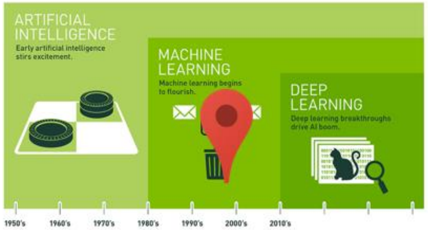
Artificial Intelligence: Naive Bayes Classification

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 !

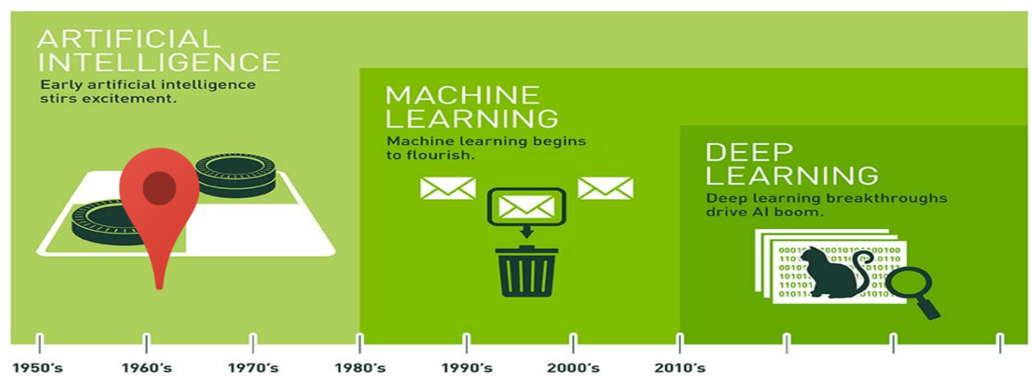


2011: Watson wins at Jeapardy!

40

Today

1. Introduction to ML
2. Naïve Bayes Classifier
3. Evaluation

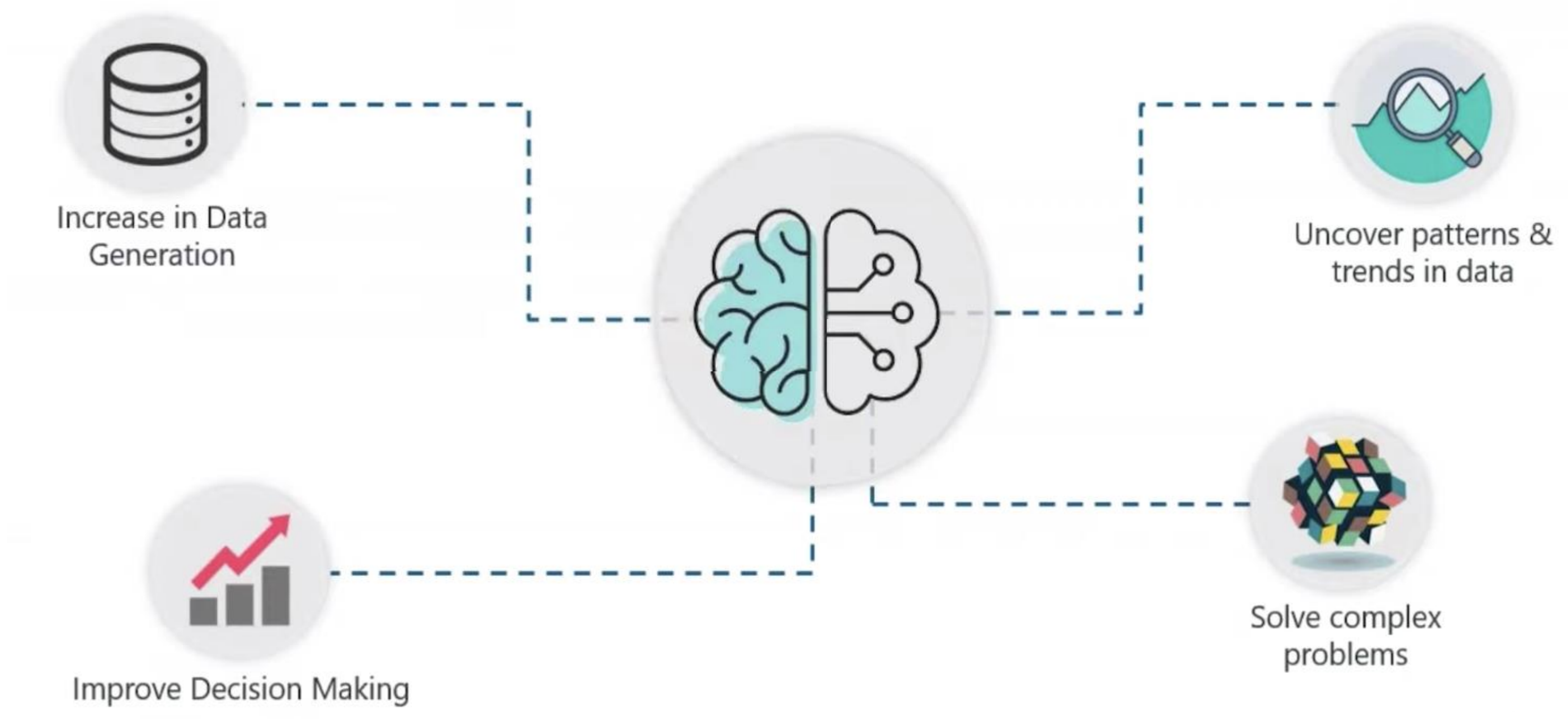


https://medium.com/machine-learning-for-humans/neural-networks-deep-learning-cdad8aeae49b

Why Machine Learning?

Over 2.5 quintillion bytes of data are created every single day, and it is only going to grow from there. By 2020, it is estimated that 1.7MB of data will be created every second for every person on earth.

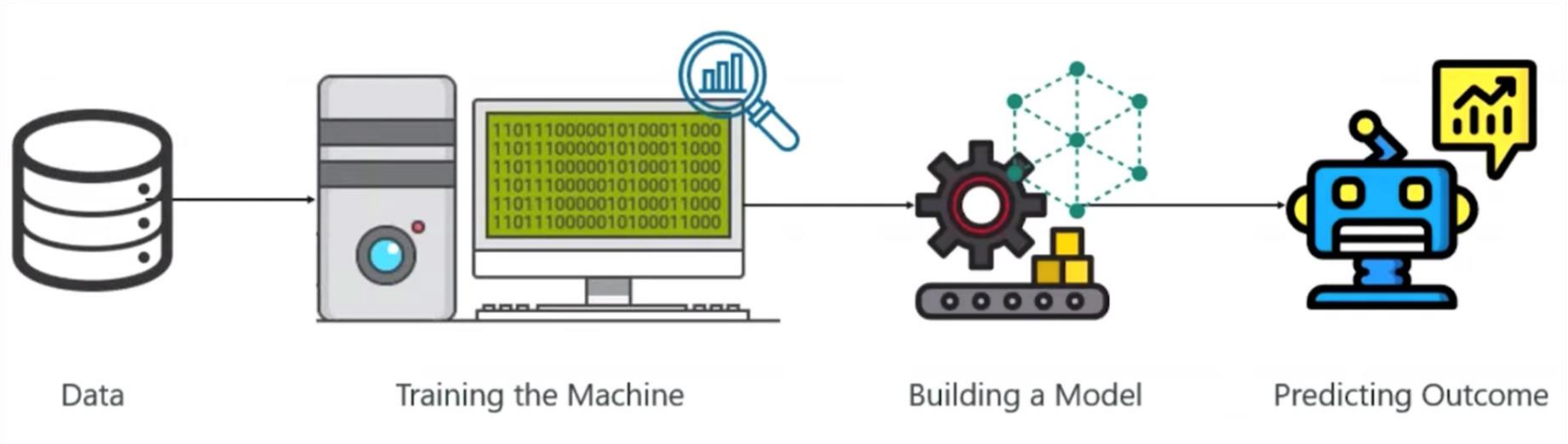
Why Machine Learning?



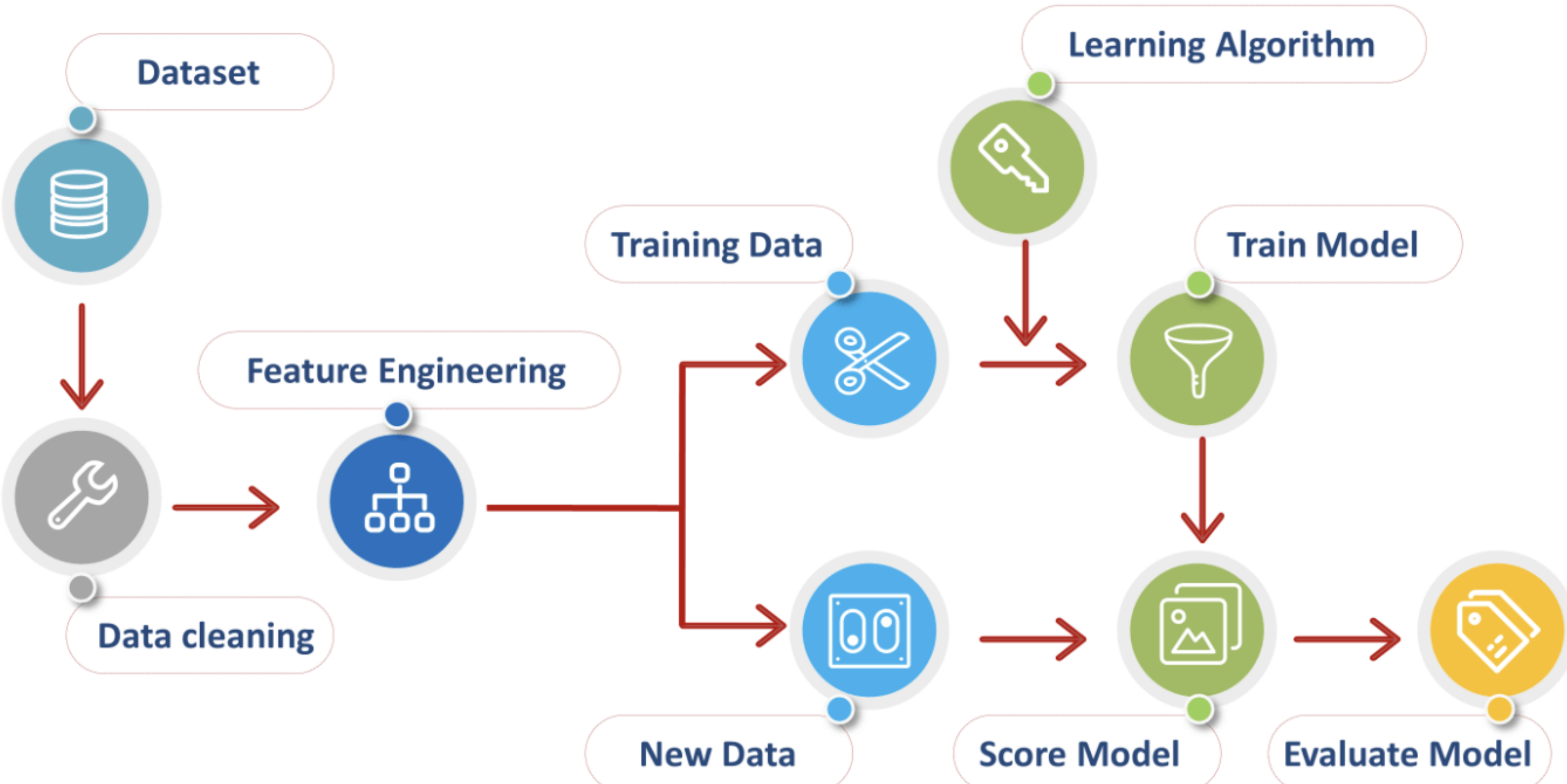
Machine Learning History

In 1959, Arthur Samuel first proposed the concept Machine Learning:

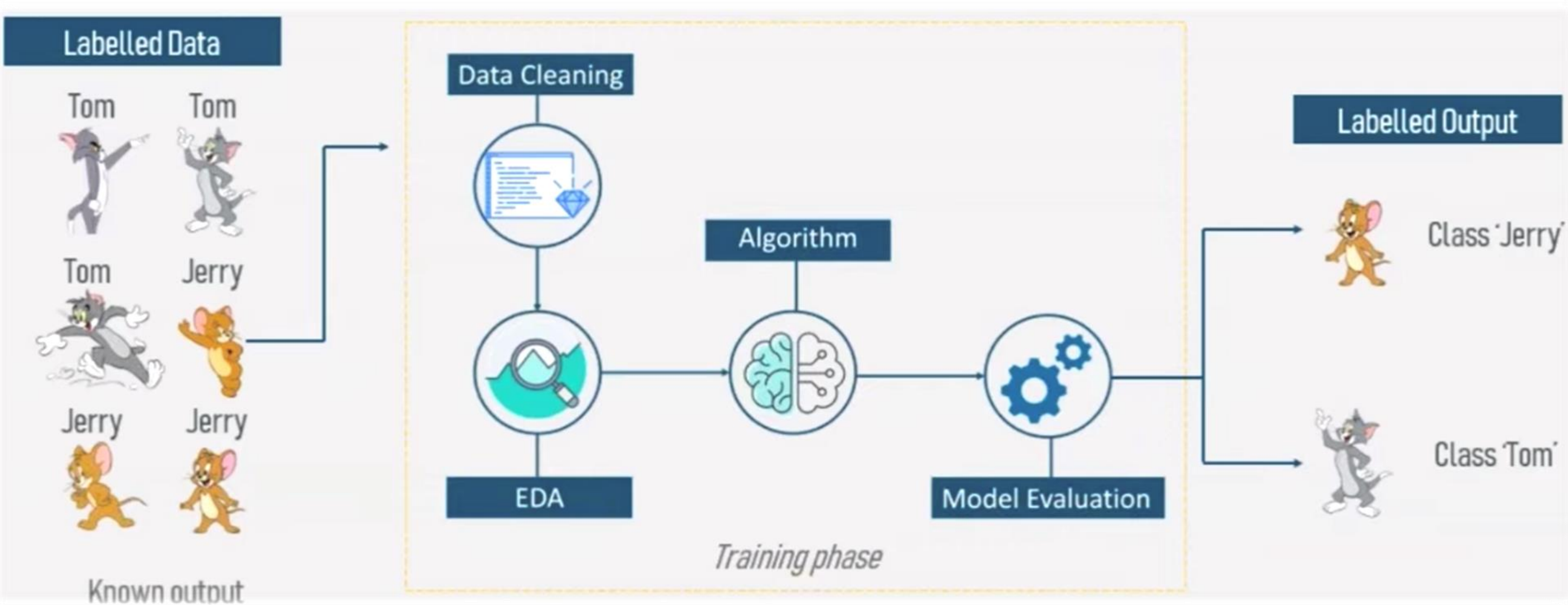
"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."



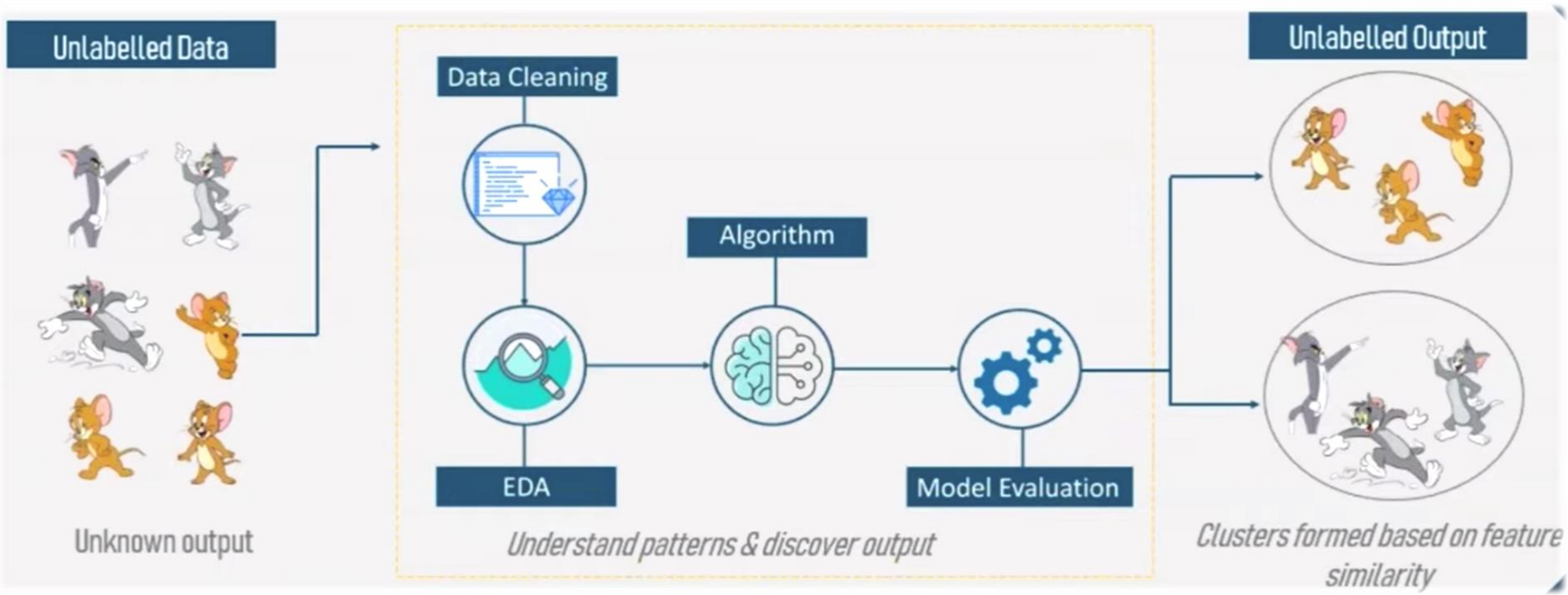
Machine Learning Process



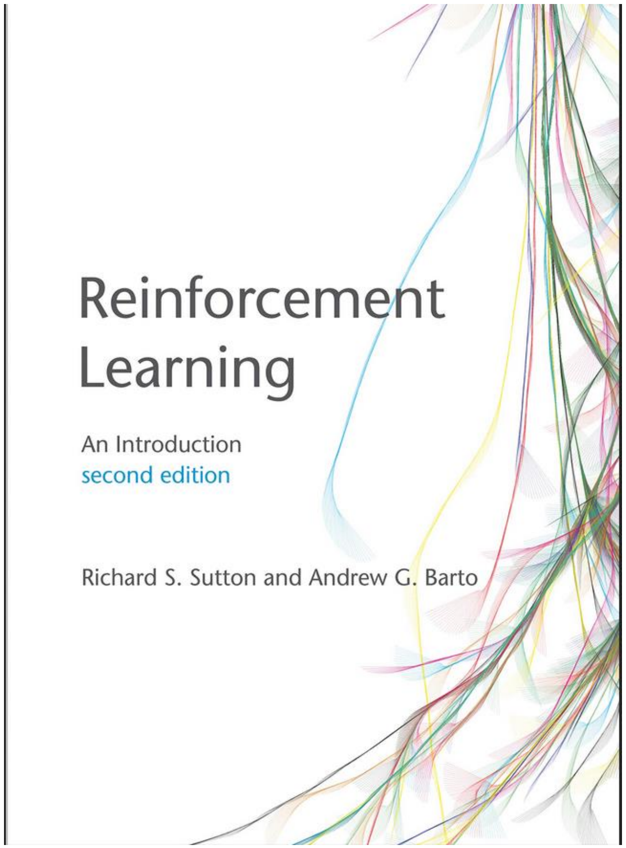
Supervised Learning

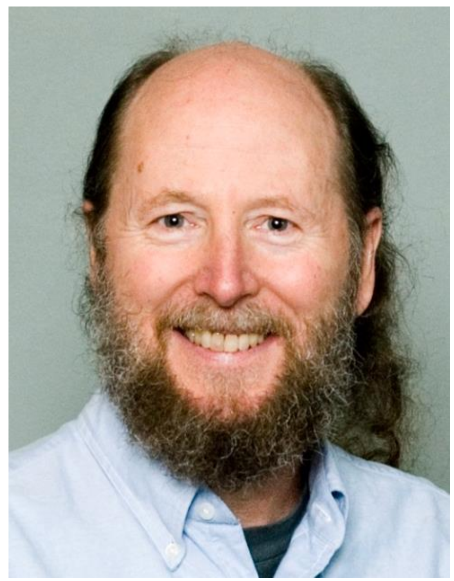


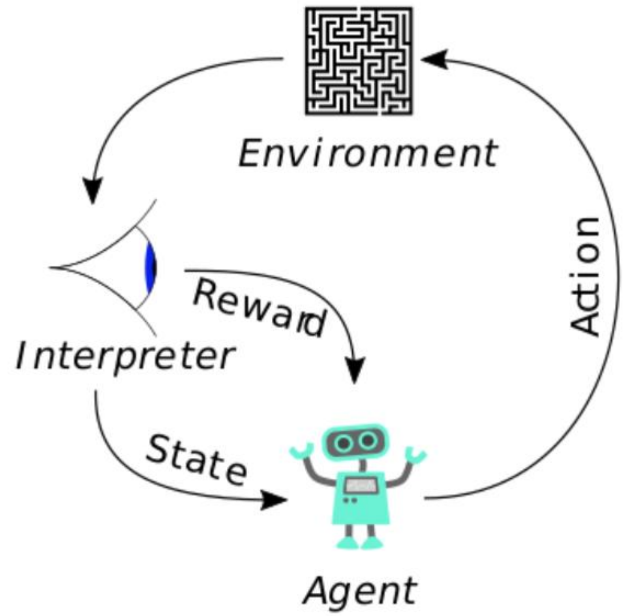
Unsupervised Learning



Reinforcement Learning

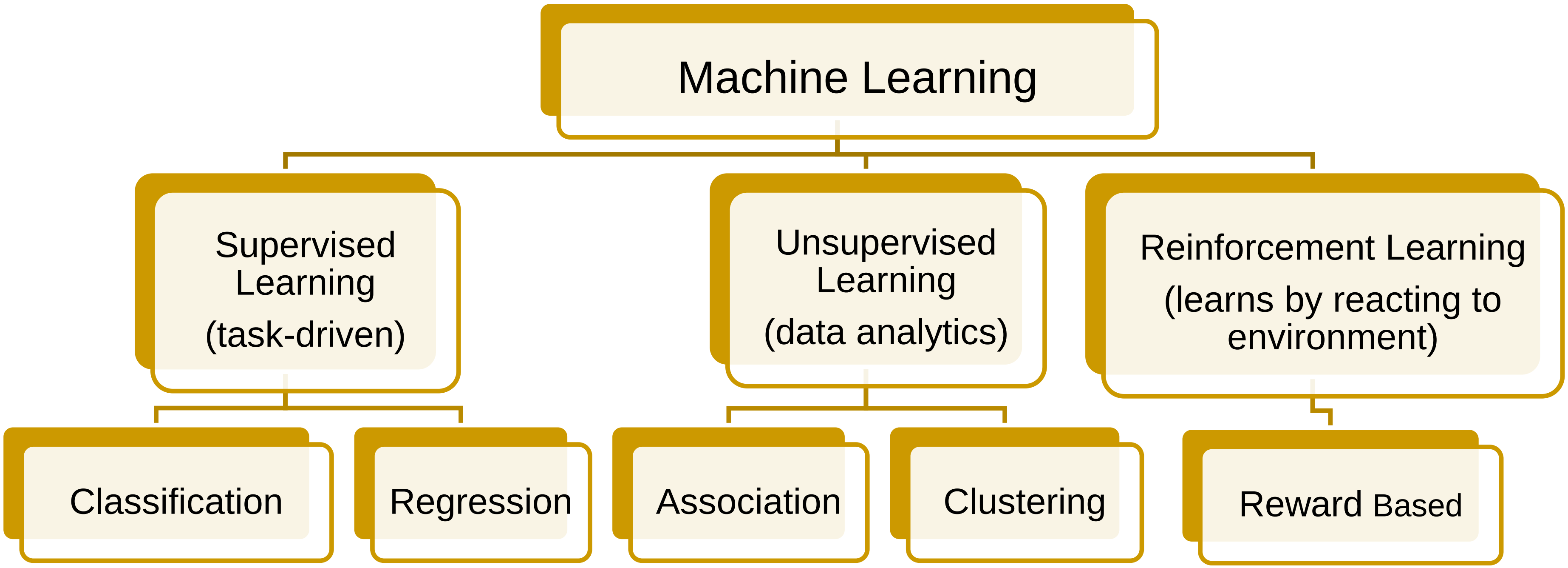






The typical RL scenario: an agent takes actions in an environment, which is interpreted into a reward and a representation of the state, which are fed back into the agent.

Types of ML Algorithms



ML outside of AI

ML is widely used in Data Mining

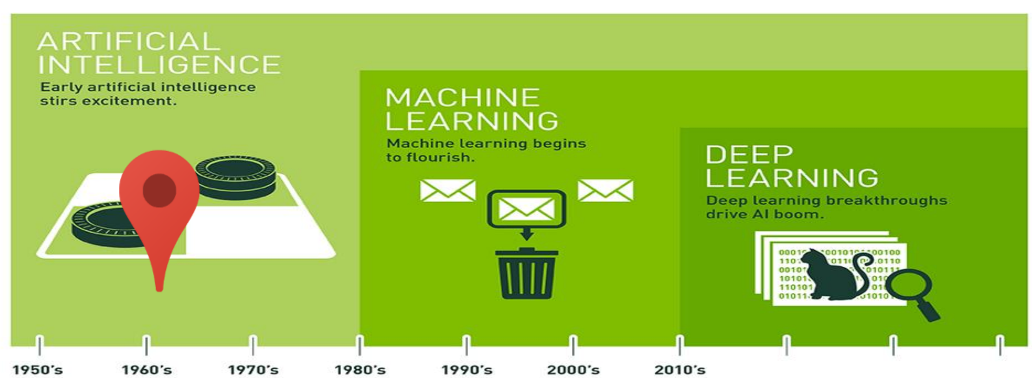
- a.k.a. Knowledge Discovery in Databases (KDD)
- e.g. Clustering, Anomaly Detection, Association Rule Mining
- Example: predict if a customer is likely to purchase certain goods according to history of shopping activities.

Types of Machine Learning

	Supervised Learning	Unsupervised Learning	Reinforcement Learning
Definition	The machine learns by using labelled data	The machine is trained on unlabeled data without any guidance	An agent interacts with its environment by producing actions & discovers errors and rewards
Types of problems	Regression &Classification	Association & Clustering	Reward based
Type of data	Labelled data	Unlabelled data	No pre-defined data
Training	External supervision	No supervision	No supervision
Approach	Map labelled input to known output	Understand patterns and discover output	Follow trail and error method
Popular Algorithms	Linear Regression, Logistic Regression, KNN, etc	K-means, C-means, etc	Q-learning, etc

Today

1. Introduction to ML
2. Naïve Bayes Classifier
3. Evaluation



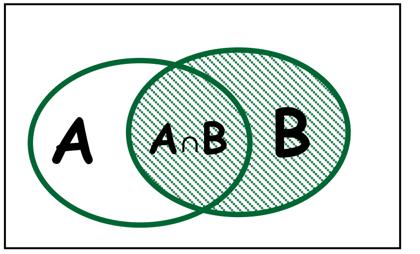
https://medium.com/machine-learning-for-humans/neural-networks-deep-learning-cdad8aeae49b

Motivation

- How do we represent and reason about non-factual knowledge?
  - It *might* rain tonight
  - If you have red spots on your face, you *might* have the measles
  - This e-mail is *most likely* spam
  - I can't read this character, but it *looks* like a "B"
  - These 2 pictures are *very likely* of the same person
  - …

Remember…

- P is a probability function:
  - $0 \leq P(A) \leq 1$
  - $P(A) = 0 \Rightarrow$ the event A will never take place
  - $P(A) = 1 \Rightarrow$ the event A must take place
  - $\sum_i P(A_i) = 1 \Rightarrow$ one of the events A_i will take place
  - $P(A) + P(\sim A) = 1$



- Joint probability
  - intersection $A_1 \cap \ldots \cap A_n$ is an event that takes place if all the events $A_1, \ldots, A_n$ take place
  - denoted $P(A \cap B)$ or $P(A,B)$
- Sum Rule
  - union $A_1 \cup \ldots \cup A_n$ is an event that takes place if at least one of the events $A_1, \ldots, A_n$ takes place
  - denoted $P(A \cup B) = P(A) + P(B) - P(A \cap B)$

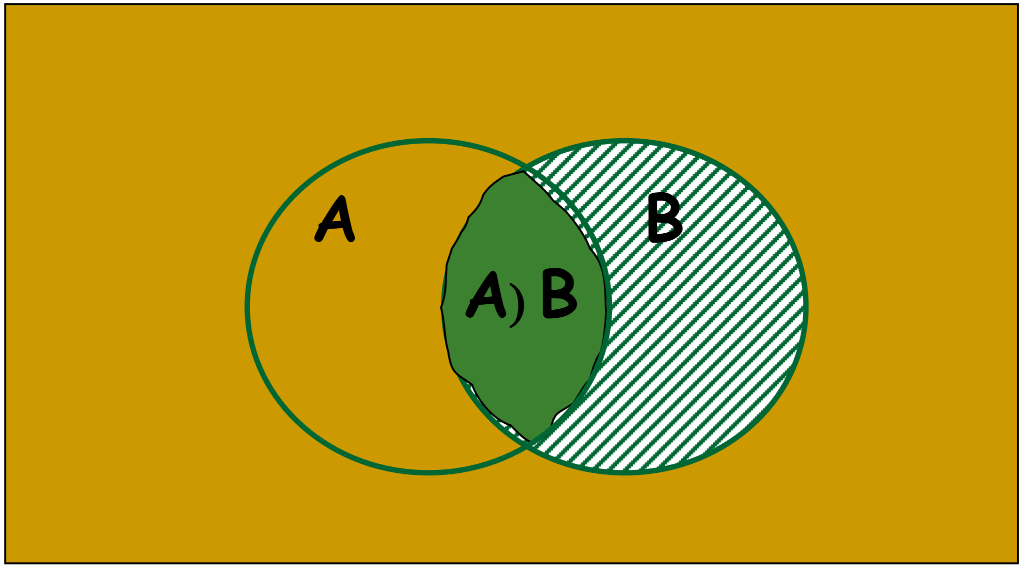
Conditional Probability

- Prior (or unconditional) probability
  - Probability of an event before any evidence is obtained
  - P(A) = 0.1 *P(rain today) = 0.1*
  - i.e. your belief about *A* given that you have no evidence

- Posterior (or conditional) probability
  - Probability of an event given that you know that B is true (B = some evidence)
  - P(A|B) = 0.8 *P(rain today| cloudy) = 0.8*
  - i.e. your belief about *A* given that you know B

Conditional Probability (con't)

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A,B)}{P(B)}$$



Example

- Rolling two dice (together):

2	3	4	5	6	7	8	9	10	11	12
1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36

- Rolling two dice one after the other, first dice rolled 1:

2	3	4	5	6	7	8	9	10	11	12
1/6	1/6	1/6	1/6	1/6	1/6	0	0	0	0	0

Chain Rule

- With 2 events, the probability that A and B occur is:

$$P(A|B)=\frac{P(A,B)}{P(B)} \text{ so } P(A,B)=P(A|B)\times P(B)$$

- With 3 events, the probability that A, B and C occur is:
  - The probability that A occurs
  - Times, the probability that B occurs, assuming that A occurred
  - Times, the probability that C occurs, assuming that A and B have occurred

- With n events, we can generalize to the Chain rule:

$P(A_1, A_2, A_3, A_4, \ldots, A_n)$

$= P(\cap A_i)$

$= P(A_1) \times P(A_2|A_1) \times P(A_3|A_1,A_2) \times \ldots \times P(A_n|A_1,A_2,A_3,\ldots,A_{n-1})$

So what?

- we can do probabilistic inference
  - i.e. infer new knowledge from observed evidence

Example 1

- Joint probability distribution:

$\boxed{P(Toothache \cap Cavity)}$

evidence

hypothesis

	Toothache	~Toothache
Cavity	0.04	0.06
~Cavity	0.01	0.89

$$P(H \mid E) = \frac{P(H \cap E)}{P(E)}$$

$$P(cavity \mid toothache) = \frac{P(cavity \cap toothache)}{P(toothache)} = \frac{0.04}{0.04 + 0.01} = 0.8$$

Getting the Probabilities

- in most applications, you just count from a set of observations

$$P(A) = \frac{count\_of\_A}{count\_of\_all\_events}$$

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} = \frac{count\_of\_A\_and\_B\_together}{count\_of\_all\_B}$$

Combining Evidence

- Assume now 2 pieces of evidence:

- Suppose, we know that
  - P(Cavity | Toothache) = 0.12
  - P(Cavity | Young) = 0.18

- A patient complains about Toothache and is Young…
  - what is P(Cavity | Toothache $\cap$ Young) ?

Combining Evidence

	Toothache		~Toothache	
	Young	~ Young	Young	~ Young
Cavity	0.108	0.012	0.072	0.008
~Cavity	0.016	0.064	0.144	0.576

P(Toothache ∩ Cavity ∩ Young)

- But how do we get the data ?
- In reality, we may have dozens, hundreds of variables
- We cannot have a table with the probability of all possible combinations of variables
  - Ex. with 16 binary variables, we would need 2^{16} entries

Independent Events

- In real life:
  - some variables are independent…
    - ex: living in Montreal & tossing a coin
    - P(Montreal, head) = P(Montreal) * P(head)
    - probability of 2 heads in a row:
      - P(head, head) = 1/2 * 1/2 = 1/4
  - some variables are not independent…
    - ex: living in Montreal & wearing boots
    - P(Montreal, boots) ≠ P(Montreal) * P(boots)

Independent Events

- Two events A and B are independent:
  - if the occurrence of one of them does not influence the occurrence of the other
  - i.e. A is independent of B if $P(A) = P(A|B)$
- If A and B are independent, then:
  - $P(A,B) = P(A|B) \times P(B)$ (by chain rule)
    $= P(A) \times P(B)$ (by independence)
- To make things work in real applications, we often assume that events are independent
  - $P(A,B) = P(A) \times P(B)$

Conditional Independent Events

- Two events A and B are conditionally independent given C:
  - Given that C is true, then any evidence about B cannot change our belief about A
  - $P(A, B \mid C) = P(A \mid C) \times P(B \mid C)$.

Bayes' Theorem

- given: $P(A|B) = \frac{P(A,B)}{P(B)}$ so $P(A,B) = P(A|B) \times P(B)$

  $P(B|A) = \frac{P(A,B)}{P(A)}$ so $P(A,B) = P(B|A) \times P(A)$

- then: $P(A|B) \times P(B) = P(B|A) \times P(A)$

- and: $P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$

So?

- We typically want to know: P(Hypothesis | Evidence)
  - P(Disease | Symptoms)… P(meningitis | red spots)
  - P(Cause | Side Effect)… P(misaligned brakes| squeaky wheels)
- But P(Hypothesis| Evidence) is hard to gather
  - *ex: out of all people who have red spots… how many have meningitis?*
- However P(Evidence | Hypothesis) is easier to gather
  - *ex: out of all people who have the meningitis … how many have red spots?*
- So

$$P(\text{Hypothesis} \mid \text{Evidence}) = \frac{P(\text{Evidence} \mid \text{Hypothesis}) \times P(\text{Hypothesis})}{P(\text{Evidence})}$$

Example 2

Assume we only have 1 hypothesis

Assume:

- P(spots=yes | meningitis=yes) = 0.4
- P(meningitis=yes) = 0.00003
- P(spots=yes) = 0.05

$$P(\text{meningitis} = \text{yes} \mid \text{spots} = \text{yes})$$
$$= \frac{P(\text{spots} = \text{yes} \mid \text{meningitis} = \text{yes}) \times P(\text{meningitis} = \text{yes})}{P(\text{spots} = \text{yes})}$$
$$= \frac{0.4 \times 0.00003}{0.05} = 0.00024$$

→ If you have spots… you are more likely to have meningitis than if we don't know about you having spots

Example 3

- Predict the weather tomorrow based on tonight's sunset...
- Assume we have 3 hypothesis...
  - H_1: *weather will be nice* $P(H_1) = 0.2$
  - H_2: *weather will be bad* $P(H_2) = 0.5$
  - H_3: *weather will be mixed* $P(H_3) = 0.3$
- And 1 piece of evidence with 3 possible values
  - E_1: today, there's a *beautiful sunset*
  - E_2: today, there's a *average sunset*
  - E_3: today, there's *no sunset*

$P(E_2 | H_1)$

$P(E_x \| H_i)$	E_1	E_2	E_3
H_1	0.7	0.2	0.1
H_2	0.3	0.3	0.4
H_3	0.4	0.4	0.2

Example 3

- Predict the weather tomorrow based on tonight's sunset...
- Assume we have 3 hypothesis...
  - H_1: *weather will be nice* $P(H_1) = 0.2$
  - H_2: *weather will be bad* $P(H_2) = 0.5$
  - H_3: *weather will be mixed* $P(H_3) = 0.3$
- And 1 piece of evidence with 3 possible values
  - E_1: today, there's a *beautiful sunset*
  - E_2: today, there's a *average sunset*
  - E_3: today, there's *no sunset*

$P(E_x \mid H_i)$	E_1	E_2	E_3
H_1	0.7	0.2	0.1
H_2	0.3	0.3	0.4
H_3	0.4	0.4	0.2

Example 3

- Observation: average sunset (E_2)
- Question: how will be the weather tomorrow?
  - $P(H_i \mid E_2)$?
  - predict the weather that maximizes the probability
  - select H_i such that $P(H_i \mid E_2)$ is the greatest

$$P(H_i \mid E_2) = \frac{P(H_i) \times P(E_2 \mid H_i)}{P(E_2)}$$

$$P(E_2) = P(H_1) \times P(E_2 \mid H_1) + P(H_2) \times P(E_2 \mid H_2) + P(H_3) \times P(E_2 \mid H_3)$$
$$= .2x.2 + .5x.3 + .3x.4 = .04 + .15 + .12 = 0.31$$

Example 3

$$P(H_1 | E_2) = \frac{P(H_1) \times P(E_2 | H_1)}{P(E_2)} = \frac{.2x.2}{.31} = .129$$

$$P(H_2 | E_2) = \frac{P(H_2) \times P(E_2 | H_2)}{P(E_2)} = \frac{.5x.3}{.31} = .484$$

$$P(H_3 | E_2) = \frac{P(H_3) \times P(E_2 | H_3)}{P(E_2)} = \frac{.3x.4}{.31} = .387$$

$\Rightarrow$ H_2 is the most likely hypothesis, given the evidence

$P(H_2 \mid E_2)$ is the highest

Tomorrow the weather will be bad

$$H_{NB} = \underset{H_i}{\operatorname{argmax}} \frac{P(H_i) \times P(E | H_i)}{P(E)}$$

Bayes' Reasoning

- Out of n hypothesis…
  - we want to find the most probable H_i given the evidence E
- So we choose the H_i with the largest $P(H_i|E)$

$$H_{NB} = \operatorname*{argmax}_{H_i} P(H_i \mid E) = \operatorname*{argmax}_{H_i} \frac{P(H_i) \times P(E \mid H_i)}{P(E)}$$

- But… P(E)
  - is the same for all possible H_i (and is hard to gather anyways)
  - so we can drop it
- So Bayesian reasoning:

$$H_{NB} = \operatorname*{argmax}_{H_i} \frac{P(H_i) \times P(E \mid H_i)}{P(E)} = \operatorname*{argmax}_{H_i} P(H_i) \times P(E \mid H_i)$$

Representing the Evidence

- The evidence is typically represented by many attributes/features
  - beautiful sunset? clouds? temperature? summer?, …
- so often represented as a feature/attribute vector

	evidence					hypothesis
	sunset a_1	clouds a_2	temp a_3.	summer a_4		weather tomorrow
e1	beautiful	no	high	yes		*Nice*

- e1 = <a_1, … , a_n>
- e1 = <sunset:beautiful, clouds:no, temp:high, summer:yes>

Combining Evidence

toothache	young	cavity
yes	yes	?

$$P(Cavity = yes \mid Toothache = yes \cap Young = yes) = ?$$

with Bayes Rule :

$$= \frac{P(Toothache = yes \cap Young = yes \mid Cavity = yes) x P(Cavity = yes)}{P(Toothache = yes \cap Young = yes)}$$

with independence assumption :

$$= \frac{P(Toothache = yes \cap Young = yes \mid Cavity = yes) x P(Cavity = yes)}{P(Toothache = yes\ ) \times P(Young = yes\ )}$$

with conditional independence assumption :

$$= \frac{P(Toothache = yes \mid Cavity = yes\ )\ \ x P(Young = yes \mid Cavity = yes) x\ P(Cavity = yes)}{P(Toothache = yes\ ) \times P(Young = yes)}$$

Now we have decomposed the joint probability distribution into much smaller pieces…

Combining Evidence

toothache	young	cavity
yes	yes	*yes? or no?*

But since we only care about <u>ranking</u> the hypothesis…

$$P(\text{Cavity}=\text{yes} \mid \text{Toothache}=\text{yes} \cap \text{Young}=\text{yes})$$
$$>$$
$$P(\text{Cavity}=\text{no} \mid \text{Toothache}=\text{yes} \cap \text{Young}=\text{yes})$$

$$\frac{P(\text{Cavity}=\text{yes}) \times P(\text{Toothache}=\text{yes} \mid \text{Cavity}=\text{yes}) \times P(\text{Young}=\text{yes} \mid \text{Cavity}=\text{yes})}{P(\text{Toothache}=\text{yes}) \times P(\text{Young}=\text{yes})}$$
$$>$$
$$\frac{P(\text{Cavity}=\text{no}) \times P(\text{Toothache}=\text{yes} \mid \text{Cavity}=\text{no}) \times P(\text{Young}=\text{yes} \mid \text{Cavity}=\text{no})}{P(\text{Toothache}=\text{yes}) \times P(\text{Young}=\text{yes})}$$

$$P(\text{Cavity}=\text{yes}) \times P(\text{Toothache}=\text{yes} \mid \text{Cavity}=\text{yes}) \times P(\text{Young}=\text{yes} \mid \text{Cavity}=\text{yes})$$
$$>$$
$$P(\text{Cavity}=\text{no}) \times P(\text{Toothache}=\text{yes} \mid \text{Cavity}=\text{no}) \times P(\text{Young}=\text{yes} \mid \text{Cavity}=\text{no})$$

$$H_{NB} = \underset{H_i}{\text{argmax}} \frac{P(H_i) \times P(E \mid H_i)}{P(E)} = \underset{H_i}{\text{argmax}}\ P(H_i) \times P(E \mid H_i) = \underset{H_i}{\text{argmax}}\ P(H_i) \times P(< a_1, a_2, a_3, \ldots, a_n > \mid H_i) = \underset{H_i}{\text{argmax}}\ P(H_i) \times \prod_{j=1}^{n} P(a_j \mid H_i)$$

Example 4

evidence

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
Day1	Sunny	Hot	High	Weak	No
Day2	Sunny	Hot	High	Strong	No
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day6	Rain	Cool	Normal	Strong	No
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	No
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day12	Overcast	Mild	High	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes
Day14	Rain	Mild	High	Strong	No

Example 4

- Goal: Given a new instance $X=\langle a_1,\ldots, a_n\rangle$, classify as Yes/No

$$H_{NB} = \underset{H_i}{\operatorname{argmax}} \frac{P(H_i) \times P(E|H_i)}{P(E)} = \underset{H_i}{\operatorname{argmax}} P(H_i) \times P(E|H_i) = \underset{H_i}{\operatorname{argmax}} P(H_i) \times P(< a_1, a_2, a_3, \ldots, a_n >| H_i) = \underset{H_i}{\operatorname{argmax}} P(H_i) \times \prod_{j=1}^{n} P(a_j | H_i)$$

- Naïve Bayes: Assumes that the attributes/features are conditionally independent

Example 4

- Goal: Given a new instance X=<a_1,…, a_n>, classify as Yes/No

$$H_{NB} = \underset{H_i}{\operatorname{argmax}}\ P(H_i) \times \prod_{j=1}^{n} P(a_j \mid H_i)$$

1st estimate the probabilities from the training examples:

a) For each hypothesis H_i

estimate $P(H_i)$

b) For each attribute value a_j of each instance (evidence)

estimate $P(a_j \mid H_i)$

Example 4

1. TRAIN:
   - compute the probabilities from the training set

$$\left.\begin{aligned} &P(PlayTennis = yes) = 9/14 = 0.64 \\ &P(PlayTennis = no) = 5/14 = 0.36 \end{aligned}\right\}$$ **prior probabilities P(H_i)**

$$\left.\begin{aligned} &P(Out = sunny \mid PlayTennis = yes) = 2/9 = 0.22 \\ &P(Out = sunny \mid PlayTennis = no) = 3/5 = 0.60 \\ &P(Out = rain \mid PlayTennis = yes) = 3/9 = 0.33 \\ &P(Out = rain \mid PlayTennis = no) = 2/5 = 0.4 \\ &\ldots \\ &P(Wind = strong \mid PlayTennis = yes) = 3/9 = 0.33 \\ &P(Wind = strong \mid PlayTennis = no) = 3/5 = 0.60 \end{aligned}\right\}$$ **conditional probabilities** $P(a_j \mid H_i)$

Example 4

2. TEST:

classify the new case: X=(Outlook: Sunny, Temp: Cool, Hum: High, Wind: Strong)

$$H_{NB} = \underset{H_i \in [yes,no]}{\operatorname{argmax}} \; P(H_i) \times P(X \mid H_i)$$

$$= \underset{H_i \in [yes,no]}{\operatorname{argmax}} \; P(H_i) \times \prod_j P(a_j \mid H_i)$$

$$= \underset{H_i \in [yes,no]}{\operatorname{argmax}} P(H_i) \times P(Outlook = sunny \mid H_i) \times P(Temp = cool \mid H_i)$$

$$\times P(Humidity = high \mid H_i) \times P(Wind = strong \mid H_i)$$

1) $P(PlayTennis = yes)$
$\times P(Outlook = sunny \mid PlayTennis = yes) x P(Temp = cool \mid PlayTennis = yes) x P(Hum = high \mid PlayTennis = yes) x P(Wind = strong \mid PlayTennis = yes)$
$= 0.0053$

2) $P(PlayTennis = no)$
$\times P(Outlook = sunny \mid PlayTennis = no) x P(Temp = cool \mid PlayTennis = no) x P(Hum = high \mid PlayTennis = no) x P(Wind = strong \mid PlayTennis = no)$
$= 0.0206$

$\Rightarrow$ answer : $PlayTennis(X) = no$

Application of Bayesian Reasoning

- Categorization: P(Category | Features of Object)
  - Diagnostic systems: P(Disease | Symptoms)
  - Text classification: P(sports_news | text)
  - Character recognition: P(character | bitmap)
  - Speech recognition: P(words | acoustic signal)
  - Image processing: P(face_person | image features)
  - Spam filter: P(spam_message | words in e-mail)
  - ...

Naive Bayes Classifier

- A simple probabilistic classifier based on Bayes' theorem
  - with strong (naive) independence assumption
  - i.e. the features/attributes are conditionally independent

- The assumption of conditional independence, often does not hold…
- But Naïve Bayes works very well in many applications anyways!
  - ex: Medical Diagnosis
  - ex: Text Categorization (spam filtering)

Ex. Application: Spam Filtering

- Task: classify e-mails (documents) into a pre-defined class
  - ex: spam / ham
  - ex: sports, recreation, politics, war, economy,…
  - ex: customer email → order, complaint, support request, …
- Given
  - N sets of training texts (1 set for each class)
  - Each set is already tagged by the class name

Strictly speaking, what we will see is called a Multinomial Naïve Bayes classifier, because we will count the number of words, as opposed to just using binary values for the presence/absence of words…

e-mail Representation

- each e-mail is represented by a vector of feature/value:
  - feature = actual words in the e-mail
  - value = number of times that word appears in the e-mail
- each e-mail in the training set is tagged with the correct category.

data instance	features / evidence / X						f(X)
	offer	money	viagra	laptop	exam	study	category
email 1	3	2	5	1	0	1	SPAM
email 2	1	1	0	5	4	3	HAM
email 3	0	3	2	1	0	1	SPAM
...							

- task: correctly tag a new e-mail

	offer	money	viagra	laptop	exam	study	category
new email	2	1	0	1	1	2	?

Naïve Bayes Algorithm

// 1. training

```
for all classes c_i    // ex. ham or spam
    for all words w_j in the vocabulary
        compute
```

$$P(w_j \mid c_i) = \frac{count(w_j, c_i)}{\sum_j count(w_j, c_i)}$$

```
for all classes c_i
    compute
```

$$P(c_i) = \frac{count(documents\ in\ c_i)}{count(all\ documents)}$$

// 2. testing a new document D

```
for all classes c_i  // ex. ham or spam
    score(c_i) = P(c_i)
    for all words w_j in the D
        score(c_i) = score(c_i) x P(w_j | c_i)

choose c* = with the greatest score(c_i)
```

	w_1	w_2	w_3	w_4	w_5	w_6
c1 : SPAM	$p(w_1 \mid c_1)$	$p(w_2 \mid c_1)$	$p(w_3 \mid c_1)$	$p(w_4 \mid c_1)$	$p(w_5 \mid c_1)$	$p(w_6 \mid c_1)$
c2 : HAM	$p(w_1 \mid c_2)$	$p(w_2 \mid c_2)$	$p(w_3 \mid c_2)$	$p(w_4 \mid c_2)$	$p(w_5 \mid c_2)$	$p(w_6 \mid c_2)$

Example

- Dataset
  - c1: SPAM
    doc1: "cheap meds for sale"
    doc2: "click here for the best meds"
    doc3: "book your trip"
  - c2: HAM
    doc4: "cheap book sale, not meds"
    doc5: "here is the book for you"
- Question:
  - doc6: “the cheap book”
  - should it be classified as HAM or SPAM?

Spam

Ham

?

Example (cont'd)

Assume

vocabulary = {best, book, cheap, sale, trip, meds}

If not in vocabulary, ignore word

1. Training:
   - P(best|SPAM) = 1/7
   - P(book|SPAM) = 1/7
   - P(cheap|SPAM) = 1/7
   - P(sale|SPAM) = 1/7
   - P(trip|SPAM) = 1/7
   - P(meds|SPAM) = 2/7
   - P(best|HAM) = 0/5
   - P(book|HAM) = 2/5
   - P(cheap|HAM) = 1/5
   - P(sale|HAM) = 1/5
   - P(trip|HAM) = 0/5
   - P(meds|HAM) = 1/5
   - P(SPAM) = 3/5
   - P(HAM) = 2/5

2. Testing: "the cheap book"
   - Score(HAM)= P(HAM) x P(cheap|HAM) x P(book|HAM)
   - Score(SPAM)= P(SPAM) x P(cheap|SPAM) x P(book|SPAM)

Be Careful: Smooth Probabilities

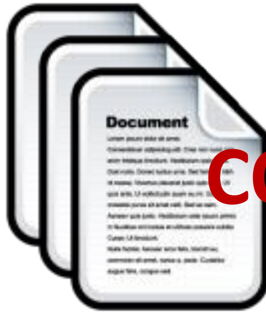
- normally: $P(w_i \mid c_j) = \frac{(\text{frequency of } w_i \text{ in } c_j)}{\text{total number of words in } c_j}$
- what if we have a $P(w_i|c_j) = 0$...?
  - ex. the word "dumbo" never appeared in the class SPAM?
  - then P("dumbo"| SPAM) = 0
- so if a text contains the word "dumbo", the class SPAM is completely ruled out !
- to solve this: we assume that every word always appears at least once (or a smaller value)
  - ex: add-1 smoothing:

$$P(w_i \mid c_j) = \frac{(\text{frequency of } w_i \text{ in } c_j) + 1}{\text{total number of words in } c_j + \text{size of vocabulary}}$$

Be Careful: Use Logs

- if we really do the product of probabilities…
  - $\text{argmax}_{cj}\ P(c_j) \prod P(w_i|c_j)$
  - we soon have numerical underflow…
  - ex: 0.01 x 0.02 x 0.05 x …
- so instead, we add the log of the probs
  - $\text{argmax}_{cj}\ \log(P(c_j)) + \sum \log(P(w_i|c))$
  - ex: log(0.01) + log(0.02) + log(0.05) + …

Example



- Dataset

c1: COOKING	c2: SPORTS
doc_1: … stove… kitchen… the… heat doc_2: … kitchen… pasta… stove… … doc_{100000}: … stove…heat… ball…	doc_1: … ball… heat… doc_2: … the… referee… player… … doc_{75000}: goal… injury …

- Assume:
  - $|V| = 100$ vocabulary = {ball, heat, kitchen, referee, stove, the, … }
  - 500,000 words in Cooking
  - 300,000 words in Sports
  - 100,000 docs in Cooking
  - 75,000 docs in Sports

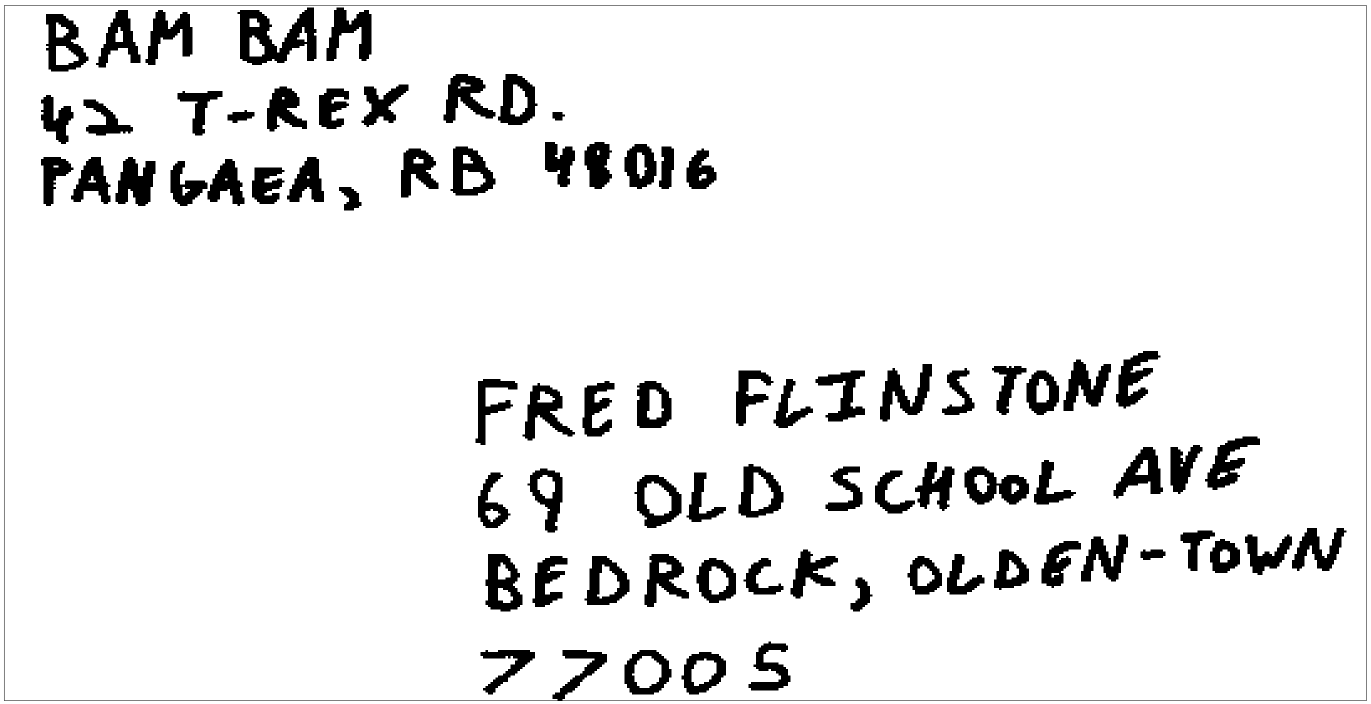
Example

- Training – Unsmoothed / Smoothed probs:
  - P(ball|COOKING) = P(ball|SPORTS) =
  - P(heat|COOKING) = P(heat|SPORTS) =
  - P(kitchen|COOKING) = P(kitchen|SPORTS) =
  - P(referee|COOKING) = P(referee|SPORTS) =
  - P(stove|COOKING) = P(stove|SPORTS) =
  - P(the|COOKING) = P(the|SPORTS) =
  - …
  - P(COOKING) = P(SPORTS) =
- Testing: "the referee hit the ~~blue~~ ~~bird~~"
  - Score(COOKING)= log() + log(P(the|COOKING)) + log(P(referee|COOKING)) + log(P(hit|COOKING)) + log(P(the|COOKING))
  - Score(SPORTS)= log() + log(P(the|SPORTS)) + log(P(referee|SPORTS)) + log(P(hit|SPORTS)) + log(P(the| SPORTS))

Example

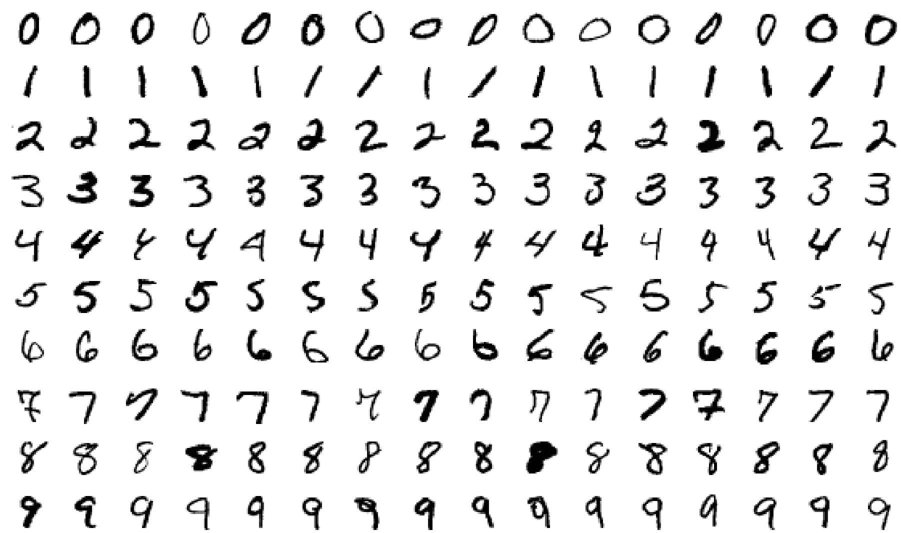
- Training – Unsmoothed / Smoothed probs:
  - P(ball|COOKING) = $\frac{10,000}{500,000}$ $\frac{??}{??}$ P(ball|SPORTS) = $\frac{10,000}{300,000}$ $\frac{??}{??}$
  - P(heat|COOKING) = $\frac{255}{500,000}$ $\frac{??}{??}$ P(heat|SPORTS) = $\frac{1,8000}{300,000}$ $\frac{??}{??}$
  - P(kitchen|COOKING) = $\frac{2,600}{500,000}$ $\frac{??}{??}$ P(kitchen|SPORTS) = $\frac{0}{300,000}$ $\frac{??}{??}$
  - P(referee|COOKING) = $\frac{0}{500,000}$ $\frac{??}{??}$ P(referee|SPORTS) = $\frac{1,500}{300,000}$ $\frac{??}{??}$
  - P(stove|COOKING) = $\frac{3,600}{500,000}$ $\frac{??}{??}$ P(stove|SPORTS) = $\frac{4}{300,000}$ $\frac{??}{??}$
  - P(the|COOKING) = $\frac{400,000}{500,000}$ $\frac{??}{??}$ P(the|SPORTS) = $\frac{19,000}{300,000}$ $\frac{??}{??}$
  - …
  - P(COOKING) = $\frac{100,000}{175,000}$ P(SPORTS) = $\frac{75,000}{175,000}$
- Testing: "the referee hit the ~~blue~~ ~~bird~~"
  - Score(COOKING)= log($\frac{100,000}{175,000}$) + log(P(the|COOKING)) + log(P(referee|COOKING)) + log(P(hit|COOKING)) + log(P(the|COOKING))
  - Score(SPORTS)= log($\frac{75,000}{175,000}$) + log(P(the|SPORTS)) + log(P(referee|SPORTS)) + log(P(hit|SPORTS)) + log(P(the| SPORTS))

Another Application: Postal Code Recognition



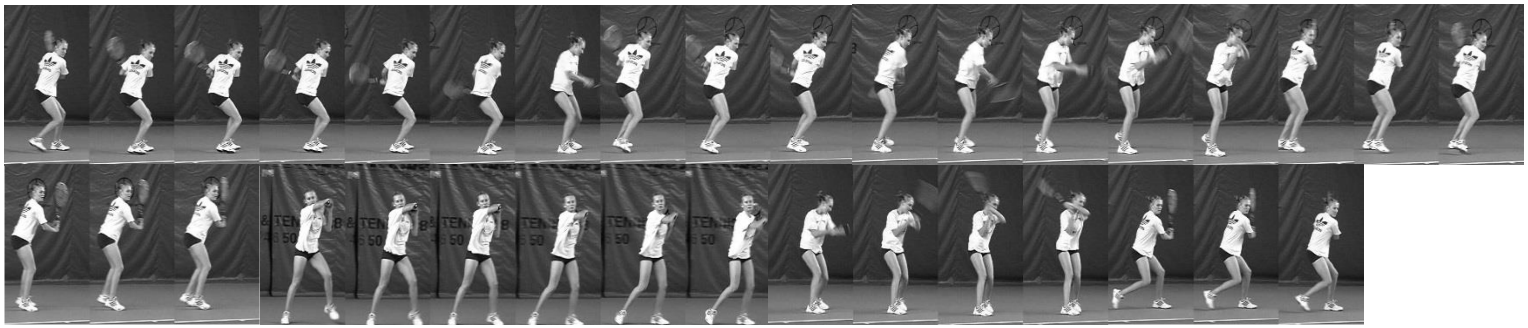
Digit Recognition

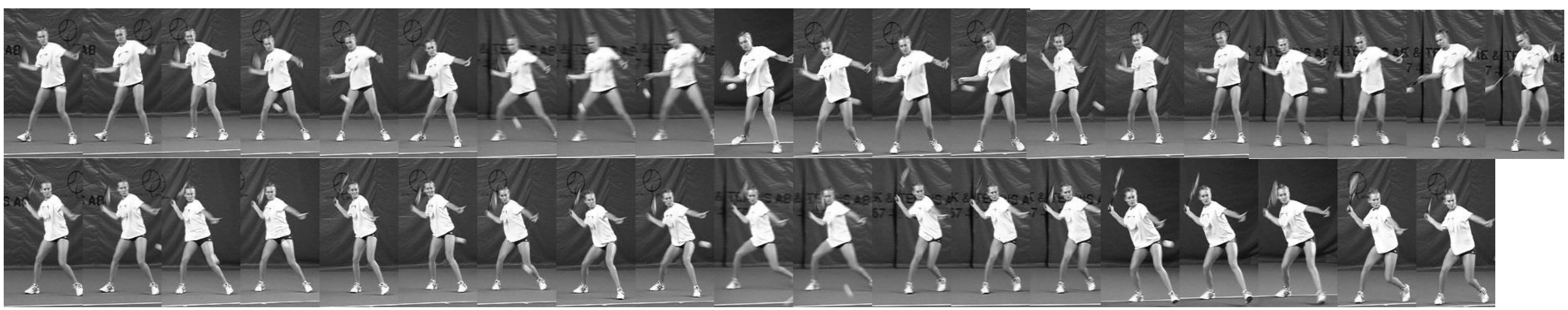
- MNIST dataset
- data set contains handwritten digits from the American Census Bureau employees and American high school students
- 28 x 28 grayscale images
- training set: 60,000 examples
- test set: 10,000 examples.
- Features: each pixel is used as a feature so:
  - there are 28x28 = 784 features
  - each feature = 256 greyscale value
- Task: classify new digits into one of the 10 classes



https://en.wikipedia.org/wiki/MNIST_database

Image Classification



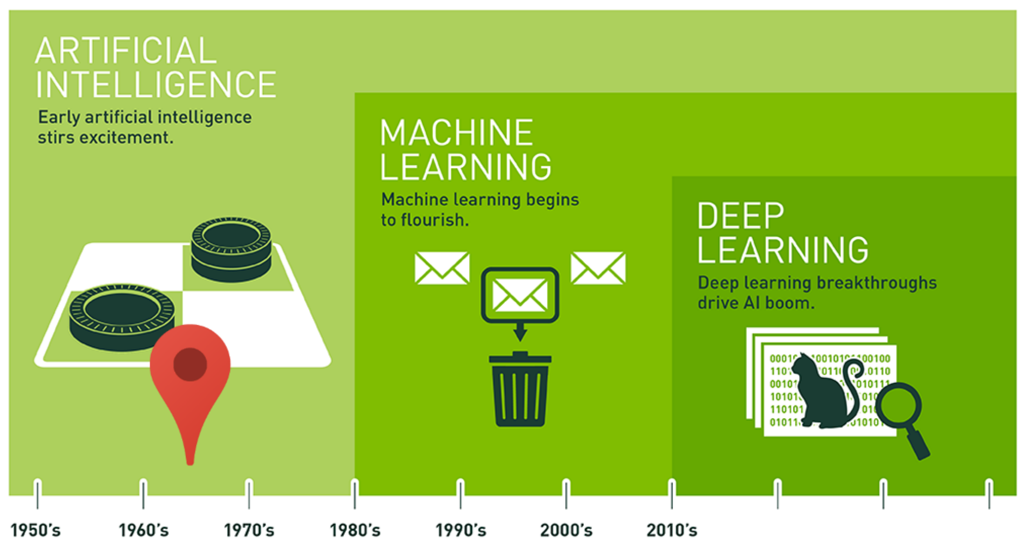


Comments on Naïve Bayes Classification

- Makes a strong assumption of conditional independence
  - that is often incorrect
  - ex: the word *ambulance* is not conditionally independent of the word *accident* given the class SPORTS
- BUT:
  - surprisingly very effective on real-world tasks
  - basis of many spam filters
  - fast, simple
  - gives confidence in its class predictions (i.e., the scores)
- Fast, easy to apply
  - often used as a *baseline* algorithm before trying other methods

Today

1. Introduction to ML
2. Naïve Baye ier
3. Evaluation



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

Evaluation of Learning Model

- How do you know if what you learned is correct?
- You run your classifier on a data set of unseen examples (that you did not use for training) for which you know the correct classification
- Split data set into 3 sub-sets
  1. Actual **training** set (~80%)
  2. **Validation** set (~20%)

  ~80%

  3. **Test** set

  ~20%

Standard Methodology

1. Collect a large set of examples (all with correct classifications)
2. Divide collection into training, validation and test set

Loop:

3. Apply learning algorithm to training set to learn the parameters
4. Measure performance with the validation set, and adjust hyper-parameters* to improve performance

5. Measure performance with the test set

- DO NOT LOOK AT THE TEST SET until step 5.

Parameters:

basic values learned by the ML model. eg.

- for NB: prior & conditional probabilities
- for DTs: features to split
- for ANNs: weights

Hyper-parameters: parameters used to set up the ML model. eg.

- for NB: value of delta for smoothing,
- for DTs: pruning level
- for ANNs: nb of hidden layers, nb of nodes per layer…

Metrics

- Accuracy
  - % of instances of the test set the algorithm correctly classifies
  - when all classes are equally important and represented
- Recall, Precision & F-measure
  - when one class is more important than the others

Accuracy

- % of instances of the test set the algorithm correctly classifies
- when all classes are equally important and represented
- problem:
  - when one class *C* is more important than the others
  - eg. when data set is unbalanced

	Target	*system 1*
	X1 √	X1 ×
	X2 √	X2 ×
	X3 √	X3 ×
	X4 √	X4 ×
	X5 √	X5 ×
	X6 ×	X6 ×
	X7 ×	X7 ×
	…	…
	…	…
	X500 ×	X500 ×

Accuracy = 495/500 = 99%

Recall, Precision, Accuracy

- Recall: How many % of instances of *C* were found correctly?
- Precision: Of the detected instances of *C*, how many % were correct?
- Accuracy: How many % were correct overall (both *C* and not *C*)?
- See confusion matrix:

Model says...	In reality, the instance is... in class *C*	Is not in class *C*
instance is in class *C*	True Positive (TP)	False Positive (FP)
instance is NOT in class *C*	False Negative (FN)	True Negative (TN)

$$\text{Precision} = \frac{TP}{TP+FP} \qquad \text{Recall} = \frac{TP}{TP+FN} \qquad \text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

Example

	Target	*system 1*	*system 2*	*system 3*
	X1 √	X1 ×	X1 √	X1 √
	X2 √	X2 ×	X2 ×	X2 √
	X3 √	X3 ×	X3 √	X3 √
	X4 √	X4 ×	X4 √	X4 √
	X5 √	X5 ×	X5 ×	X5 √
	X6 ×	X6 ×	X6 ×	X6 √
	X7 ×	X7 ×	X7 ×	X7 √
	… ×	…	… ×	… ×
	… ×	…	… ×	… ×
	X500 ×	X500 ×	X500 ×	X500 ×
Accuracy		495/500 = 99% !	498/500 = 99.6%	498/500 = 99.6%
Precision		0/0	3/3 = 100%	5/7 = 71%
Recall		0/5 = 0%	3/5 = 60%	5/5 = 100%

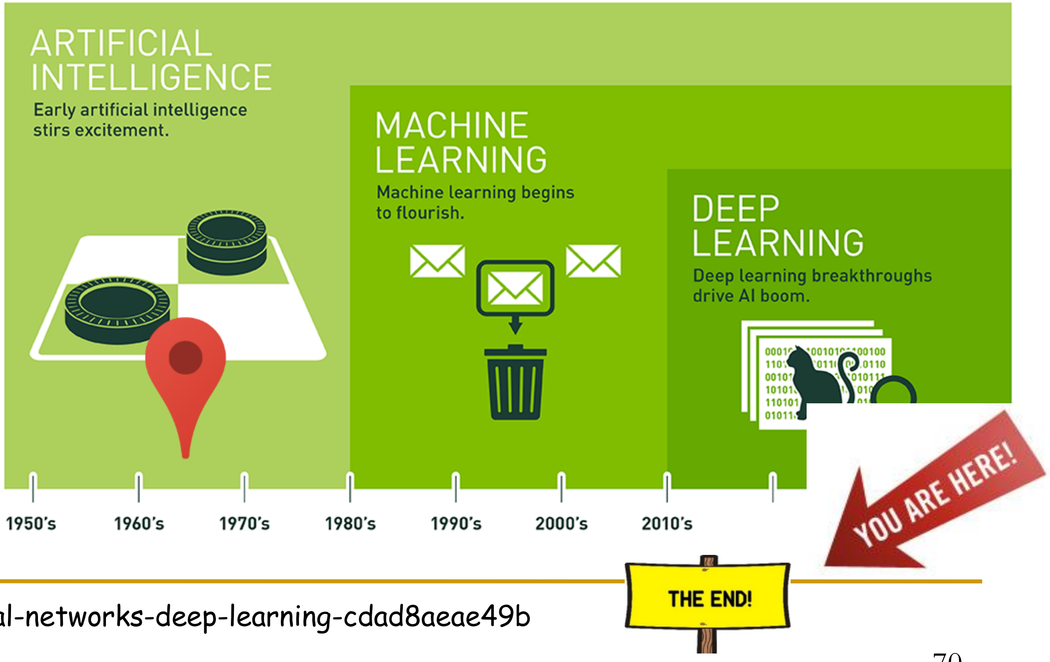
Error Analysis

- Where did the learner go wrong ?
- Use a confusion matrix / contingency table

correct class (that should have been assigned)	classes assigned by the learner							
	C1	**C2**	**C3**	**C4**	**C5**	**C6**	...	**Total**
C1	94	3	0	0	3	0		100
C2	0	93	3	4	0	0		100
C3	0	1	94	2	1	2		100
C4	0	1	3	94	2	0		100
C5	0	0	3	2	92	3		100
C6	0	0	5	0	10	85		100
...								

Today

1. Introduction to ML
2. Naïve Bayes Classifier
3. Evaluation



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