Just a machine that learns

K. Nielbo
kln@cas.dk
knielbo.github.io

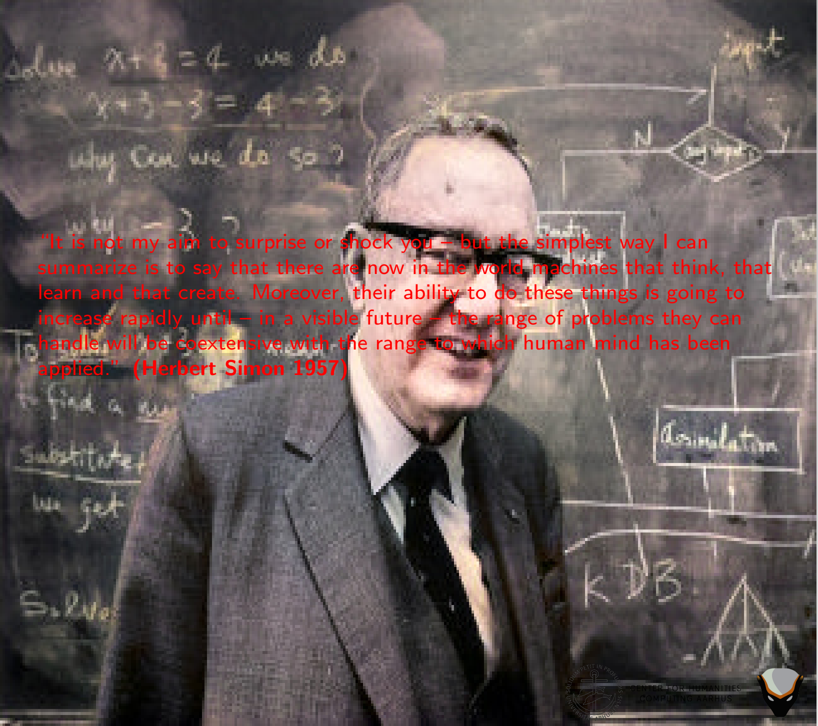
Center for Humanities Computing Aarhus|chcaa.io
Aarhus University, Denmark

CENTER FOR HUMANITIES
COMPUTING AARHUS

outline

CENTER FOR HUMANITIES COMPUTING AARHUS

"It is not my aim to surprise or shock you – but the simplest way I can summarize is to say that there are now in the world machines that think, that learn and that create. Moreover, their ability to do these things is going to increase rapidly until – in a visible future – the range of problems they can handle will be coextensive with the range to which human mind has been applied."



"It is not my aim to surprise or shock you – but the simplest way I can summarize is to say that there are now in the world machines that think, that learn and that create. Moreover, their ability to do these things is going to increase rapidly until – in a visible future – the range of problems they can handle will be coextensive with the range to which human mind has been applied." **(Herbert Simon 1957)** was trained to imitate humans. The second model is fixed, because the researchers found that updating the parameters of both agents led to divergence from human language as the agents developed their own language for negotiating. At the end of every dialog, the agent is given a

Daniel Gross @danielgross

When you let AI negotiate with itself, it realizes there are better options than English. A sign of what's to come.code.facebook.com/posts/16866720…

5:29 AM - Jun 15, 2017

232 144 people are talking about this

Facebook's AI accidentally created its own language

by BRYAN CLARK — 10 months ago in ARTIFICIAL INTELLIGENCE

'Terminator' Come To Life? – Facebook Shuts Down Artificial Intelligence After It Developed Its Own Language

Facebook AI Invents Language That Humans Can't Understand: System Shut Down Before It Evolves Into Skynet

30 July 2017, 9:45 pm EDT By Aaron Mamiit Tech Times

Edward Grefenstette @egrefen

What f***ing trashy excuse of a journalist writes this sh***y sensationalist s***? DO YOU GUYS NOT HAVE EDITORS??digitaljournal.com/tech-and-scien…

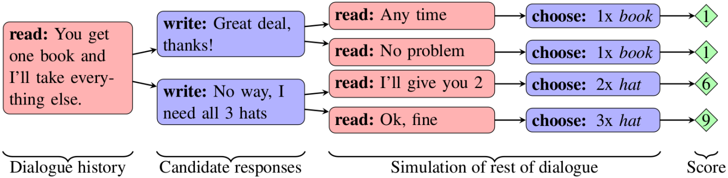
8:20 PM - Jul 29, 2017

Researchers shut down AI that invented its own language

An artificial intelligence system being developed at Facebook has created its own language. It developed a system of code words to

digitaljournal.com

183 85 people are talking about this



I'd like the books and the hats

I will take the books and hats

I really need the books and at least one other object

Ok, you can have one book and one ball

If I can have all the books, I can leave you the rest

Deal

You can have the ball if I can have everything else

No can't do that the ball is not going to give me anything

I need the hats and you can have the rest

Deal

Great! Thank you!

https://github.com/facebookresearch/end-to-end-negotiator i can i i everything else . . . .

balls have zero to me to me to me to me to me to me to me to me to

you i everything else . . . .

balls have a ball to me to me to me to me to me to me to me

i i can i i i everything else . . . .

balls have a ball to me to me to me to me to me to me to me

i . . . .

balls have zero to me to me to me to me to me to me to me to me to

you i i i i i everything else . . . .

balls have 0 to me to me to me to me to me to me to me to me to

you i i i everything else . . . .

balls have zero to me to me to me to me to me to me to me to me to

compromise0: I$_{\text{PRON}}$ will$_{\text{AUX}}$ take$_{\text{VERB}}$ the$_{\text{DET}}$ books$_{\text{NOUN}}$ and$_{\text{CONJ}}$ hats$_{\text{NOUN}}$

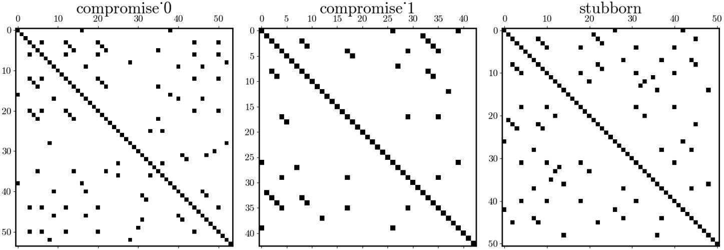
compromise1: You$_{\text{PRON}}$ can$_{\text{AUX}}$ have$_{\text{VERB}}$ the$_{\text{DET}}$ ball$_{\text{NOUN}}$ if$_{\text{SCONJ}}$ I$_{\text{PRON}}$ can$_{\text{AUX}}$ have$_{\text{VERB}}$ everything$_{\text{NOUN}}$ else$_{\text{ADJ}}$

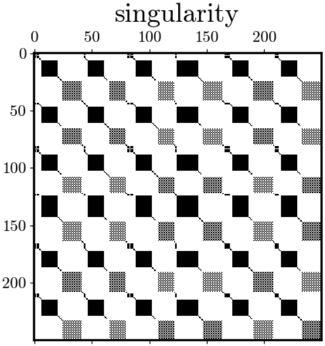
stubborn: I$_{\text{PRON}}$ get$_{\text{VERB}}$ all$_{\text{DET}}$ the$_{\text{DET}}$ balls$_{\text{NOUN}}$?$_{\text{PUNCT}}$

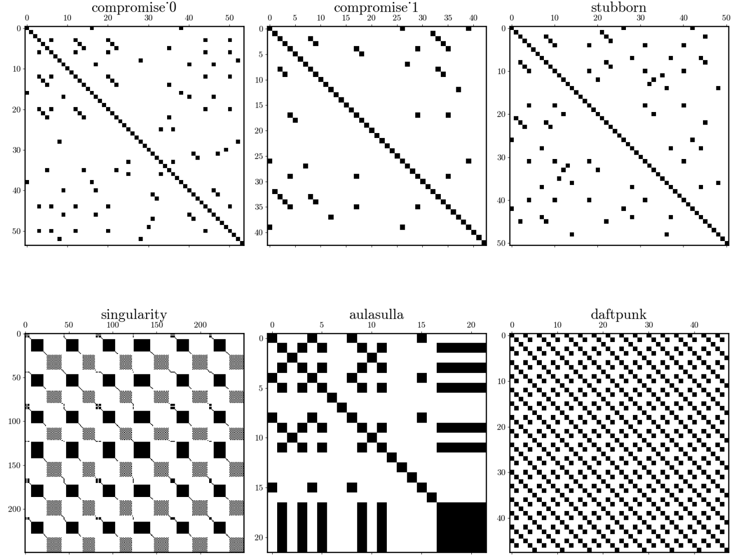
singularity: balls$_{\text{NOUN}}$ have$_{\text{VERB}}$ zero$_{\text{ADJ}}$ to$_{\text{ADP}}$ me$_{\text{PRON}}$ to$_{\text{ADP}}$ me$_{\text{PRON}}$ to$_{\text{ADP}}$ me$_{\text{PRON}}$ to$_{\text{ADP}}$ me$_{\text{PRON}}$ to$_{\text{ADP}}$ me$_{\text{PRON}}$ to$_{\text{ADP}}$ me$_{\text{PRON}}$ to$_{\text{ADP}}$ me$_{\text{PRON}}$ to$_{\text{ADP}}$ me$_{\text{PRON}}$ to$_{\text{PART}}$

	compromise0	**compromise1**	**stubborn**	**singularity**
$H(X)$	2.53 (1.16)	2.3 (1.35)	2.59 (0.84)	1.62 (0.51)
TTR	0.92 (0.09)	0.94 (0.07)	0.96 (0.09)	0.5 (0.27)

CENTER FOR HUMANITIES
COMPUTING AARHUS


compromise˙0
compromise˙1
stubborn
0 10 20 30 40 50
0 5 10 15 20 25 30 35 40
0 10 20 30 40 50


singularity
0
50
100
150
200
0
50
100
150
200


compromise˙0
0 10 20 30 40 50
compromise˙1
0 5 10 15 20 25 30 35 40
stubborn
0 10 20 30 40 50
singularity
0 50 100 150 200
aulasulla
0 5 10 15 20
daftpunk
0 10 20 30 40

Elon Musk
"With Artificial Intelligence, we are summoning the demon"

Andrew Ng
"Fearing a rise of killer robots is like worrying about overpopulation on Mars"

Geoffrey Hinton
"Whether or not it turns out to be a good thing depends entirely on the social system, and doesn't depend at all on the technology"

OpenAI's transformer-based model

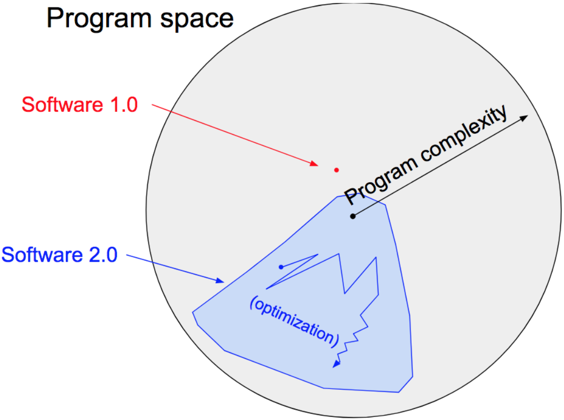
OpenAI on GPT-2

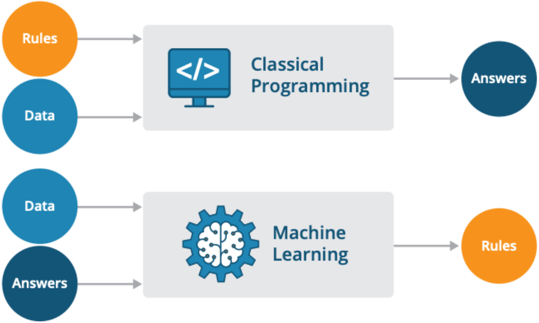
"We've trained a large-scale unsupervised language model which generates coherent paragraphs of text, achieves state-of-the-art performance on many language modeling benchmarks, and performs rudimentary reading comprehension, machine translation, question answering, and summarization—all without task-specific training."

"Due to concerns about large language models being used to generate deceptive, biased, or abusive language at scale, we are only releasing a much smaller version of GPT-2 along with sampling code. We are not releasing the dataset, training code, or GPT-2 model weights."

- **PR Focus** - reporters were given early information
- **Gatekeeping** - malicious uses were hypothesized and we have no way of testing
- **Misdirected** - not releasing affects researchers more than malicious actors due to the model price
- **Dual use** - OpenAI did not discuss dual-use technology

AI from the perspective of software development





Software 1.0 involves manually writing rules. Software 2.0 is about learning these rules from data (credit: S. Charrington)

Andrej Karpathy
"they [neural networks] represent the beginning of a fundamental shift in how we write software."

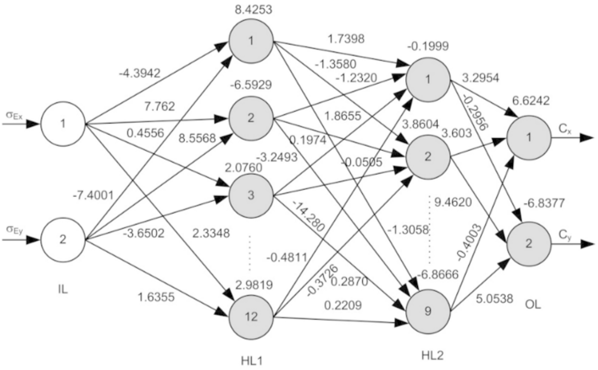
```
1  class Person(object):
2      def __init__(self, name):
3          self.name = name
4      def says_hello(self):
5          print(‘Hello, my name is’, self.name)
6
7  class Researcher(Person):
8      def __init__(self, title=None, areas=None, **kwargs):
9          super(Researcher, self).__init__(**kwargs)
10         self.title = title
11         self.areas = areas
12
13 KLN = Researcher(name = ‘Kristoffer L Nielbo’,\
14         title = ‘Associate professor’,\
15         areas = [‘Humanities Computing’, ‘Culture Analytics’, ‘eScience’])
16
17 KLN.says_hello()
```

Software 1.0

- each line 1–17 produce a behavior (do this, then this ...)
- utilizes a programming language, e.g., Python, C++
- human-friendly code

Software 2.0

- specify some goal on the behavior and write a solution architecture
- search and optimization problem
- abstract weights in a neural network



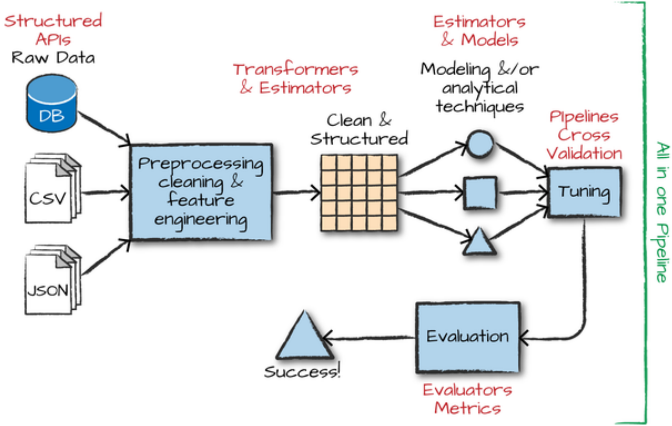
Just a machine that learns

Machine learning emerged from AI - **build a computer system that automatically improves with experience**

- application requires pattern recognition in large data
- application is too complex for a manually designed algorithm
- application needs to customize its operational environment after it is fielded

Mitchell's well-posed learning problem
A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E

Historically, ML is "just" part of the **industrial age's efforts towards perfecting task automation**



Machine learning pipeline (credit: Spark - The Definitive Guide)

Traditionally, ML pipelines have often overlooked the importance of **data curation and data lifecycle management**

Vikram Bisen
Though, nowadays many tasks can be independently performed by AI-enabled devices, systems or machines without the help of humans. But developing such machines is not possible without the help of humans. So, Human-in-the-Loop or HITL is a model or concept require human interaction.

Human-in-the-Loop is a process where optimization of learning algorithms requires human intervention

- algorithms are not understanding the input
- when data input is interpreted incorrectly
- algorithms do not know how to perform the task
- to make the machine learning model more accurate
- when the cost of errors is too high in development
- when the data is rare or not available

Humanities research meets machine learning

As a consequence of the data surge, we are (also) “jumping the automation bandwagon”

– plus theoretical innovations that rely on ML/DL (e.g., lexical $\rightarrow$ compositional semantics)

Inherent challenges in data and users

– data are unstructured, heterogeneous, need normalization, low resource varieties

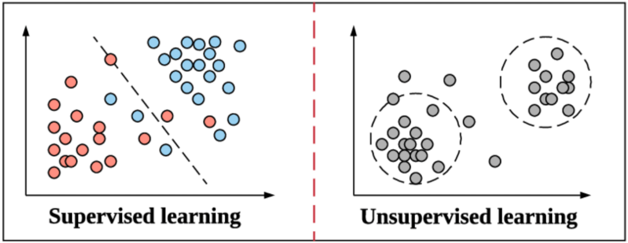
– users lack of computational literacy, gab between technology and domain knowledge

Types of problems solved by ML:

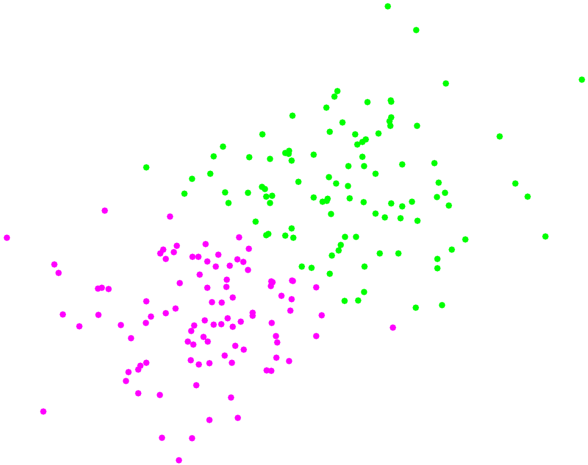
– initially ML was the solution to a(-ny) research problem

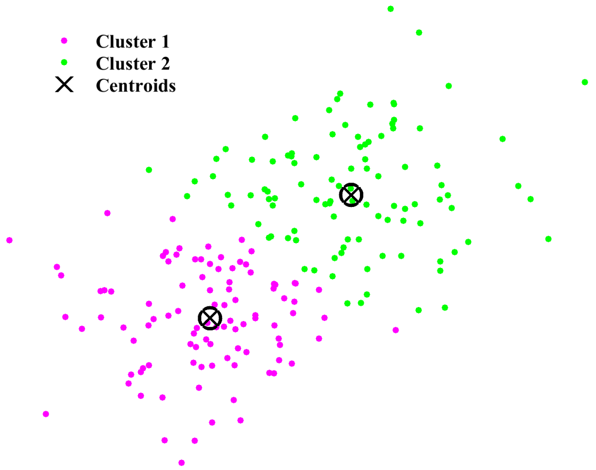
– increasingly, ML solves auxiliary tasks related to automation

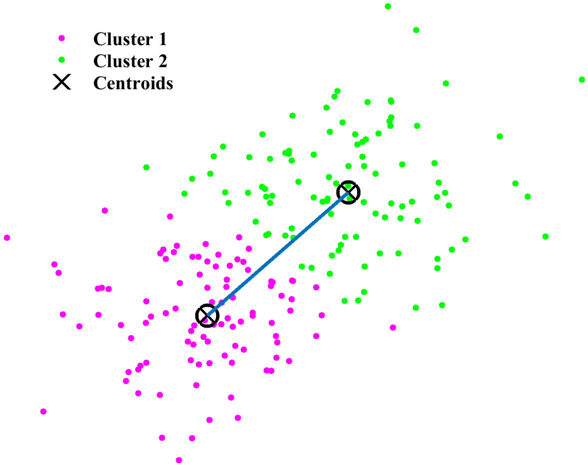
Supervised learning
machine learning algorithms used to draw inferences from data sets consisting of input data with labeled responses

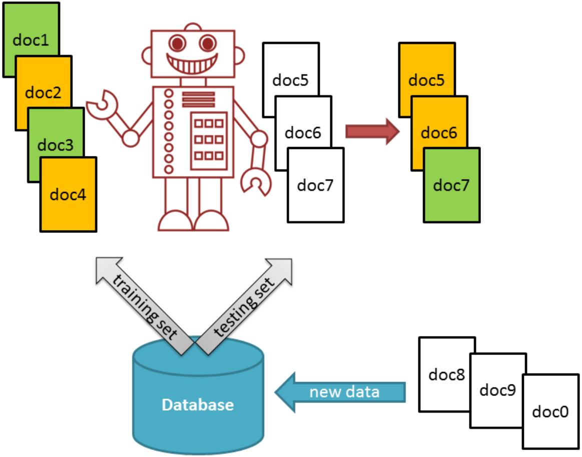


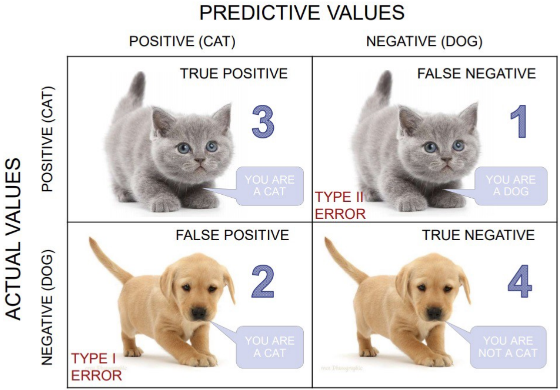
Unsupervised learning
machine learning algorithms used to draw inferences from data sets consisting of input data without labeled responses




Cluster 1
Cluster 2
Centroids


Cluster 1
Cluster 2
Centroids


doc1
doc2
doc3
doc4
doc5
doc6
doc7
doc5
doc6
doc7
training set
testing set
Database
new data
doc8
doc9
doc0



Confusion matrix for binary classification task (credit: Towards Data Science)

		PREDICTED	
		positive	negative
TRUE	positive	TP	FN
	negative	FP	TN

TP Correctly assigns positive class membership

TN Correctly rejects class membership

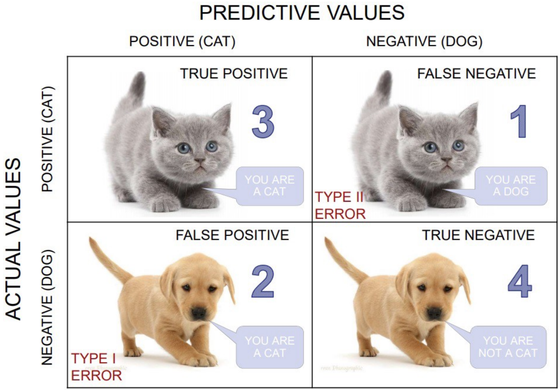
FP Fail to rejects class membership (Type I error)

FN Rejects class membership incorrectly (Type II error)

Prediction Accuracy (ACC): $\frac{TP+TN}{TP+TN+FP+FN}$

Precision (P) $= \frac{TP}{TP+FP}$

Recall (R) $= \frac{TP}{TP+FN}$

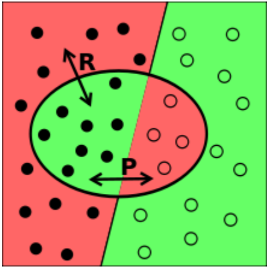


Confusion matrix for binary classification task (credit: Towards Data Science)

Prediction Accuracy (ACC): $\frac{TP+TN}{TP+TN+FP+FN} = \frac{3+4}{3+4+2+1} = 0.7$

Precision (P) $= \frac{TP}{TP+FP} = \frac{3}{3+2} = 0.6$

Recall (R) $= \frac{TP}{TP+FN} = \frac{3}{3+1} = 0.75$



← `relevant objects` (e.g., cat, ham)

→ `irrelevant objects` (e.g., dog, spam)

⬭ objects classified with relevant class label

`ERROR`

`CORRECT`

Precision: fraction of retrieved instances that are relevant

$$P = \frac{TP}{TP + FP}$$

Recall: fraction of relevant instances that are retrieved

$$R = \frac{TP}{TP + FN}$$

P and R are inversely related. Identify balance through a Precision-Recall curve.

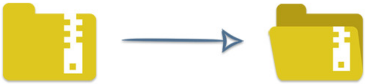
Impossibility results

> "Suppose we want to determine the risk that a person is a carrier for a disease Y, and suppose that a higher fraction of women than men are carriers. Then our results imply that in any test designed to estimate the probability that someone is a carrier of Y, at least one of the following undesirable properties must hold: (a) the test's probability estimates are systematically skewed upward or downward for at least one gender; or (b) the test assigns a higher average risk estimate to healthy people (non-carriers) in one gender than the other; or (c) the test assigns a higher average risk estimate to carriers of the disease in one gender than the other. The point is that this trade-off among (a), (b), and (c) is not a fact about medicine; it is simply a fact about risk estimates when the base rates differ between two groups"

Assume differing base rates, $Pr_a(Y=1) \neq Pr_b(Y=1)$, and an imperfect learning algorithm, $C \neq Y$, then you cannot simultaneously achieve:

- **Precision parity** $Pr_a(Y=1 \mid C=1) = Pr_b(Y=1 \mid C=1)$
- **True positive parity** $Pr_a(C=1 \mid Y=1) = Pr_b(C=1 \mid Y=1)$
- **False positive parity** $Pr_a(C=1 \mid Y=0) = Pr_b(C=1 \mid Y=0)$

Kleinberg J., S. Mullainathan, & M. Raghavan (2016), Inherent Trade-Offs in the Fair Determination of Risk Scores, arXiv:1609.05807



Ethical issues

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	**racist/sexist robots**

top nine ethical issues identified by J. Bossmann (credit: T. Eliassi-Rad)

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

"the threat of automation & the future of work"

CENTER FOR HUMANITIES COMPUTING AARHUS

unemployment
wealth inequality
humanity

artificial stupidity
evil genies
singularity

security
robot rights
racist/sexist robots

if end of work, then “shared prosperity” or “increasing inequality”

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

AI altering human behaviors and interactions, ex. fake news, click-baiting

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

adversarial ML that exploits stupidity

CENTER FOR HUMANITIES COMPUTING AARHUS

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

unintended consequences due to poorly defined tasks or faulty experience/data

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

the possibility of a super-intelligence emerging for AI

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

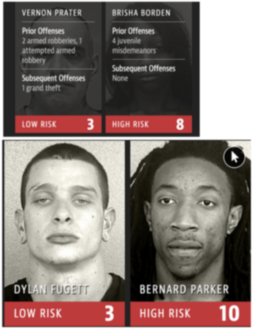
weaponization of AI in both physical and cyberspace

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

when is a robot a moral agent?

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

fairness, accountability, and transparency for AI regarding biases



racially biased COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) risk scores (credit: ProPublica)

assessment tool correctly predicts subsequent offence in 0.61 cases, BUT the accuracy is not uniform for whites and african americans

class	white	african american
high risk & not re-offend	.24	.45
low risk & re-offend	.48	.28

$$P(low|white) > P(low|black) \;\&\; P(high|white) < P(high|black)$$

		Predictive values		
		Positive	Negative	Total
Actual values	Positive	TP	FN	$TP + FN$
	Negative	FP	TN	$FP + TN$
	Total	$TP + FP$	$FN + TN$	N

TP: model correctly predicts the positive class

TN: model correctly predicts the negative class

FP: model incorrectly predicts the positive class

FN: model incorrectly predicts the negative class

CENTER FOR HUMANITIES COMPUTING AARHUS

a wolf/no wolf classifier for confusion matrix:

TP	FN
FP	TN

wolf	wolf
no wolf	no wolf

state matrix for binary classification

‘wolf’	‘no wolf’
‘wolf’	‘no wolf’

shepherd statement matrix for binary classification

shepherd:hero	sheep:dead
villagers:angry	everyone:no problem

outcome matrix for binary classification

"Untergang der Titanic" by Willy Stöwer, 1912

		Predictive values		
		Survived	Dead	Total
Actual values	Survived	68	41	109
	Dead	17	142	159
	Total	85	183	268

accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	0.78
precision	$\frac{TP}{TP+FP}$	0.62

Data from Kaggle Titanic: Machine Learning from Disaster (credit: Vikrant Sahu)

MALE

		Predictive values		
		Survived	Dead	Total
Actual values	Survived	4	33	37
	Dead	5	132	137
	Total	9	165	174

accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	0.78
precision	$\frac{TP}{TP+FP}$	0.11

FEMALE

		Predictive values		
		Survived	Dead	Total
Actual values	Survived	64	8	72
	Dead	12	10	22
	Total	76	18	94

accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	0.78
precision	$\frac{TP}{TP+FP}$	0.89

– the model fails to predict the survival of 0.89 male in contrast to only 0.11 female passengers because its has learned that:

$$BIAS : P(survival|woman) > P(survival|man)$$

bias in computer systems

preexisting
originates in social institutions, practices, and attitudes → computer systems embody biases that exist independently, and usually prior to the creation of the system

technical
product of technical constraints or consideration due to limitations of computer tools (e.g., databases, hardware), decontextualized algorithms, random number generation, and formalization of human constructs

emergent
arises in a context of use with real users as a result of changing societal knowledge, population, or cultural values (e.g., new societal knowledge, mismatch between user and system design)

"We conclude by suggesting that freedom from bias should be counted among the select set of criteria – including reliability, accuracy, and efficiency – according to which the quality of systems in use in society should be judged"

Friedman & Nissenbaum, 1996, Bias in Computer Systems, ACM Transactions on Information Systems

fairness ⇒ parity

"fairness" is probabilistically defined as *parity*

- many parity definitions: demographic, accuracy, true positive, predictive value, **precision**, ...
- Fairness and machine learning – Limitations and Opportunities
- Decisions should be in some sense **probabilistically independent of sensitive features values** (such as gender, race)

ensure that common measures of predictive performance are equal across all classes

$$Pr_{male}(Y = 1 \mid C = 1) = Pr_{female}(Y = 1 \mid C = 1)$$
$$0.11 \neq 0.89$$

iow: the titanic survival rate classifier does not achieve **precision parity**

CENTER FOR HUMANITIES COMPUTING AARHUS

Impossibility results revisited

X is a dataset that contains feature on an individuals (e.g., income level, age)
– X incorporates all sorts of measurement biases
A is a sensitive attribute (e.g., ethnicity, religion, gender)
– A is often unknown, ill-defined, misreported, or inferred
Y is the true outcome (i.e., ground truth, e.g., survival)
C is an ML algorithm that uses X and A to predict the value of Y (e.g., whether a passenger survives)

– the sensitive attribute A divides the population into two groups a (e.g., male) and b (e.g., female)
– the ML algorithm C outputs 0 (e.g.. predicts dead) and 1 (e.g, predicts survive)
– the true outcome Y is 0 (e.g., dead) and 1 (e.g., survive)

---

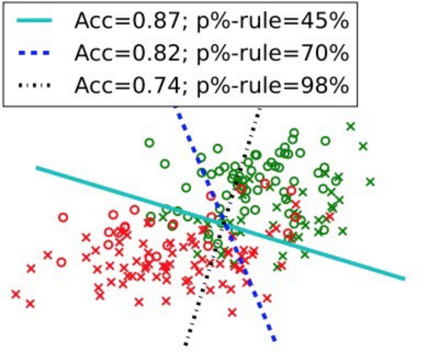
then you cannot simultaneously achieve,

$$Pr_a(Y = 1 \mid C = 1) = Pr_b(Y = 1 \mid C = 1)$$
$$Pr_a(C = 1 \mid Y = 1) = Pr_b(C = 1 \mid Y = 1)$$
$$Pr_a(C = 1 \mid Y = 0) = Pr_b(C = 1 \mid Y = 0)$$

or, precision parity and equalized odds are not simultaneously possible

How to achieve parity?

The trade-off among P, TP and FP is simply a fact about risk estimates when the base rates differ between two or more groups!



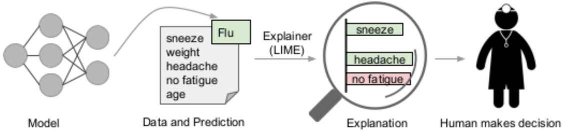
Simple models allow for fine-grained control on the degree of fairness, often at a small cost in terms of accuracy

Demographic Parity, also called Independence, Statistical Parity, is one of the most well-known criteria for fairness.

$$C \text{ is independent of } A \text{ if } Pr_a(C = c) = Pr_b(C = c) \forall c \in \{0, 1\}$$

M. B. Zafar, I. Valera, M. G. Rodriguez, and K. P. Gummadi (2015) Fairness Constraints: Mechanisms for Fair Classification, arXiv:1507.05259

Solutions



LIME, an algorithm that can explain the predictions of any classifier or regressor in a faithful way, by approximating it locally with an interpretable model (source: 1602.049338:arXiv)

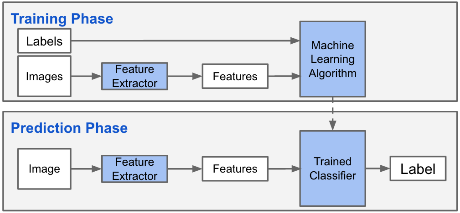
Technical

- proprocessing the data to make it less biased
- learn fair representations that encode data while obfuscating sensitive attributes
- penalize the algorithm to encourage it to learn fairly
- allow the sensitive attributes during training, but not during inference time
- causal inference

Policy

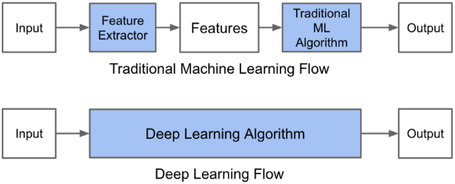
- regulations (e.g., GDPR)
- laws that grant users the right to a logical explanation of how an algorithm uses our personal data
- explainability at the level of predictive performance

Basic supervised pipeline



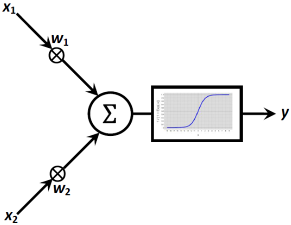
Machine Learning Phases

The emergence of deep learning



Neurons

Basic computational unit of a neural network



A neuron takes inputs, x_1, x_2, does *some math on them*, and generates an output, y

The input is weighted

$$x_1 \rightarrow x_1 \times w_1$$
$$x_2 \rightarrow x_2 \times w_2$$

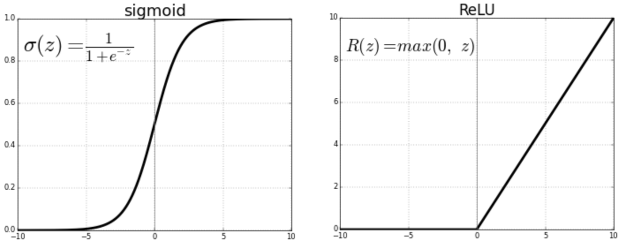
then added with a bias

$$(x_1 \times w_1) + (x_2 \times w_2) + b$$

and finally passed through an activation function

$$y = f(x_1 \times w_1 + x_2 \times w_2 + b)$$

A word on the activation functions



The sigmoid activation function "squashes" an unbounded $(-\infty, +\infty)$ to a bounded $(0, 1)$ set. Computationally simpler activation functions, such as rectifiers, have to a large extent replaced sigmoids.

Example

cat/dog classifier where x_1 "has fur" and x_2 "barks" and we are generally more likely to encounter dogs, so when "it has fur and barks", then:

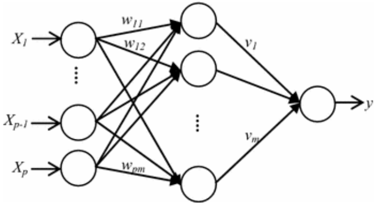
$$
\begin{aligned}
w &= [0, 1] \\
b &= 2
\end{aligned}
$$

$$
\begin{aligned}
(w \cdot x) + b &= ((w_1 \times x_1) + (w_2 \times x_2)) + b \\
&= 1 \times 0 + 1 \times 1 + 2 \\
&= 3
\end{aligned}
$$

$$f(w \cdot x + b) = f(3) = \frac{1}{1 + e^{-}3} = 0.953$$

Neurons in a network

An artificial neural network is just a set of neurons wired together (typically) in a layered structure.



Feedforward neural network with one hidden layer of size *m*. A hidden layer is any layer between the input and output. Hidden layers perform transformations on the input or previous hidden layers. A network can have many hidden layers.

A neural network can have any number of neurons and layers. *Deep* in deep learning just refers to representations learned in multi-layered (deep) structures. The core idea is to propagate input forward through the transformations of the hidden layers in order to get an output.

Example

continue example from before (cat/dog), with one hidden layer and two hidden units, $w = [0, 1]$, $b = 0$, and $x = [0, 1]$:

$$
\begin{aligned}
h_1 = h_2 &= f(w \cdot x + b) \\
&= f((0 \times 0) + (1 \times 1) + 0) \\
&= f(1) \\
&= 0.731
\end{aligned}
$$

$$
\begin{aligned}
o_1 &= f(w \cdot [h_1, h_2] + b) \\
&= f((0 \times h_1) + (1 \times h_2) + 0) \\
&= f(0.731) \\
&= 0.675
\end{aligned}
$$

Training the model

It is impossible to compute the perfect weights for a neural network. Instead learning becomes an optimization problem and algorithms are used to run through the space of possible weights that the model can use to make a good prediction.

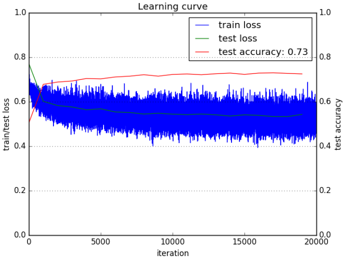


Figure: Training is an optimization problem: minimizing loss function

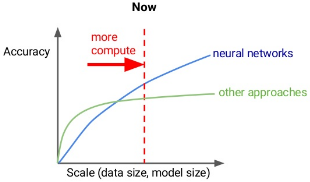


Figure: Currently there seems to be no upper limit on performance - except for the perfect classifier

– Training consists of iteratively adjusting the weights in order to minimize a loss function.

– Neural network models are typically trained using the *gradient descent* optimization algorithm and weights are updated using the backpropagation (of error) algorithm

Loss function

Mean squared error loss:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_{true} - y_{pred})^2$$

– a good prediction lowers loss $\rightarrow$ training a network $\sim$ trying to minimize loss

– iow: a loss function maps the networks output onto the “loss” associated with a prediction $\sim$ evaluated how well the neural network captures the data structure

If the goal is to minimize loss of the network, the loss is a function of weights w and biases b. For a fully connected one-layered feedforward network $(2 \times 2 \times 1)$ then:

$$L(w_1, w_2, w_3, w_4, w_5, w_6, b_1, b_2, b_3)$$

Modifying w_1 then, will change L as $\frac{\partial L}{\partial w_1}$. Using the chain rule:

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial y_{pred}} \times \frac{\partial y_{pred}}{\partial w_1}$$

Assume a simple binary classifier, $True : 1$, $MSE = (1 - y_{pred})^2$, then:

$$\frac{\partial L}{\partial y_{pred}} = \frac{\partial (1 - y_{pred})^2}{\partial y_{pred}} = -2(1 - y_{pred})$$

CENTER FOR HUMANITIES COMPUTING AARHUS

For $\frac{\partial y_{pred}}{w_1}$, let h_1, h_2, o_1 be the output of the neurons they represent, then:

$$y_{pred} = o_1 = f(w_5 h_1 + w_6 h_2 + b_3)$$

where f is the sigmoid activation function.

Because w_1 only modulates h_1 and not h_2:

$$\frac{\partial y_{pred}}{w_1} = \frac{\partial y_{pred}}{\partial h_1} \times \frac{\partial h_1}{\partial w_1}$$

and with the chain rule:

$$\frac{\partial y_{pred}}{\partial h_1} = w_5 \times f'(w_5 h_1 + w_6 h_2 + b_3)$$

Repeat procedure for $\frac{\partial h_1}{\partial w_1}$:

$$h_1 = f(w_1 x_1 + w_2 x_2 + b1)$$

$$\frac{\partial h_1}{\partial w_1} = x_1 \times f'(w_1 x_1 + w_2 x_2 + b1)$$

Compute the derivative of the sigmoid function:

$$f(x) = \frac{1}{1 + e^{-x}}$$

$$f'(x) = \frac{e^{-x}}{(1 + e^{-x})^2} = f(x) \times (1 - f(x))$$

Put it all together and we can compute:

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial y_{pred}} \times \frac{\partial y_{pred}}{\partial h_1} \times \frac{\partial h_1}{\partial w_1}$$

as:

$$-2(1 - y_{pred}) \times w_5 \times f'(w_5 h_1 + w_6 h_2 + b_3) \times x_1 \times f'(w_1 x_1 + w_2 x_2 + b1)$$

BACKPROPAGATION The system of computing the partial derivatives by working backwards. Backpropagation in this form was derived by Stuart Dreyfus in 1962.

Dreyfus, S (1962). The numerical solution of variational problems. Journal of Mathematical Analysis and Applications. 5(1)

Training with Backprop

The most widely used training algorithm is *Stochastic Gradient Descent*, which is a set of formal steps for modifying weights and biases to minimize loss:

$$w_1 \leftarrow w_1 - \eta \frac{\partial L}{\partial w_1}$$

where the learning η rate controls the speed of training

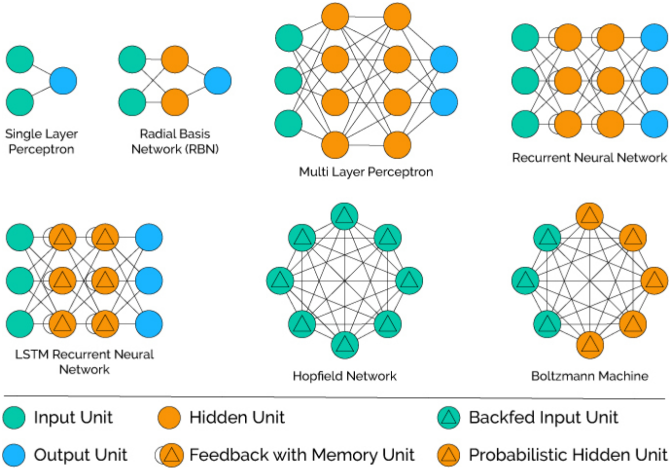
- if $\frac{\partial L}{\partial w_1}$ is positive, then w_1 will decrease and L decrease
- if $\frac{\partial L}{\partial w_1}$ is negative, then w_1 will increase and L decrease

Algorithm 1 Gradient Descent

while $t < maxiter$ **do**
    for *all* i, j **do**
        $w_{ij} = w_{ij} - \eta \frac{\partial L}{\partial w_{ij}}$
    end for
end while

Underlying AI is just rather "dumb" system that improves its performance on a pre-specified task over time by **recursively sending the output of its computations backwards to the parent**.

ANN architectures



THANKS

kln@au.dk
knielbo.github.io
chcaa.io

SLIDES

knielbo.github.io/files/kln_somewhere.pdf

CENTER FOR HUMANITIES
COMPUTING AARHUS