What is the definition of a **good Machine Learning algorithm**?

*After 60 years, is this a closed problem? And if not …*
AI and ML everywhere in the medias today
1. What does work

2. Limitations

3. Learning comes with which guarantees?
   - Induction: how to win this game?
   - The statistical learning theory
   - A closed case? Not so sure

4. Other paradigms? An historical perspective

5. Is there a paradigmatic change in sight?

6. Conclusions
What does work
Object recognition in images

The ImageNet competition

- More than 15M high resolution labeled images
- Approximately 22K categories
- Taken from the Web and labeled using Amazon Mechanical Turk
Illustration: ImageNet

The ImageNet competition

• More than 15M high resolution labeled images
• Approximately 22K categories
• Taken from the Web and labeled using Amazon Mechanical Turk
Results: 8 ILSVRC-2010 test images

- Results
Object recognition

Test Image

Retrieved Images

[Krizhevsky, Sutskever and Hinton (2012)]
Image annotating

Figure 2.11: “A group of young people playing a game of frisbee”—that caption was written by a computer with no understanding of people, games or frisbees.
The SuperVision network

Image classification with deep convolutional neural networks


- 7 hidden “weight” layers
- 650K neurons
- **60M** parameters
- 630M connections
Speech recognition

- Works reasonably well

Comparison (2012) of the word error rates achieved by traditional GMMs and DNNs, reported by three different research groups on three different benchmark.

GMM  DNN
Switchboard (Microsoft)  27.4%  18.5%
Youtube (Google)  52.3%  47.6%
Broadcast News (IBM)  17.2%  14.9%
Machine translation

- Still far from perfect, but ...

From Hofstädter (2018)
Game playing with Reinforcement Learning

- E.g. AlphaGo

![Diagram of AlphaGo gameplay]
Outline

1. What does work

2. Limitations

3. Learning comes with which guarantees?
   - Induction: how to win this game?
   - The statistical learning theory
   - A closed case? Not so sure

4. Other paradigms? An historical perspective

5. Is there a paradigmatic change in sight?

6. Conclusions
Limitations
Requires enormous training sets

- Image recognition
  - Object localization for 1000 categories.
    Millions of images
Requires enormous training sets

- Image recognition
  - Object localization for 1000 categories.
    
    Millions of images

- AlphaGo

iCube speech - 2018 « What is a good ML algorithm? » (A. Cornuéjols)
Requires enormous training sets

- **Image recognition**
  - Object localization for 1000 categories.
  - *Millions of images*

- **AlphaGo**
  - Training on KGS dataset led to overfitting
  - Self-play data (**30 million** distinct positions, each sampled from a separate game)
Requires enormous training sets

- **Image recognition**
  - Object localization for 1000 categories.
    - Millions of images

- **AlphaGo**
  - Training on KGS dataset led to overfitting
  - Self-play data (30 million distinct positions, each sampled from a separate game)
  - Over the course of millions of AlphaGo vs AlphaGo games, the system progressively learned the game of Go from scratch, accumulating thousands of years of human knowledge during a period of just a few days. (In the first three days AlphaGo Zero played 4.9 million games against itself in quick succession.)
Exclusively focused on error rate

• The Netflix prize
  – The winner system was not used afterwards!!

• Machine translation
  – Good on easy and mundane texts
  – Bad on interesting texts
Weak account of the structure

- **Texts** as bags of words
- **Images** as simple correlations

Example: detection of the action “giving a phone call”

[Oquab et al., CVPR (2014)] (~70% correct (SOTA))
Weak account of the structure

Example: detection of the action “giving a phone call”

The learning algorithm is statistically correct!

In a typical image dataset, when an image shows a person near a phone (both in the same image), chances are that the person is giving a phone call.
Learning systems do not work together flawlessly

- Two sub-systems
  - One locating the ads links
  - The other the adds
- That influence each other
  - Each takes into account the clicks
  - Which depends in part from the actions of the other sub-system
  - In addition of other uncontrolled factors (price, user’s queries, ...)

The Simpson’s paradox

- Physicians would like to know whether drug A is more or less efficient than drug B
- Two groups of 350 patients each are chosen. One is given drug A, and the other drug B

<table>
<thead>
<tr>
<th>Treatment A: Open surgery</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>78% (273/350)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Treatment B: Percutaneous nephrolithotomy</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>83% (289/350)</td>
</tr>
</tbody>
</table>

B is best?
### The Simpson’s paradox

<table>
<thead>
<tr>
<th></th>
<th>Overall</th>
<th>Patients with small stones</th>
<th>Patients with large stones</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treatment A:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Open surgery</td>
<td>78% (273/350)</td>
<td><strong>93% (81/87)</strong></td>
<td><strong>73% (192/263)</strong></td>
</tr>
<tr>
<td>Treatment B:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Percutaneous nephrolithotomy</td>
<td><strong>83% (289/350)</strong></td>
<td>87% (234/270)</td>
<td>69% (55/80)</td>
</tr>
</tbody>
</table>

**Table 1:** A classic example of Simpson’s paradox. The table reports the success rates of two treatments for kidney stones (Charig et al., 1986, Tables I and II). Although the overall success rate of treatment B seems better, treatment B performs worse than treatment A on both patients with small kidney stones and patients with large kidney stones. See Section 2.3.

For instance, the empirical comparison of certain kidney stone treatments illustrates this difficulty (Charig et al., 1986). Table 2.3 reports the success rates observed on two groups of 350 patients treated with respectively open surgery (treatment A, with 78% success) and percutaneous nephrolithotomy (treatment B, with 83% success). Although treatment B seems more successful, it was more frequently prescribed to patients suffering from small kidney stones, a less serious condition. Did treatment B achieve a high success rate because of its intrinsic qualities or because it was preferentially applied to less severe cases? Further splitting the data according to the size of the kidney stones reverses the conclusion: treatment A now achieves the best success rate for both patients suffering from large kidney stones and patients suffering from small kidney stones. Such an inversion of the conclusion is called Simpson’s paradox (Simpson, 1951).

The stone size in this study is an example of a **confounding variable**, that is an uncontrolled variable whose consequences pollute the effect of the intervention. Doctors knew the size of the kidney stones, chose to treat the healthier patients with the least invasive treatment B, and therefore caused treatment B to appear more effective than it actually was. If we now decide to apply treatment B to all patients irrespective of the stone size, we break the causal path connecting the stone size to the outcome, we eliminate the illusion, and we will experience disappointing results.

When we suspect the existence of a confounding variable, we can split the contingency tables and reach improved conclusions. Unfortunately we cannot fully trust these conclusions unless we are certain to have taken into account all confounding variables. The real problem therefore comes from the confounding variables we do not know.

Randomized experiments arguably provide the only correct solution to this problem (see Stigler, 1992). The idea is to randomly chose whether the patient receives treatment A or treatment B. Because this random choice is independent from all the potential confounding variables, known and unknown, they cannot pollute the observed effect of the treatments (see also Section 4.2). This is why controlled experiments in ad placement (Section 2.2) randomly distribute users between treatment and control groups, and this is also why, in the case of an ad placement engine, we should be somehow concerned by the practical impossibility to randomly distribute both users and advertisers.

**Influencing factor**

The choice of the patients for each group was function of the severity of the pathology.
Learning systems do not work together flawlessly

- Two sub-systems
  - One locating the ads links
  - The other the adds
- That influence each other
  - Each takes into account the clicks
  - Which depends in part from the actions of the other sub-system
  - In addition of other uncontrolled factors (price, user’s queries, ...)

Importance of identifying the causal graph

Thus, is the sky so blue?

Learning systems ... 

1. Require **enormous amounts** of training data

2. Are exclusively focused on **error rates**

3. Do not fully take advantage of **structures**

4. Do not cooperate well
   - Software engineering with adaptive components is **yet to be solved**
Outline

1. What does work

2. Limitations

3. Learning comes with **which guarantees**?
   - Induction: how to win this game?
   - The statistical learning theory
   - A closed case? Not so sure

4. Other paradigms? An **historical perspective**

5. Is there a **paradigmatic change in sight**?

6. Conclusions
Which guarantees?

The **statistical theory** of learning
Supervised induction

- We want to be able to predict the class of unseen examples
Supervised learning

Given a training set

\[ S_m = \{(x_1, y_1), (x_2, y_2), \ldots, (x_i, y_i), \ldots, (x_m, y_m)\} \]

- **Find** an hypothesis \( h \in \mathcal{H} \) such that \( h(x_i) \approx y_i \)

- Hoping that it generalizes well:

\[ \forall x \in \mathcal{X} : \quad h(x) \approx y \]
One example that tells a lot ...

- Examples described using:
  - Number (1 or 2); size (small or large); shape (circle or square); color (red or green)
- They belong either to class ‘+’ or to class ‘-’
One example that tells a lot ...

- Examples described using:
  
  * **Number** (1 or 2); **size** (small or large); **shape** (circle or square); **color** (red or green)

- They belong either to class ‘+’ or to class ‘-’

<table>
<thead>
<tr>
<th>Description</th>
<th>Your prediction</th>
<th>True class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 large red square</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 large green square</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 small red squares</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 large red circles</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 large green circle</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>1 small red circle</td>
<td></td>
<td>+</td>
</tr>
</tbody>
</table>
One example that tells a lot ...

- Examples described using:
  
  *Number* (1 or 2); *size* (small or large); *shape* (circle or square); *color* (red or green)

<table>
<thead>
<tr>
<th>Description</th>
<th>Your prediction</th>
<th>True class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 large red square</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 large green square</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 small red squares</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 large red circles</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 large green circle</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>1 small red circle</td>
<td></td>
<td>+</td>
</tr>
</tbody>
</table>

How many possible functions altogether from X to Y? \(2^4 = 2^{16} = 65,536\)

How many functions do remain after 6 training examples? \(2^{10} = 1024\)
Examples described using:

*Number* (1 or 2); *size* (small or large); *shape* (circle or square); *color* (red or green)

<table>
<thead>
<tr>
<th>Description</th>
<th>Your prediction</th>
<th>True class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 large red square</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 large green square</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 small red squares</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 large red circles</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 large green circle</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>1 small red circle</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>1 small green square</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 small red square</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>2 large green squares</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 small green squares</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 small red circles</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>1 small green circle</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>2 large green circles</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>2 small green circles</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>1 large red circle</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>2 large red squares</td>
<td></td>
<td>?</td>
</tr>
</tbody>
</table>
One example that tells a lot ...

- Examples described using:
  
  *Number* (1 or 2); *size* (small or large); *shape* (circle or square); *color* (red or green)

<table>
<thead>
<tr>
<th>Description</th>
<th>Your prediction</th>
<th>True class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 large red square</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 large green square</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 small red squares</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>2 large red circles</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>1 large green circle</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>1 small red circle</td>
<td></td>
<td>+</td>
</tr>
</tbody>
</table>

How many possible functions with 2 descriptors from $X$ to $Y$? $2^2 = 2^4 = 16$

How many functions do remain after 3 ≠ training examples? $2^1 = 2$
Induction: an impossible game?

• A bias is need

• Types of bias
  – Representation bias (declarative)
  – Research bias (procedural)
Interpreting – completion of percepts
Interpreting – completion of percepts
Induction and its illusions
Clustering

Original unclustered data
Clustering

Original unclustered data

Clustered data
The perceptron

- Rosenblatt (1958-1962)
The perceptron

- Rosenblatt (1958-1962)

\[
\sigma(i) = \sum_{j=0}^{d} w_{ji} x^{(j)}
\]

\[
y_i = \text{sign}\left\{ g\left( \sum_{j=0}^{d} w_{ji} x^{(j)} \right) \right\}
\]
The perceptron: a linear discriminant
The perceptron learning rule

- Adjustments of the weight $W_i$

Principle (*Perceptron’s rule*): learn only in case of prediction error

---

**Algorithm 1:** The perceptron learning algorithm

**Data:** A training sample: $S_m = \{(x_i, y_i)\}_{1 \leq i \leq m}$

**Result:** A weight vector $w$

while not convergence do

  if the randomly drawn $x_i$ is st. $\text{sign}(w \cdot x_i) = y_i$ then
  do nothing

  else

  $w(t + 1) = w(t) + \eta x_i y_i$

Randomly select next training example $x_i$
The perceptron

NO reasoning !!!
Some remarkable properties!!

- **Convergence** in a finite number of steps
  - Independently of the **number** of examples
  - Independently of the **distribution** of the examples
  - Independently of the **dimension** of the input space

If there exists a linear separator of the training examples
The statistical theory of learning
Guarantees on generalization ??

- Theorems about the performance with respect to the training set

- We want guarantees about future examples
**Statistical study** for $|\mathcal{H}|$ hypotheses

It leads to:

$$\forall h \in \mathcal{H}, \forall \delta \leq 1 : \quad P^m \left[ R(h) \leq \hat{R}(h) + \frac{\log |\mathcal{H}| + \log \frac{1}{\delta}}{m} \right] > 1 - \delta$$

The **Empirical Risk Minimization principle**

is sound **only if** there exists a limit (a bias) on the expressivity of $\mathcal{H}$

The **size $m$ of the training set** must be large enough w.r.t. to capacity of $\mathcal{H}$
Bounds on the difference between the true risk and the empirical risk

- $\mathcal{H}$ finite, realizable case

$$\forall h \in \mathcal{H}, \forall \delta \leq 1 : \quad P^m \left[ R(h) \leq \hat{R}(h) + \frac{\log |\mathcal{H}| + \log \frac{1}{\delta}}{m} \right] > 1 - \delta$$

- $\mathcal{H}$ finite, non realizable case

$$\forall h \in \mathcal{H}, \forall \delta \leq 1 : \quad P^m \left[ R(h) \leq R(h) + \sqrt{\frac{\log |\mathcal{H}| + \log \frac{1}{\delta}}{2m}} \right] > 1 - \delta$$
Statistical theory of learning as a theory of justification

Use of the ERM principle (fitting the data) is justified as long as the expressiveness (or capacity) of $\mathcal{H}$ is controlled (and limited)

\[
\forall h \in \mathcal{H}, \forall \delta \leq 1 : \quad P^m \left[ R(h) \leq \hat{R}(h) + R_S(\mathcal{H}) + 3\sqrt{\frac{\log \frac{2}{\delta}}{2m}} \right] > 1 - \delta
\]
From a theory of justification
to THE recipe for
inventing algorithms

A powerful paradigm
HOW TO ... devise learning algorithms

1. Define an appropriate **regularized** (inductive) **criterion**
   1. Translate the cost of errors of prediction in the domain into a **loss function**
   2. Define a **regularization term** that expresses **assumptions about the underlying regularities of the world**
   3. If possible, make the resulting **optimization** problem a **convex** one

   \[
   h_{opt} = \underset{h \in \mathcal{H}}{\text{ArgMin}} \left[ \frac{1}{m} \sum_{i=1}^{m} l(h(x_i), y_i) + \lambda \text{reg}(\mathcal{H}) \right]
   \]

   \[\begin{align*}
   &\text{empirical risk} \\
   &\text{bias on the world}
   \end{align*}\]

2. Use or develop an **efficient optimization solver**
Learning **sparse linear** approximator

- The **hypothesis** is of the form \( h(x) = w \cdot x \)
- **A priori assumption**: few non zero coefficients

**Ridge regression**

\[
\begin{align*}
    w_{\text{ridge}}^* &= \text{Argmin}_w \left\{ \sum_{i=1}^{m} (y_i - w x_i)^2 + \lambda \|w\|_2^2 \right\}
\end{align*}
\]

**Lasso regression**

\[
\begin{align*}
    w_{\text{lasso}}^* &= \text{Argmin}_w \left\{ \sum_{i=1}^{m} (y_i - w x_i)^2 + \lambda \|w\|_1 \right\}
\end{align*}
\]
3.3 du chapitre [3] Ainsi, étant donnés un échantillon source étiqueté \( S = \{(x^s_i, y^s_i)\}_{i=1}^m \) constitué de \( m \) exemples i.i.d. selon \( P_S \) et un échantillon cible non étiqueté \( T = \{(x^t_i)\}_{i=1}^m \) composé de \( m \) exemples i.i.d. selon \( D_T \), en posant \( S_u = \{x^s_i\}_{i=1}^m \) l'échantillon \( S \) privé de ses étiquettes, on veut minimiser :

\[
\min_{\rho_w} c m \ R_S(G_{\rho_w}) + a m \ \text{dis}_{\rho_w}(S_u, T_u) + \text{KL}(\rho_w \parallel \pi_0),
\]

où \( \text{dis}_{\rho_w}(S_u, T_u) = \left| \mathbb{E}_{(h, h') \sim \rho_w} R_{S_u}(h, h') - \mathbb{E}_{(h, h') \sim \rho_w} R_{T_u}(h, h') \right| \) est le désaccord empirique entre \( S_u \) et \( T_u \) spécialisé à une distribution \( \rho_w \) sur l'espace \( \mathcal{H} \) des classifiants linéaires considéré. Les réels \( a > 0 \) et \( c > 0 \) sont des hyperparamètres de l'algorithme. Notons que les constantes \( A \) et \( C \) du théorème 7.7 peuvent être retrouvées à partir de n'importe quelle valeur de \( a \) et \( c \). Étant donnée la fonction \( \ell_{\text{dis}}(x) = 2 \ell_{\text{Err}}(x) \ell_{\text{Err}}(-x) \) (illustrée sur la figure 7.1), pour toute distribution \( D \) sur \( X \), on a :

\[
\mathbb{E}_{(h, h') \sim \rho_w} R_D(h, h') = \mathbb{E}_{x \sim D} \mathbb{E}_{(h, h') \sim \rho_w} I[h(x) \neq h'(x)] = 2 \mathbb{E}_{x \sim D} \mathbb{E}_{h \sim \rho_w} I[h(x) = 1] I[h'(x) = -1] = 2 \mathbb{E}_{x \sim D} \ell_{\text{Err}} \left( \frac{\langle w, x \rangle}{\|x\|} \right) \ell_{\text{Err}} \left( -\frac{\langle w, x \rangle}{\|x\|} \right) = 2 \mathbb{E}_{x \sim D} \ell_{\text{dis}} \left( \frac{\langle w, x \rangle}{\|x\|} \right).
\]

Ainsi, trouver la solution optimale de l’équation (7.5) revient à chercher le vecteur \( w \) qui minimise :

\[
c \sum_{i=1}^m \ell_{\text{Err}} \left( y^t_i \frac{\langle w, x_i^t \rangle}{\|x_i^t\|} \right) + a \sum_{i=1}^m \left[ \ell_{\text{dis}} \left( \frac{\langle w, x_i^s \rangle}{\|x_i^s\|} \right) - \ell_{\text{dis}} \left( \frac{\langle w, x_i^t \rangle}{\|x_i^t\|} \right) \right] + \frac{\|w\|^2}{2}.
\]

L'équation précédente est fortement non convexe. Afin de rendre sa résolution plus facilement contrôlable, nous remplacons la fonction \( \ell_{\text{Err}}(\cdot) \) par sa relaxation convexe \( \ell_{\text{Err}_\alpha}(\cdot) \) (comme pour PBGD3 et illustrée sur la figure 7.1). L'optimisation se réalise ensuite par une descente de gradient. Le gradient de l'équation 7.6 étant :
A very alluring framework

1. Based on a justification theory
   - Bounds on the generalization error can be claimed
     (very important for having paper accepted)
   - Valid for the worst case: against any possible distribution of the data

2. Seemingly very benign assumptions on the world
   - Data (and future questions) supposedly i.i.d.
   - $f \in H$ or $f \not\in H$

3. Provides a recipe to produce learning algorithms
   - Very generic applicability: minimization of a regularized empirical risk
   - Learning = optimization
A lot of “Lamppost theorems”

Theorems that guarantee that:

– **If** the world obeys **my a priori assumptions**

– **Then** the learning algorithm will end up with a good hypothesis (closed to the “real” one)

– **Otherwise** learning can lead to very bad hypotheses

  *(e.g. *If the world is not sparse*)
« Deep learning »
as THE universal solution (2006– ...)

"The paper focuses on a subject that might be of limited importance at ICML, *given the current trend towards neural networks*."
Artificial Neural Networks

- With numerous hidden layers (possibly > 100s)
- And a very large number of parameters (~ $10^7$ – $10^8$ parameters)
- Learn hierarchical and compositional representations

object models

object parts (combination of edges)

edges
GoogleNet

- A mécano of neural networks
BUT ... does deep learning bring big trouble (for the theory of induction)?
Troubling findings

A paper

  “Understanding deep learning requires rethinking generalization”

Extensive experiments on the classification of images

- The AlexNet (> 1,000,000 parameters) + 2 other architectures

- The CIFAR-10 data set:
  - 60,000 images categorized in 10 classes (50,000 for training and 10,000 for testing)
  - Images: 32x32 pixels in 3 color channels
Troubling findings

Experiments

1. **Original dataset without modification**
   - Results?
     - **Training** accuracy = 100% ; **Test** accuracy = 89%
     - Speed of convergence ~ 5,000 steps
Troubling findings

Experiments

1. Original dataset without modification
   - Results?
     - Training accuracy = 100% ; Test accuracy = 89%
     - Speed of convergence ~ 5,000 steps

Expected behavior if the capacity of the hypothesis space is limited

i.e. the system cannot fit any (arbitrary) training data

\[ \forall h \in \mathcal{H}, \forall \delta \leq 1 : \quad P^m \left[ R(h) \leq \hat{R}(h) + 2 \Rad_m(\mathcal{H}) + 3 \sqrt{\frac{\ln(2/\delta)}{m}} \right] > 1 - \delta \]
Troubling findings

Experiments

1. **Original dataset without modification**
   - Results?
     - **Training** accuracy = 100% ; **Test** accuracy = 89%
     - Speed of convergence $\sim$ 5,000 steps

2. **Random labels**
   - **Training** accuracy = 100% !?? ; **Test** accuracy = 9.8%
   - Speed of convergence = similar behavior $\sim$ 10,000 steps
Troubling findings

Experiments

1. Original dataset without modification
   - Results?
     - Training accuracy = 100% ; Test accuracy = 89%
     - Speed of convergence ~ 5,000 steps

2. Random labels
   - Training accuracy = 100% !?? ; Test accuracy = 9.8%
   - Speed of convergence = similar behavior (~ 10,000 steps)

3. Random pixels
   - Training accuracy = 100% !?? ; Test accuracy ~ 10%
   - Speed of convergence = similar behavior (~ 10,000 steps)

Now, we are in trouble!!
Troubling findings

- Deep NNs can accommodate ANY training set

\[
\forall h \in \mathcal{H}, \forall \delta \leq 1 : \quad P^m \left[ R(h) \leq \hat{R}(h) + 2 \widehat{\text{Rad}}_m(\mathcal{H}) + 3 \sqrt{\frac{\ln(2/\delta)}{m}} \right] > 1 - \delta
\]

Can grow without limit!!

But then,

why are deep NNs so good on image classification tasks?
Adversarial learning

Illustration

Adversarial input can fool a machine-learning algorithm into misperceiving images.
What makes a good visual explanation?

- Propagation
- High-resolution outputs
- Interpretable without altering their architecture, thus performance for more transparency into the working of the model.

Class Activation Mapping (CAM) for identifying class-discriminative modules for uninterpretable ones that achieve greater performance. In contrast, localization approaches like CAM or our proposed method Gradient-weighted Class Activation Mapping (Grad-CAM) are over 200-layers deep and are often good at localizing discriminative regions used by a restricted class of image classification. The discriminative ability of Grad-CAM significantly reduces while being orders of magnitude cheaper to compute.

In contrast, important regions of the image which correspond to the 'tiger cat' class (top) and 'boxer' (dog) class (bottom). This is shown in Figures 1a, 1b, 1c, 1d, 1e, 1f, 1g, 1h, 1i, 1j, 1k, 1l.

As such, deep models are beginning to explore the best of both worlds. We show that it is possible to fuse existing pixel-space gradient visualizations, e.g., of current CNNs (Section 6.1) and Deconvolution (Section 4.1) – a 'good' visual explanation from multi-modal inputs (Section 3.1) is important for predicting that particular variety of cat. As a result, important regions of the image which correspond to the 'tiger cat' class (top) and 'boxer' (dog) class (bottom).

To summarize, our contributions are as follows:

1. We visualize ResNets and deep neural networks as we encounter layers with different output dimensionality.
2. We propose Grad-CAM, a class-discriminative localization technique for existing top-performing classification networks that are both high-resolution and class-discriminative. We apply Grad-CAM to existing top-performing classification networks and highlight fine-grained details in the image, but are not often good at localizing discriminative regions used by a restricted class of image classification. The discriminative ability of Grad-CAM significantly reduces while being orders of magnitude cheaper to compute.

In order to combine the best of both worlds, we show that it is possible to fuse existing pixel-space gradient visualizations, e.g., of current CNNs (Section 6.1) and Deconvolution (Section 4.1) – a 'good' visual explanation from multi-modal inputs (Section 3.1) is important for predicting that particular variety of cat. As a result, important regions of the image which correspond to the 'tiger cat' class (top) and 'boxer' (dog) class (bottom). This is shown in Figures 1a, 1b, 1c, 1d, 1e, 1f, 1g, 1h, 1i, 1j, 1k, 1l.

As such, deep models are beginning to explore the best of both worlds. We show that it is possible to fuse existing pixel-space gradient visualizations, e.g., of current CNNs (Section 6.1) and Deconvolution (Section 4.1) – a 'good' visual explanation from multi-modal inputs (Section 3.1) is important for predicting that particular variety of cat. As a result, important regions of the image which correspond to the 'tiger cat' class (top) and 'boxer' (dog) class (bottom). This is shown in Figures 1a, 1b, 1c, 1d, 1e, 1f, 1g, 1h, 1i, 1j, 1k, 1l.

Explanations and deep neural networks

Evaluation protocol: comparison between explanations

- Which robot do you trust most?

54 subjects on Amazon Turk -> robot B evaluated 1.27 (between -2 et +2)

Optical illusions: how to explain them?

Car in a swimming pool

• ... or no car ...?

Is this less of a car because the context is wrong?

[Léon Bottou (ICML-2015, invited talk) « Two big challenges in Machine Learning »]
Assessment

1. A theory
   - for stationary environments and i.i.d. data and queries!!
   - focused on the expectation of the cost of errors
     - Prior knowledge must be encoded in the cost
   - that can produce learning algorithms when combined with optimization techniques

2. Deep NNs
   - depart from this framework
     - Demand at least a reworking of the theory
     - Prior knowledge encoded in the architecture
   - Still
     - require enormous amount of data
     - Focused on error rates
     - Based on correlations
Outline

1. What does work

2. Limitations

3. Learning comes with which guarantees?
   - Induction: how to win this game?
   - The statistical learning theory
   - A closed case? Not so sure

4. Other paradigms? An historical perspective

5. Is there a paradigmatic change in sight?

6. Conclusions
Are there other paradigms?

An historical perspective on ML
Learning ...

... as a means to improve the efficiency of a problem solver
E.g. The PRODIGY system


PRODIGY: An Integrated Architecture for Planning and Learning
Jaime Carbonell, Oren Etzioni*, Yolanda Gil, Robert Joseph
Craig Knoblock, Steve Minton†, and Manuela Veloso

PRODIGY’s basic reasoning engine is a general-purpose problem solver and planner [10] that searches for sequences of operators (i.e., plans) to accomplish a set of goals from a specified initial state description. Search in PRODIGY is guided by a set of control rules that apply at each decision point.

PRODIGY’s reliance on explicit control rules, which can be learned for specific domains, distinguishes it from most domain independent problem solvers. Instead of using a least-commitment search strategy, for example, PRODIGY expects that any important decisions will be guided by the presence of appropriate control knowledge. If no control rules are relevant to a decision, then PRODIGY makes a quick, arbitrary choice. If in fact the wrong choice is made, and costly backtracking proves necessary, an attempt will be made to learn the control knowledge that must be missing.
Illustration: LEX (Tom Mitchell)

Génération de problèmes

Résolution de problèmes

Critique

Généralisation

Calculer la primitive de :
\[ \int 3x \cos(x) \, dx \]

Espace des versions pour l'utilisation de l'opérateur OP2 :

\[ S = \{ \int 3x \cos(x) \, dx \rightarrow \text{Appliquer OP2 avec } u = 3x, dv = \cos(x) \, dx \} \]

\[ G = \{ \int f_1(x) f_2(x) \, dx \rightarrow \text{Appliquer OP2 avec } u = f_1(x), dv = f_2(x) \, dx \} \]
Calculer la primitive de :
\[ \int 3x \cos(x) \, dx \]

\[ \int 3x \cos(x) \, dx = 3 \sin(x) - \int 3 \sin(x) \, dx \]

\[ \int 3x \cos(x) \, dx = 3 \sin(x) - 3 \int x \sin(x) \, dx + C \]

\[ 3 \sin(x) - 3 \int x \sin(x) \, dx \]

\[ \int 3x \cos(x) \, dx \quad \text{OP2 avec :} \quad u = 3x \]
\[ \quad \quad \quad \quad \quad \quad dv = \cos(x) \, dx \]

\[ 3 \sin(x) - \int 3x \sin(x) \, dx \quad \text{OP1} \]

\[ 3 \sin(x) - 3 \int x \sin(x) \, dx \quad \text{OP5} \]
Calculer la primitive de : 
\[ \int 3x \cos(x) \, dx \]

Un des exemples positifs proposés :
\[ \int 3x \cos(x) \, dx \rightarrow \text{Appliquer OP2 avec :} \]
\[ u = 3x \]
\[ dv = \cos(x) \, dx \]
Illustration: LEX (Tom Mitchell)

Calculer la primitive de :
\[ \int 3x \cos(x) \, dx \]

Espace des versions pour l'utilisation de l'opérateur \( \text{OP2} \):

\[ S = \{ \int 3x \cos(x) \, dx \rightarrow \text{Appliquer \ OP2} \] avec : \( u = 3x \)
\( dv = \cos(x) \, dx \} \]
\[ G = \{ \int f1(x) f2(x) \, dx \rightarrow \text{Appliquer \ OP2} \] avec : \( u = f1(x) \)
\( dv = f2(x) \, dx \} \]

Un des exemples positifs proposés :
\[ \int 3x \cos(x) \, dx \rightarrow \text{Appliquer \ OP2} \] avec :
\( u = 3x \)
\( dv = \cos(x) \, dx \)
Learning from one example

Explanation-Based Learning

1. From a single example

2. Try to prove the “fork”

3. Generalize
Ex : **learn the concept**  \( \text{stackable}(\text{Object1}, \text{Object2}) \)

- **Domain theory** :
  
  (T1) : \( \text{weight}(X, W) :- \text{volume}(X, V), \text{density}(X, D), W = V \cdot D \).
  
  (T2) : \( \text{weight}(X, 50) :- \text{is}_a(X, \text{table}) \).
  
  (T3) : \( \text{lighter}_\text{than}(X, Y) :- \text{weight}(X, W_1), \text{weight}(X, W_2), W_1 < W_2 \).

- **Operationality constraint** :
  
  - Concept should be expressible using \( \text{volume}, \text{density}, \text{color}, \ldots \)

- **Positive example** (**solution**) :
  
  \begin{align*}
  & \text{on} (\text{obj1, obj2}). \\
  & \text{is}_a(\text{object1, box}). \\
  & \text{is}_a(\text{object2, table}). \\
  & \text{color}(\text{object1, red}). \\
  & \text{color}(\text{object2, blue}). \\
  & \text{made}_\text{of} (\text{object2, wood}). \\
  & \text{volume}(\text{object1, 1}). \\
  & \text{volume}(\text{object2, 0.1}). \\
  & \text{owner}(\text{object1, frederic}). \\
  & \text{density}(\text{object1, 0.3}). \\
  & \text{Made}_\text{of}(\text{object1, cardboard}). \\
  & \text{owner}(\text{object2, marc}).
  \end{align*}
Explanation-Based Learning

Generalized search tree resulting from regression of the target concept in the proof tree by computing at each step the most general literals allowing this step.
Explanation-Based Learning

• Induction **from a single example**
  – ... plus a strong domain theory

• Based on
  – **Logic-based** knowledge representation
  – **Reasoning Operators** (deduction, goal regression in a proof tree, ...)

*Now used in SAT “solvers”*
Explanation-Based Learning

• What was the **aim** of learning?
• What was a **good theory/ method** of learning?
Explanation-Based Learning

• What was the aim of learning?
• What was a good method of learning?

1. Method improving the problem solving performances
   – [Steve Minton (1990) « Quantitative results concerning the utility of Explanation-Based Learning »]

2. Method that simulates the performances (and limits) of a natural cognitive agent (human or animal)
Learning and reasoning

Papers like


But

No measure of generalization performance independent of the problem-solver

Difficulties to scale up and to face noisy data

... when data started to pour down
Outline

1. What does work

2. Limitations

3. Learning comes with which guarantees?
   - Induction: how to win this game?
   - The statistical learning theory
   - A closed case? Not so sure

4. Other paradigms? An historical perspective

5. Is there a paradigmatic change in sight?

6. Conclusions
New learning scenarios

=> Is there a paradigmatic change in sight?
Identification of causal relationships

- In images
- With unsupervised learning!!

Results

- Top anticausal features have higher object scores for all twenty categories.
- The probability that this happens for all 20 classes out of chance is $2^{-20} \approx 10^{-6}$.

The AlphaGo case

- Plays like an « alien »
- Out of the blue moves
- A revolution in the theory of the game
- Excitement in the go schools
The AlphaGo case: understanding

Fan Hui, Gu Li, Zhou Ruyang (very strong Go players) turn to the activity of analyzing the games played by AlphaGo

• Kind on of exegis. Explanations a posteriori
• Necessary for
  – communication
  – Teaching

And event AlphaGo might err
Explanations in Expert Systems

MYCIN : L'infection du malade a-t-elle été contractée à l'hôpital ?
Utilisateur : pourquoi ?
TEIRESIAS : [pourquoi est-ce important de savoir cela ?]

Cela aidera à déterminer la catégorie à laquelle appartient l'organisme responsable de l'infection :
- il a déjà été établi que 1° l'organisme est à Gram négatif, 2° c'est un bâtonnet, 3° il se développe à l'air ou sans air,
- donc si l'infection par cet organisme n'a pas été contractée lors du séjour à l'hôpital alors il est presque certain (0,8) que l'organisme est de la catégorie des enterobacteriaceae; de plus, il se peut (0,1 ; règle 37) que ce soit la pseudomonia aeruginosa.

Utilisateur : pourquoi n'avez-vous pas envisagé la possibilité que l'organisme soit un streptocoque ?
TEIRESIAS : la règle 33 aurait du être utilisée pour identifier l'organisme comme un streptocoque, mais la seconde clause "la morphologie est de type coque" était déjà connue comme fausse et la règle 33 n'a jamais été essayée.
• Quel est l’âge du patient ?
• ** Pourquoi
• Pour déterminer si on peut prescrire de la tétracycline
  – Si le patient a moins de 8 ans
  – Alors on ne peut pas prescrire de tétracycline
  – [Règle 122]
• ** Pourquoi ?
• ...

Pourquoi ne faut-il pas prescrire de tétracycline à un enfant de moins de 8 ans ?
**Explanations in Expert Systems**

*Why should we not prescribe tetracycline to a child under the age of 8?*

**Expert justifications**

- Drug depot on developing bones
- Definitive *blackening* of the teeth
- *Socially unwanted* coloration
- *Do not administer* tetracycline to children under the age of

Notion of undesirable *side effects*

*Causality* relationships
Transfer Learning

**Definition** [Pan, TL-IJCAI’13 tutorial]

- Ability of a system to **recognize** and **apply** knowledge and skills learned in **previous domains/tasks** to **novel domains/tasks**

**Example**

- We have **labeled images** (person / no person) from a **web corpus**
- Novel task: **is there a person** in unlabeled images from a **video corpus**?

```plaintext
Person  no Person

Web corpus       Video corpus

Is there a Person?
```

(A. Cornuéjols)
Transfert learning: questions

• What can be the basis of transfer learning?

  How to translate formally:

  “the target domain is like the source domain”?

• What determine a good transfer?
  – A “good source”?
  – A high “similarity” between source and target?

• What formal guarantees can we have on the transferred hypothesis?
Transfer and analogy

Why should ‘a a b a b c d’ be any better than ‘a b d’?
Transfer and sequence effects

• t

iCube speech - 2018 « What is a good ML algorithm? » (A. Cornuéjols)
Long-life learning

- Learning organized in a sequence of tasks
  - Very far from the i.i.d. scenario

- Learning will be affected by the history of the system

- We need a theory of the dynamics of learning
  1. Which sequence effects can we expect?
  2. How to best organize the curriculum of a learning system?
Conclusions
The current situation

- Inductive learning *needs biases*
  - No objective bias-free results

- The **theory**
  - Is focused entirely on the **error rate**
  - Assumes stationary environment and random inputs (i.i.d.)
  - Requires large enough data sets w.r.t. to the capacity of $\mathcal{H}$

- We do not understand well deep neural networks

- Correlations ≠ structures, semantics, causation
We start to pay attention to new demands

1. The need for explanations
   - Structures
   - **Causal** reasoning
   - No more only error rate
We start to pay attention to new demands

1. The need for explanations
   - Structures
   - **Causal** reasoning
   - No more only error rate

2. The need for transfer learning
   - **What** should be transferred?
   - **Conditions** for positive / negative transfer?
We start to pay attention to new demands

1. The need for explanations
   - Structures
   - **Causal** reasoning
   - No more only error rate

2. The need for **transfer learning**
   - **What** should be transferred?
   - **Conditions** for positive / negative transfer?

3. Scenarios **away from the i.i.d. assumption**
   - Online learning / **changing environments**
   - **Curriculum** learning
   - Long-life learning
Conclusions: “new” scenarios

• **Limited** data sources
  
  – We often learn from (very) few examples

• The past **history** of learning affects learning: **Education**
  
  – Sequence effects

• We learn in order to **build “theories”**
  
  – All the time: small and large theories

For instance, what would you like to ask?