DL3: MACHINE LARNING AND AI

Kristoffer Nielbo

AARHUS
UNIVERSITY
CENTER FOR HUMANITIES COMPUTING

CENTER FOR
HUMANITIES
COMPUTING

OUTLINE

1. Just a Machine that Learns
2. Software 2.0
3. Easy as Pie...
4. Conceptual Prerequisites
5. Ethical Issues

OUTLINE

1. Just a Machine that Learns
2. Software 2.0
3. Easy as Pie...
4. Conceptual Prerequisites
5. Ethical Issues

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."

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

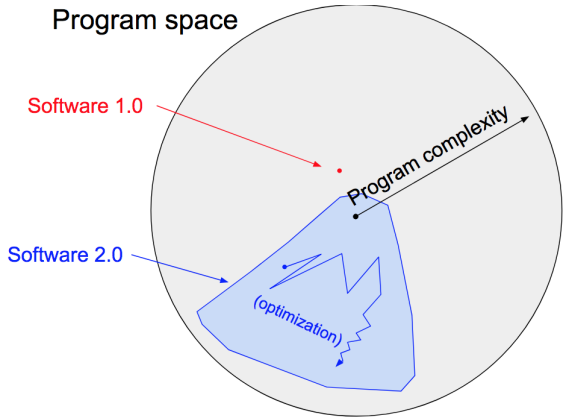
Mitchell's *well-posed learning problem*
A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E

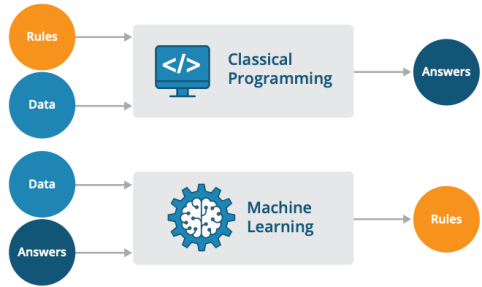
Historically, ML is "just" part of the **industrial age's efforts towards perfecting task automation**

OUTLINE

1. Just a Machine that Learns
2. Software 2.0
3. Easy as Pie...
4. Conceptual Prerequisites
5. Ethical Issues

Software 2.0


Program space
Software 1.0
Program complexity
Software 2.0
(optimization)



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."

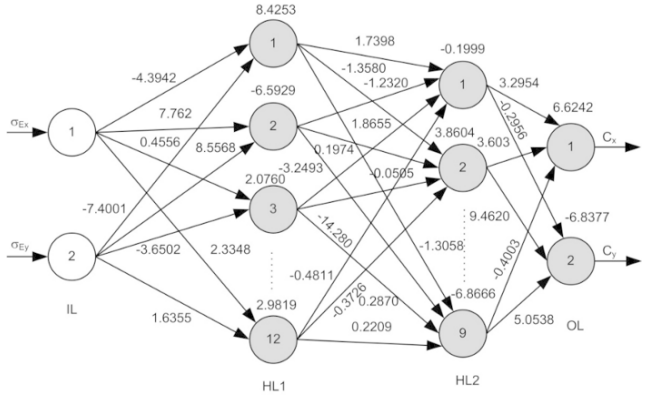
```
1  class Person(object):
2      def __init__(self, name):
3          self.name = name
4      def says_hello(self):
5          print('Hello, my name is', self.name)
6
7  class Researcher(Person):
8      def __init__(self, title=None, areas=None, **kwargs):
9          super(Researcher, self).__init__(**kwargs)
10         self.title = title
11         self.areas = areas
12
13 KLN = Researcher(name = 'Kristoffer L Nielbo',\
14         title = '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

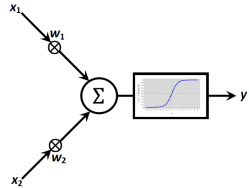
- specify some goal on the behavior and write a solution architecture
- search and optimization problem
- abstract weights in a neural network



OUTLINE

1. Just a Machine that Learns
2. Software 2.0
3. **Easy as Pie...**
4. Conceptual Prerequisites
5. Ethical Issues

Computational unit for learning



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

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

Algorithmic Learning

a *Loss Function* maps the models output onto the "loss" associated with a prediction

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

GOAL: 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:

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

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:

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

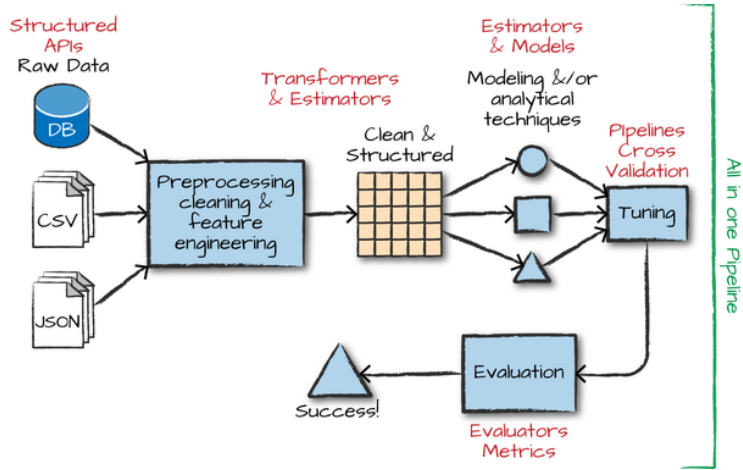
as:

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

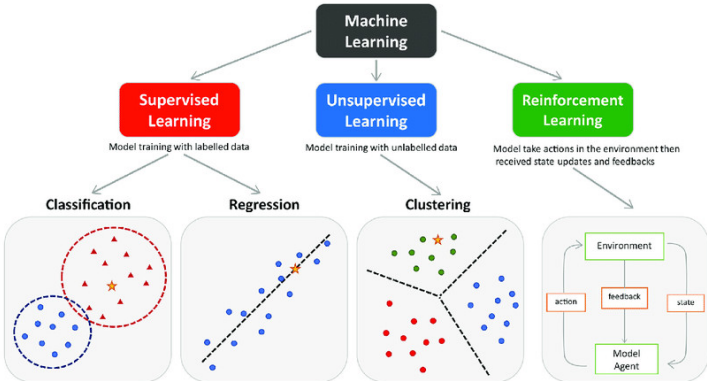
BACKPROPAGATION The system of computing the partial derivatives by working backwards. Backpropagation in this form was derived by Stuart Dreyfus in 1962.

OUTLINE

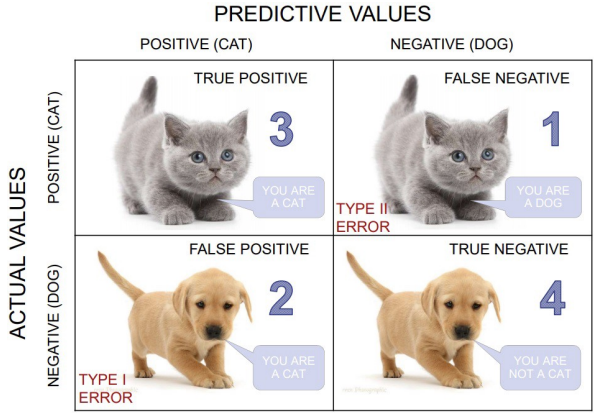
1. Just a Machine that Learns
2. Software 2.0
3. Easy as Pie...
4. **Conceptual Prerequisites**
5. Ethical Issues



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


Machine Learning
Supervised Learning
Model training with labelled data
Unsupervised Learning
Model training with unlabelled data
Reinforcement Learning
Model take actions in the environment then received state updates and feedbacks
Classification
Regression
Clustering
Environment
action
feedback
state
Model Agent

Machine Learning



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

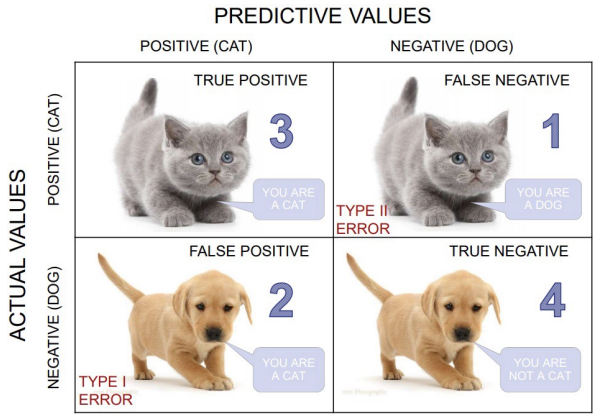
FP Fail to rejects class membership (Type I error)

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

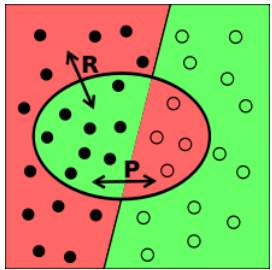


Confusion matrix for binary classification task (credit: Towards Data Science)

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`

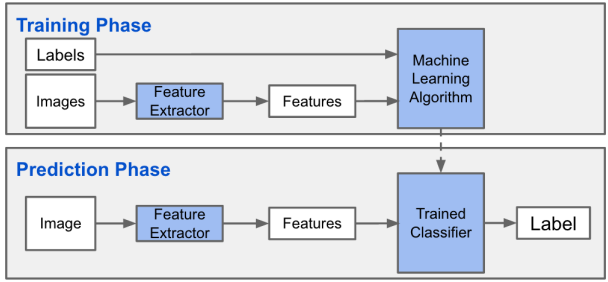
Precision: fraction of retrieved instances that are relevant

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

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.

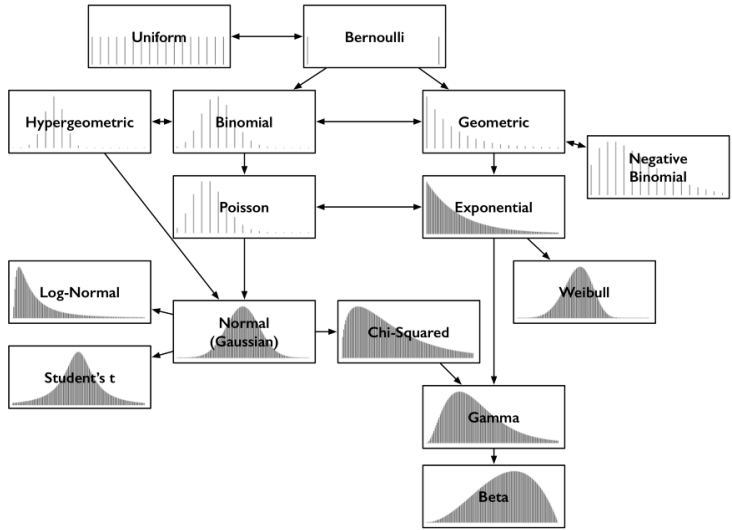


Machine Learning Phases

Traditional Machine Learning Flow

Deep Learning Flow

Statistics


Uniform
Bernoulli
Hypergeometric
Binomial
Geometric
Negative Binomial
Poisson
Exponential
Log-Normal
Normal (Gaussian)
Chi-Squared
Weibull
Student's t
Gamma
Beta

OUTLINE

1. Just a Machine that Learns
2. Software 2.0
3. Easy as Pie...
4. Conceptual Prerequisites
5. **Ethical Issues**

Impossibility results in AI

"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"[1]

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

Ethical issues in ML

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

top nine ethical issues identified by J. Bossmann.

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

"the threat of automation & the future of work"

unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	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
wealth inequality
humanity

artificial stupidity
evil genies
singularity

security
robot rights
racist/sexist robots

adversarial ML that exploits stupidity

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

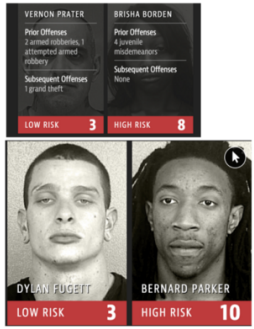
weaponization of AI in both physical and cyberspace

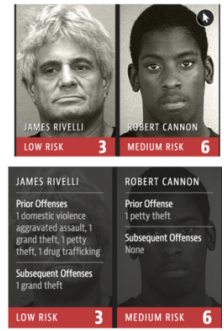
unemployment	artificial stupidity	security
wealth inequality	evil genies	robot rights
humanity	singularity	racist/sexist robots

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 offense 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

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

"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

Impossibility results revisited

X is a dataset that contains features 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,

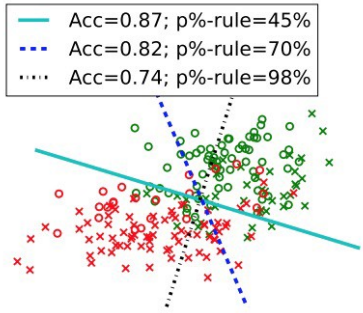
$$Pr_a(Y = 1 \mid C = 1) = Pr_b(Y = 1 \mid C = 1)$$

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

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

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!

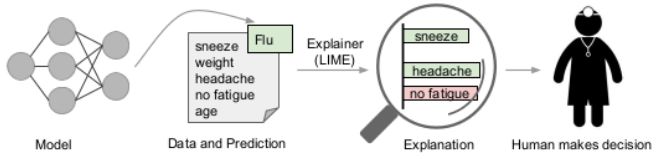


Simple models allow for fine-grained control on the degree of fairness, often at a small cost in terms of accuracy

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

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

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

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

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)

```
if questions:
   try:
      answer()
   except RunTimeError:
      pass
else:
   print('THANKS')
```

THANKS

kln@cas.au.dk
chc.au.dk
knielbo.github.io

SLIDES

knielbo.github.io/files/kln_<fname>.pdf

ACKNOWLEDGEMENTS

...

[1] Kleinberg, J., Mullainathan, S., and Raghavan, M. (2016). Inherent trade-offs in the fair determination of risk scores.