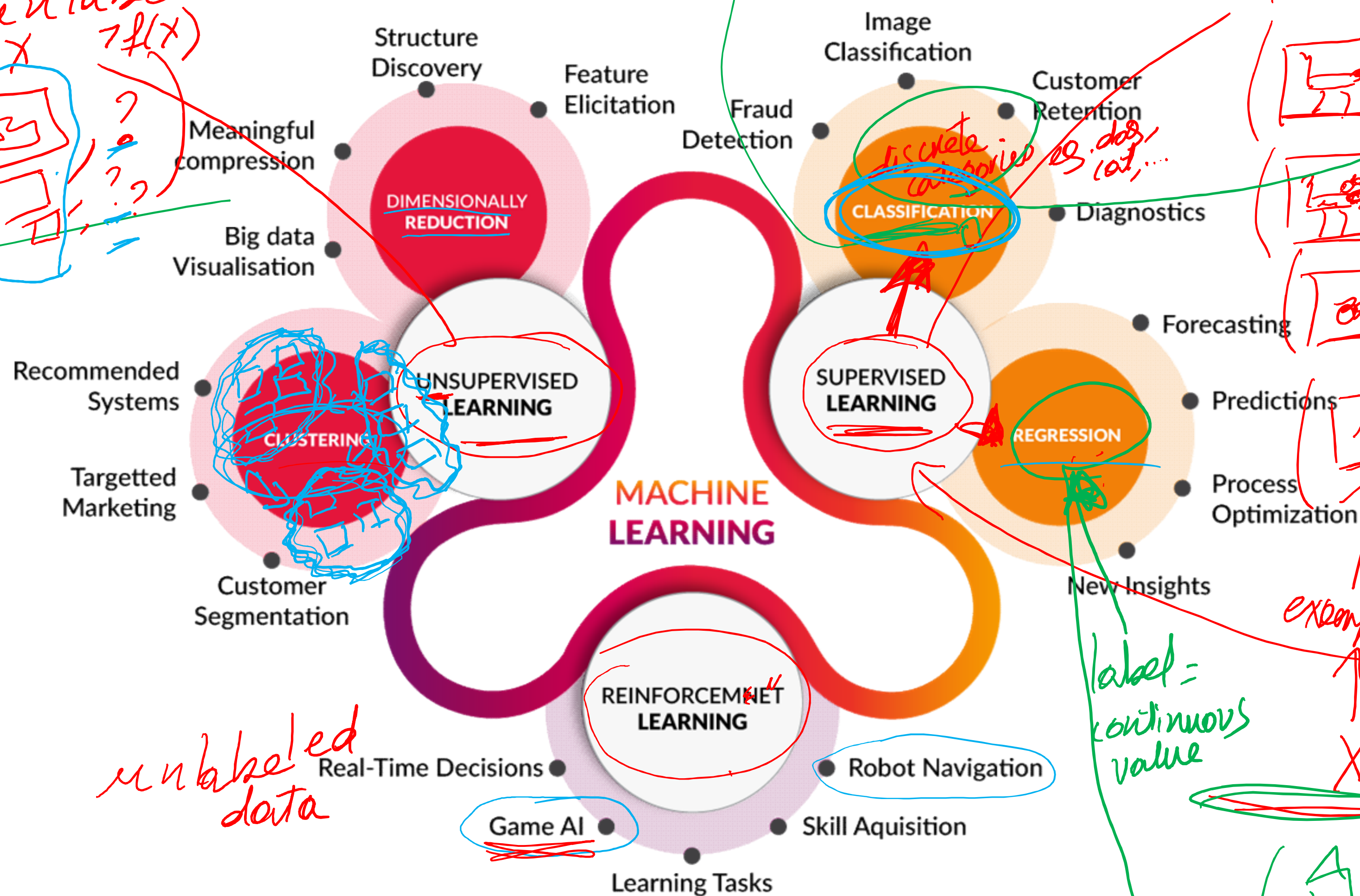
- Too many to list here!
  - Recommender systems (eg. Netflix)
  - Pattern Recognition (eg. Handwriting recognition)
  - Detecting credit card fraud
  - Computer vision (eg. Object recognition)
  - Discovering Genetic Causes of Diseases
  - Natural Language Processing (eg. Spam filtering)
  - Speech Recognition / Synthesis
  - Medical Diagnostics
  - Information Retrieval (eg. Image search)
  - Learning heuristics for game playing
  - …
  - *Oh… I'm out of space*



# What is Machine Learning?

- Learning = crucial characteristic of an intelligent agent
- ML
  - Constructs algorithms that learn from data
  - ie perform tasks that were not explicitly programmed and improve their performance the more tasks they accomplish
  - generalize from given experiences and are able to make judgments in new situations

# Types of Machine Learning




unlabeled data

$x$ ? $f(x)$

a.k.a categorization

discrete categories e.g. dog, cat, ...

labeled data

dog

cat

dog

cat

example label

$x$ $f(x)$

label = continuous value

375,000$

299,000$

unlabeled data

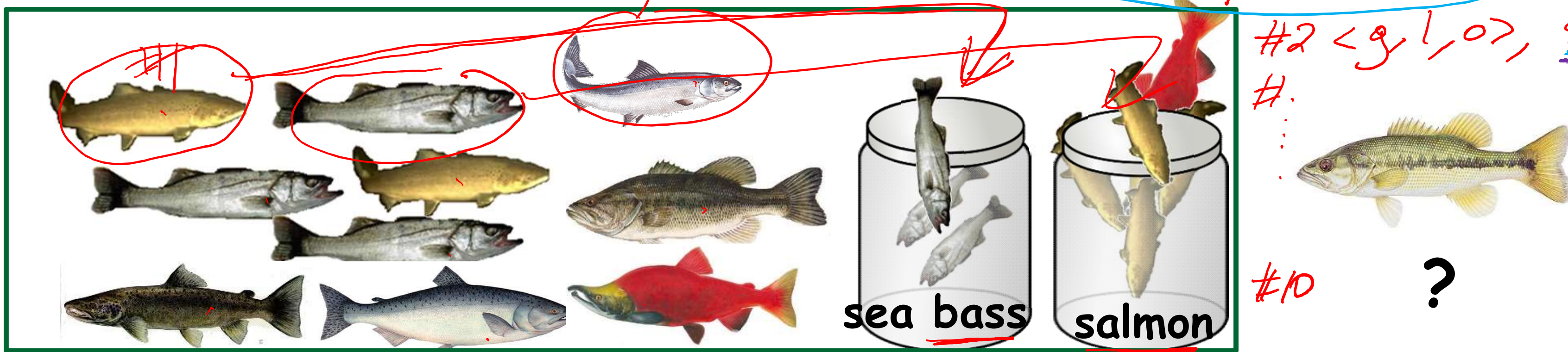
http://www.cognub.com/index.php/cognitive-platform/

# Types of Machine Learning

- Supervised learning
  - We are given a training set of (X, f(X)) pairs
  - X = <color, length>

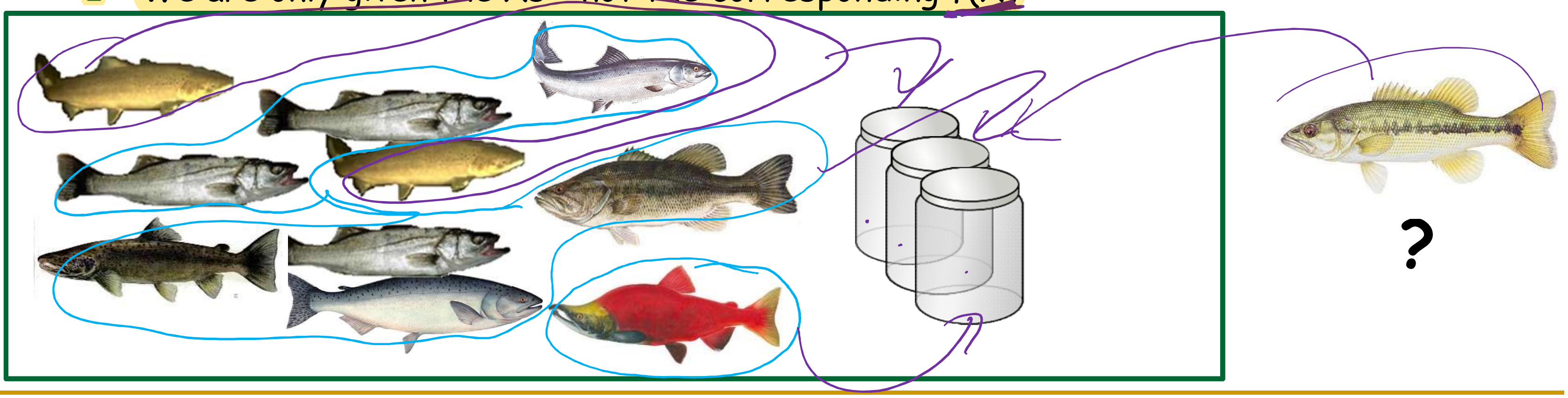
example → correct labeled

training set
fish #1 <y, m, c> bass




#2 <g, l, o>, salmon
#.
#10

- Unsupervised learning
  - We are only given the Xs - not the corresponding f(X)



# Types of Learning

- In **Supervised** learning
  - We are given a training set of (X, f(X)) pairs

X

$f(x) \equiv$ label

| | | | | |
|---|---|---|---|---|
| big nose | big teeth | big eyes | no moustache | f(X) = not person |
| small nose | small teeth | small eyes | no moustache | f(X) = person |

| | | | | |
|---|---|---|---|---|
| small nose | big teeth | small eyes | moustache | f(X) = ? |

- In **Reinforcement** learning
  - We are not given the (X, f(X)) pairs

| | | | | |
|---|---|---|---|---|
| small nose | big teeth | small eyes | moustache | f(X) = ? |

  - But we get a reward when our learned f(X) is right, and we try to maximize the reward
  - Goal: maximize the nb of right answers

- In **Unsupervised** learning
  - We are only given the Xs - not the corresponding f(X)

$\neq$ given $f(x)$

| | | | | |
|---|---|---|---|---|
| big nose | big teeth | big eyes | no moustache | *not given* |
| small nose | small teeth | small eyes | no moustache | *not given* |

| | | | | |
|---|---|---|---|---|
| small nose | big teeth | small eyes | moustache | f(X) = ? |

bill → s.a.h / elder / young

  - No *teacher* involved / Goal: find regularities among the Xs (clustering)
  - Data mining

# Logical Inference

to reason

- Inference: process of deriving new facts from a set of premises

- Types of logical inference:
  1. Deduction
  2. Abduction
  3. Induction ≡ ML from examples

# Deduction

- aka Natural Deduction
- Conclusion follows necessary from the premises.
- From $A \Rightarrow B$ and $A$, we conclude that $B$
- We conclude from the general case to a specific example of the general case
- Ex:

known to be true

*All men are mortal.*
*Marcus is a man.*

---

*Marcus is mortal.*

conclude

$\forall x \; man(x) \Rightarrow mortal(x)$

$man(marcus)$

---

$mortal(marcus)$

# Abduction

- Conclusion is one hypothetical (most probable) explanation for the premises
- From $A \Rightarrow B$ and B, we conclude A
- Ex:

premises

*Drunk people do not walk straight.* $\forall x\ drunk(x) \Rightarrow \neg walkS(x)$

*John does not walk straight.* $\neg walkS(john)$

---

*John is drunk.* $\Rightarrow drunk(john)$

- Not sound… but may be most likely explanation for B
- Used in medicine…
  - in reality… disease $\Rightarrow$ symptoms
  - patient complains about some symptoms… doctor concludes a disease

# Induction

- Conclusion about all members of a class from the examination of only a few members of the class.
- From $A \wedge C \Rightarrow B$ and $A \wedge D \Rightarrow B$, we conclude $A \Rightarrow B$
- We construct a general explanation based on a specific case.
- Ex:

premise *All CS students in COMP 472 are smart.* $\forall x\ cs(x) \wedge comp472(x) \Rightarrow smart(x)$.

*All CS students on vacation are smart.* $\forall x\ cs(x) \wedge vacation(x) \Rightarrow smart(x)$.

conclusion *All CS students are smart.* $\forall x\ cs(x) \Rightarrow smart(x)$

feature feature feature

- Not sound
- But, can be seen as hypothesis construction or generalisation

# Inductive Learning

- = learning from examples
- Most work in ML
- Training set is given:
  - that includes examples of already classified examples
  - i.e. pairs of (X, f(X))
  - Ex:

supervised learning

f(x)

| | | | | |
|---|---|---|---|---|
| small nose | big teeth | small eyes | moustache | tiger |

  - used to "train" a model
- Test set: given:
  - that includes examples of non classified examples
  - for which the model must to make accurate predictions

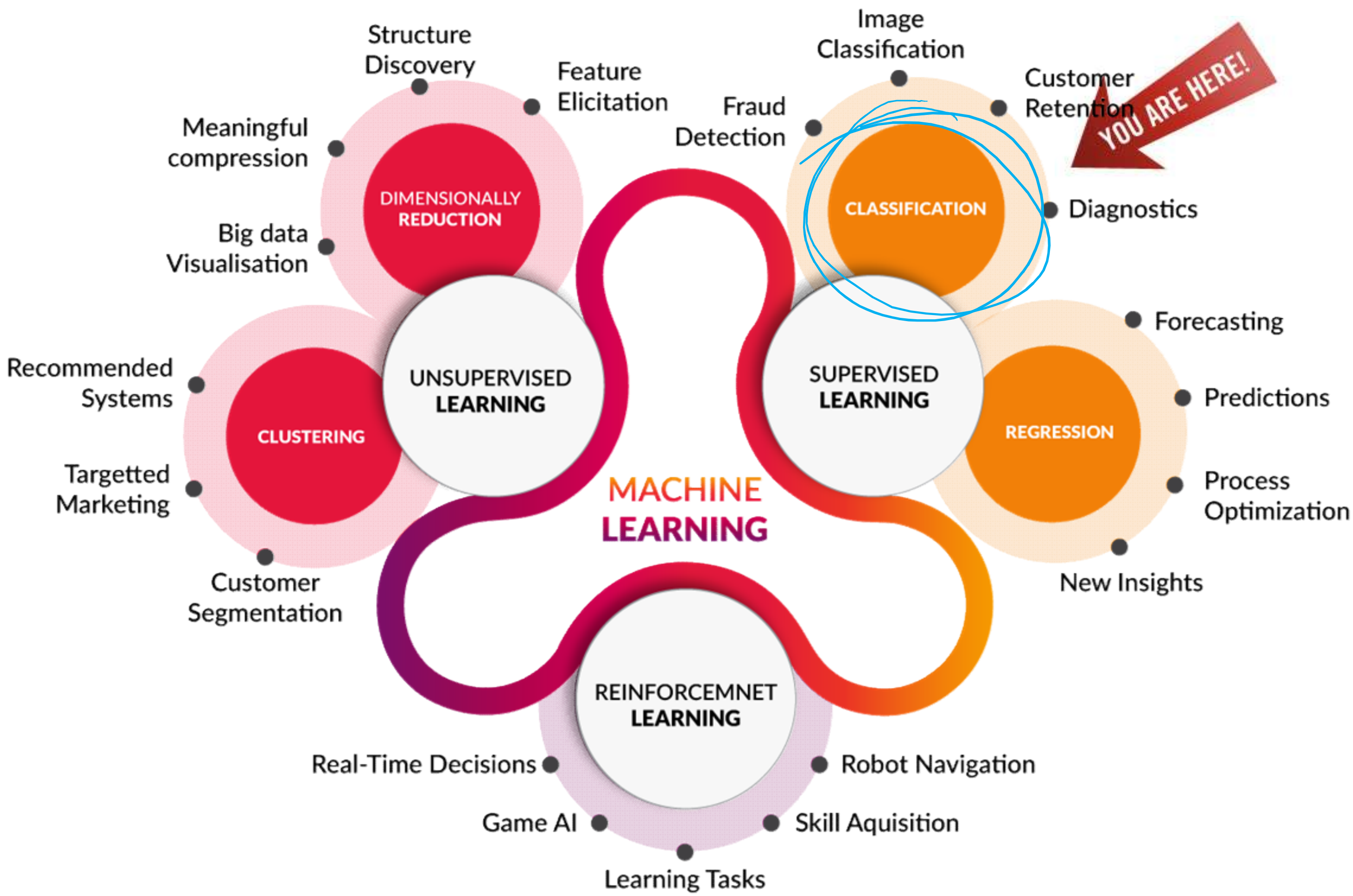
model will need to make its prediction

| | | | | |
|---|---|---|---|---|
| big nose | big teeth | small eyes | moustache | ? |

X

- Can be seen as learning a function
  - the model uses the training set to find an estimate of the function f(X) given X
  - so that given a new instance X it has never seen before, the model can predict f(X)

# Types of Machine Learning




http://www.cognub.com/index.php/cognitive-platform/

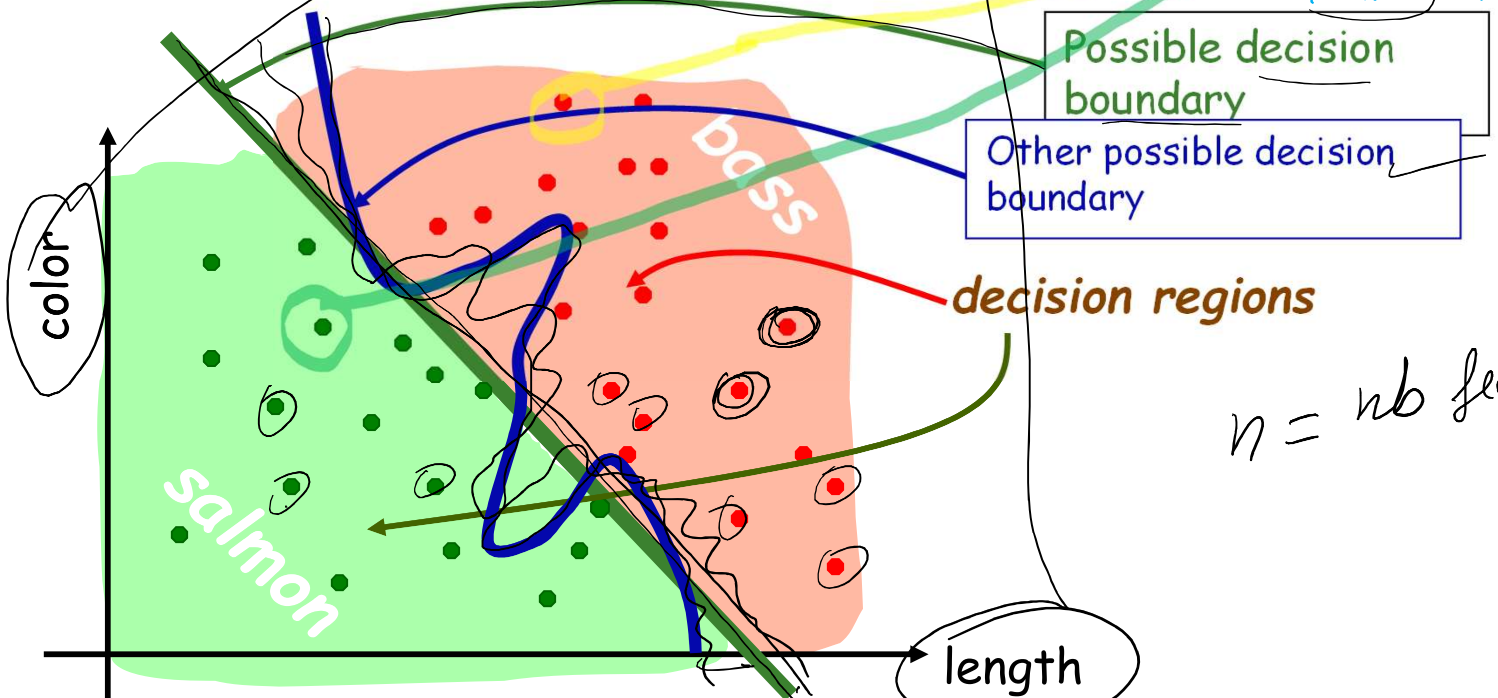
# Example

(<color, length>, bass salmon f(X))

X

#1 (<s, m>, bass)
#2 (<g, l>, salmon)
#3
#1000

- Given pairs (X,f(X)) (the training set – the data points)
- Find a function that fits the training set well




n = nb features.

- Note: choosing one function over another beyond just looking at the training set is called **inductive bias** (eg. prefer "smoother" functions)

# Inductive Learning Framework

- Input data are represented by a vector of features, X
- Each vector X is a list of (attribute, value) pairs.
  - Ex: x = [nose:big, teeth:big, eyes:big, moustache:no]

feature — value of feature

- The number of attributes is fixed (positive, finite)
- Values can be
  1. categorical (aka nominal) - unordered values
     - eg. "convertible", "minivan", "SUV", ... "yes", "no"
  2. ordinal - ordered values
     - e.g., "large", "medium" or "small"
  3. numerical – ordered
     - e.g. frequency of a word in an email (40, 55)
     - e.g. height of a person (1.7, 1.6)
- Each example X can be interpreted as a point in a n-dimensional feature space, where n is the number of attributes

feature : Value
TypeofCar : - convert
- minivan
- SUV

Note: *attribute* == *feature*

# Example

feature

X

f(x)

| has-hair? | has-scales? | has-feathers? | flies? | lives in water? | lays eggs? | |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | Dog |
| 1 | 0 | 0 | 0 | 0 | 0 | Cat |
| 1 | 0 | 0 | 1 | 0 | 0 | Bat |
| 1 | 0 | 0 | 0 | 1 | 0 | Whale |
| 0 | 0 | 1 | 1 | 0 | 1 | Canary |
| 0 | 0 | 1 | 1 | 0 | 1 | Robin |
| 0 | 0 | 1 | 1 | 0 | 1 | Ostrich |
| 0 | 1 | 0 | 0 | 0 | 1 | Snake |
| 0 | 1 | 0 | 0 | 0 | 1 | Lizard |
| 0 | 1 | 0 | 0 | 1 | 1 | Alligator |

instance#1

#2

#10

Real ML applications typically require hundreds, thousands or millions of examples

*source: Alison Cawsey: The Essence of AI (1997).*

# Techniques in ML

- Probabilistic Methods
  - output a probability associated with the result of the classification prediction
- Decision Trees
  - Use only discriminating features as questions in a big if-then-else tree
- Neural networks
  - Also called parallel distributed processing or connectionist systems
  - Intelligence arise from having a large number of simple computational units 1 neuron

- ...

# Today

1. Introduction to ML ✓
2. Naïve Bayes Classification
   a. Application to Spam Filtering
3. Decision Trees
4. ( Evaluation
5. Unsupervised Learning )
6. Neural Networks
   a. Perceptrons
   b. Multi Layered Neural Networks

# Up Next

1. Introduction to ML
2. Naïve Bayes Classification – video #2
   a. Application to Spam Filtering – video #3
3. Decision Trees
4. ( Evaluation
5. Unsupervised Learning )
6. Neural Networks
   a. Perceptrons
   b. Multi Layered Neural Networks