

Initiation à l'IA explicable

Céline Hudelot, Wassila Ouerdane et Jean-Philippe Poli

Plate-Forme Intelligence Artificielle (PFIA) Strasbourg – 06 Juillet 2023



Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Sommaire

Introduction

Constat et motivations

Qu'est ce que l'IA eXplicable ?

Terminologie et définitions

Principales approches

Modèles explicables par conception (transparents)

Modèles de régression

Generalised Additive Models (GAM)

Modèles à base d'arbres

Explications post-hoc

Méthodes indépendantes du modèle

Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Constat

Avènement du Deep Learning

- ▶ Abondance de données, performances des nouveaux processeurs (GPU)
- ▶ Le DL a relancé l'intérêt pour l'IA en explosant le nombre d'applications possibles
- ▶ Les résultats défient les compétences humaines
- ▶ Les humains se sentent de plus en plus concernés

Constat

Avènement du Deep Learning

- ▶ Abondance de données, performances des nouveaux processeurs (GPU)
- ▶ Le DL a relancé l'intérêt pour l'IA en explosant le nombre d'applications possibles
- ▶ Les résultats défient les compétences humaines
- ▶ Les humains se sentent de plus en plus concernés

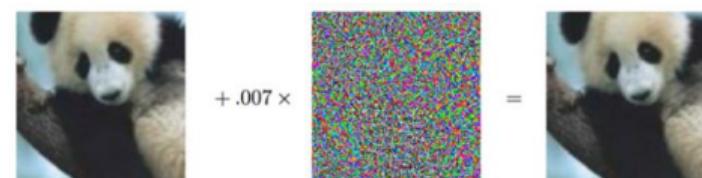
Facebook, Google, Apple... Pour les Gafam, nos conversations valent de l'or



Publié le : 14/08/2019 - 17:59 Modifié le : 14/08/2019 - 18:02



Ces dernières années, les données sonores sont devenues la nouvelle cible des Gafam. ©TR, Josh Edelson, Loïc Venance, Emmanuel Dunand, AFP



“panda”

57.7% confidence

noise

“gibbon”

99.3% confidence

Image credits [Explaining and Harnessing Adversarial Examples](#)

Constat : des IAs persuasives

Conversation avec chat GPT (23/03/2023)

Could you convert 1 V/micron to MV/cm ?

Yes, I can convert 1 V/micron to MV/cm.

1 V/micron is equivalent to 1000 MV/meter (since $1 \text{ micron} = 10^{-6} \text{ meters}$), and 1 meter is equal to 100 centimeters. Therefore :

$1 \text{ V/micron} = 1000 \text{ MV/meter} = 1000 \text{ MV}/10^2 \text{ cm}$ (since $1 \text{ meter} = 10^2 \text{ cm}$) = 10 MV/cm

Therefore, 1 V/micron is equal to 10 MV/cm.

Constat : des IAs persuasives

Conversation avec chat GPT (23/03/2023)

Could you convert 1 V/micron to MV/cm ?

Yes, I can convert 1 V/micron to MV/cm.

1 V/micron is equivalent to 1000 MV/meter (since $1 \text{ micron} = 10^{-6} \text{ meters}$), and 1 meter is equal to 100 centimeters. Therefore :

$1 \text{ V/micron} = 1000 \text{ MV/meter} = 1000 \text{ MV}/10^2 \text{ cm}$ (since $1 \text{ meter} = 10^2 \text{ cm}$) = 10 MV/cm

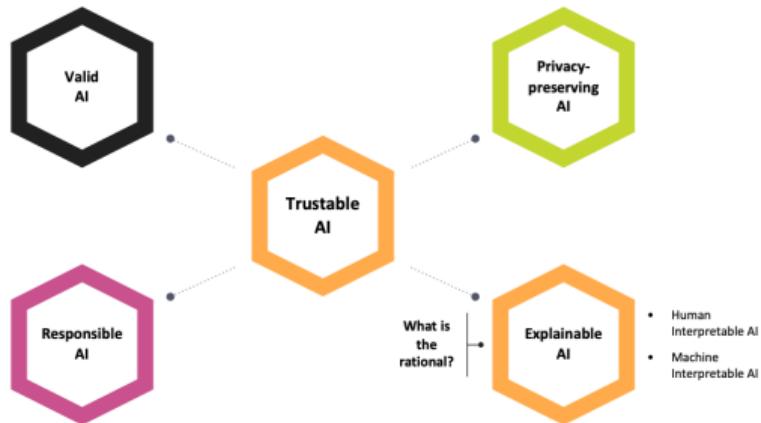
Therefore, 1 V/micron is equal to 10 MV/cm.

Seems it is wrong !

Vers une IA de confiance

En quoi avoir confiance ?

- ▶ en sa **validité** : preuve d'algorithmes et de code, tests, ...
- ▶ en sa **responsabilité** : éthique, frugalité, ...
- ▶ en ses **données** : respect de la vie privée, représentativité, équilibre, ...
- ▶ en ses **modèles** : compréhension, déterminisme, ...
- ▶ en ses **décisions** : restitution, compréhensibilité, ...



IA de confiance : réglementation de l'IA

Exemples :

- ▶ SR 11-7 : Guidance on Model Risk Management (USA, 2011)

Robustesse des modèles

<https://www.federalreserve.gov/supervisionreg/srletters/sr1107.htm>

- ▶ RGPD (Europe, 2016)

Mesures pour la protection des données privées

<https://eur-lex.europa.eu/legal-content/FR/TXT/HTML/?uri=CELEX:32016R0679>

- ▶ California Consumer Privacy Act (Californie, 2018)

Gestion des données personnelles en entreprise

<https://www.oag.ca.gov/privacy/ccpa>

- ▶ Algorithmic Accountability Act (USA, 2019)

Les entreprises doivent évaluer le risque des décisions prises automatiquement vis-à-vis de la vie privée et de la sécurité de leurs clients

<https://www.congress.gov/bill/116th-congress/house-bill/2231/text>

- ▶ ...



Sommaire

Introduction

Constat et motivations

Qu'est ce que l'IA eXplicable ?

Terminologie et définitions

Principales approches

Modèles explicables par conception (transparents)

Modèles de régression

Generalised Additive Models (GAM)

Modèles à base d'arbres

Explications post-hoc

Méthodes indépendantes du modèle

Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

XAI : eXplainable Artificial Intelligence

Appel à projet

- ▶ En 2016, la DARPA lance un nouveau programme pour financer la recherche en IA, avec une contrainte d'explicabilité,
- ▶ Son auteur, David Gunning, appelle ce champs XAI

XAI : eXplainable Artificial Intelligence

Appel à projet

- ▶ En 2016, la DARPA lance un nouveau programme pour financer la recherche en IA, avec une contrainte d'explicabilité,
- ▶ Son auteur, David Gunning, appelle ce champs XAI

Définition

Le but d'un système XAI est de rendre son comportement plus intelligible pour les humains en leur fournissant des explications. Un tel système doit être capable :

- ▶ d'expliquer ses capacités et sa compréhension de l'environnement,
- ▶ d'expliquer ce qu'il a fait, ce qu'il fait, ce qu'il fera ensuite,
- ▶ de mettre en évidence les informations pertinentes qu'il utilise.

XAI : eXplainable Artificial Intelligence

Appel à projet

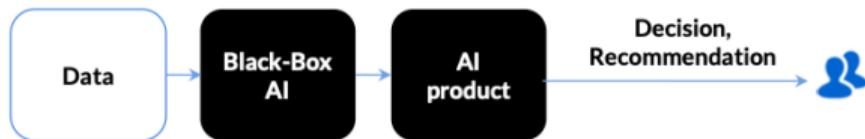
- ▶ En 2016, la DARPA lance un nouveau programme pour financer la recherche en IA, avec une contrainte d'explicabilité,
- ▶ Son auteur, David Gunning, appelle ce champs XAI

Autre définition (Arrieta et al, 2020)

Une XIA est un système qui fournit les détails ou les raisons afin de rendre son fonctionnement clair ou facile à comprendre.

XAI : eXplainable Artificial Intelligence

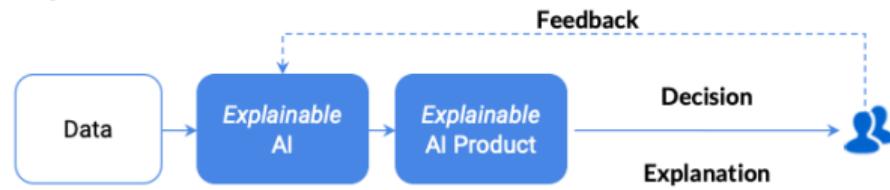
Black Box AI



Confusion with Today's AI Black Box

- Why did you do that?
- Why did you not do that?
- When do you succeed or fail?
- How do I correct an error?

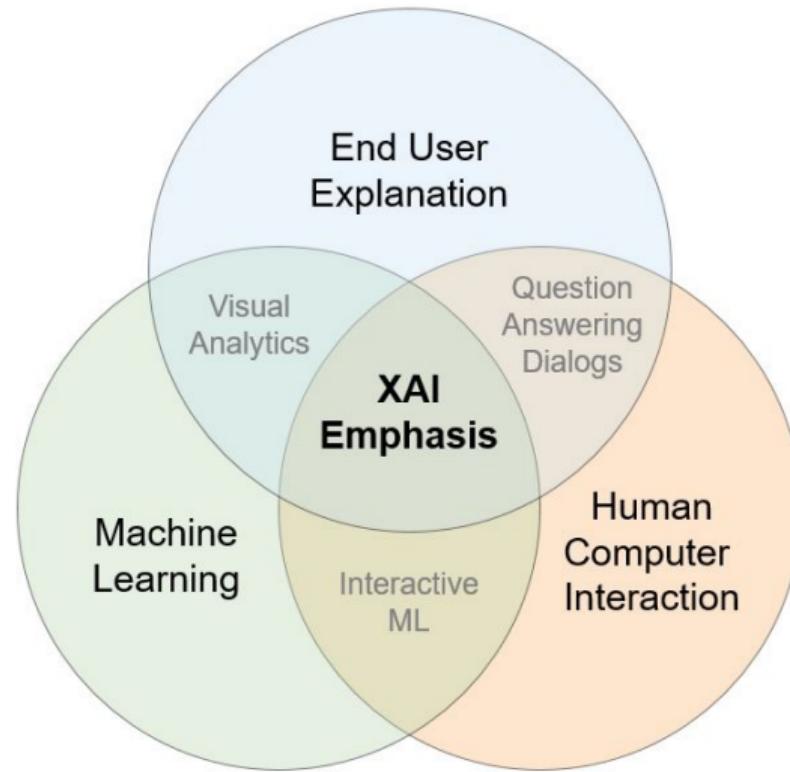
Explainable AI



Clear & Transparent Predictions

- I understand why
- I understand why not
- I know why you succeed or fail
- I understand, so I trust you

XAI : un domaine de recherche multi-disciplinaire



XAI : eXplainable Artificial Intelligence

Plusieurs questions ?

- ▶ Est-ce qu'une IA doit vraiment donner des explications ?
- ▶ Qu'est-ce qu'une explication ?
- ▶ Dans les modèles existants, y'a-t-il des candidats ?
- ▶ Comment évaluer une XAI ?

Une IA doit-elle s'expliquer ?



Yann LeCun @ylecun · Feb 5, 2020

We often hear that AI systems must provide explanations and establish causal relationships, particularly for life-critical applications.

Yes, that can be useful. Or at least reassuring....

1/n



45



372



988



...



Yann LeCun @ylecun · Feb 5, 2020

But sometimes people have accurate models of a phenomenon without any intuitive explanation or causation that provides an accurate picture of the situation. In many cases of physical phenomena, "explanations" contain causal loops where A causes B and B causes A.

2/n



5



14



144



...



Yann LeCun @ylecun · Feb 5, 2020

A good example is how a wing causes lift. The computational fluid dynamics model, based on Navier-Stokes equations, works just fine. But there is no completely-accurate intuitive "explanation" of why airplanes fly.

3/n



Une IA doit-elle s'expliquer ?

Table ronde sur l'explicabilité, DigiHall Days 2019

Pourquoi une IA aurait-elle besoin d'expliquer ses décisions ? Quand vous allez chez le médecin, vous ne lui demandez pas pourquoi il vous prescrit tel ou tel médicament et pourquoi il a posé ce diagnostic.

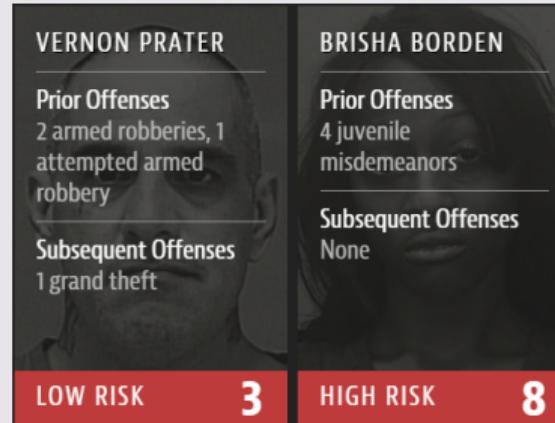
Anonyme

Une IA doit-elle s'expliquer ?

Les boîtes noires posent le problème du biais dans les données.

Machine bias

There's software used across the country to predict future criminals. And it's biased against blacks.



<https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing>

Une IA doit-elle s'expliquer ?

Les boîtes noires posent le problème du biais dans les données.

Lésions de la peau

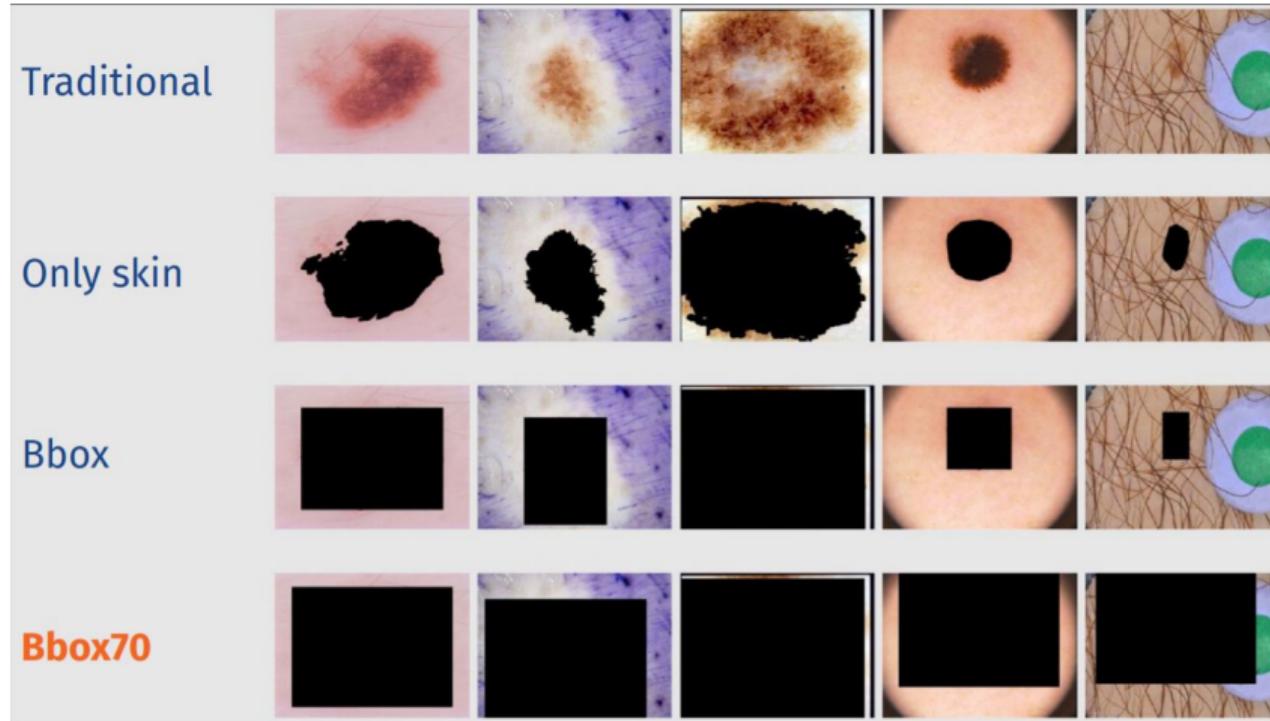
- ▶ La base ISIC est une base d'images dermoscopiques annotées
- ▶ Utilisée dans différents challenges
- ▶ Les réseaux de neurones profonds ($AUC=71\%$) ont de meilleurs résultats que les dermatologues ($AUC=67\%$)

https://openaccess.thecvf.com/content_CVPRW_2019/papers/ISIC/Bissoto_DeConstructing_Bias_on_Skin_Lesion_Datasets_CVPRW_2019_paper.pdf

https://www.researchgate.net/publication/331287430_Comparing_artificial_intelligence_algorithms_to_157_German_dermatologists_the_melanoma_classification_benchmark

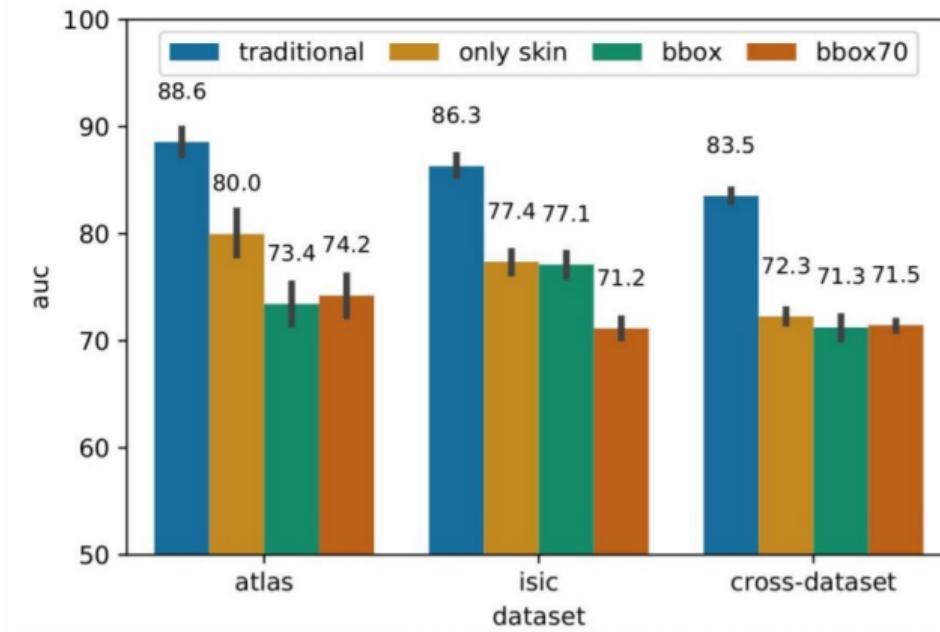
Une IA doit-elle s'expliquer ?

Les boîtes noires posent le problème du biais dans les données.



Une IA doit-elle s'expliquer ?

Les boîtes noires posent le problème du biais dans les données.



Une IA doit-elle s'expliquer ?

Tentative de réponse

La fourniture d'une explication d'une décision est nécessaire :

- ▶ d'un point de vue sociétal pour éviter de renforcer les inégalités dues aux biais
- ▶ d'un point de vue réglementaire pour satisfaire les nouvelles lois (RGPD, CCPA...)
- ▶ pour fournir aux utilisateurs finaux les éléments par lesquels une décision est prise (et ainsi augmenter leur adhésion).

Tous les domaines ne sont pas égaux devant cette nécessité :

- ▶ Médecine
- ▶ Détection de visage (appareils photos/smartphones)
- ▶ Sciences fondamentales
- ▶ Véhicule autonome
- ▶ Lecture des codes postaux
- ▶ ...



Une IA doit-elle s'expliquer ?

Le mot de la fin

[...] current efforts face unprecedented difficulties : contemporary models are more complex and less interpretable than ever; [AI systems are] used for a wider array of tasks, and are more pervasive in everyday life than in the past; and [AI is] increasingly allowed to make (and take) more autonomous decisions (and actions). Justifying these decisions will only become more crucial, and there is little doubt that this field will continue to rise in prominence and produce exciting and much needed work in the future.

(Biran and Cotton, 2017)

Sommaire

Introduction

Constat et motivations

Qu'est ce que l'IA eXplicable ?

Terminologie et définitions

Principales approches

Modèles explicables par conception (transparents)

Modèles de régression

Generalised Additive Models (GAM)

Modèles à base d'arbres

Explications post-hoc

Méthodes indépendantes du modèle

Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Quelques définitions

Le domaine de l'XAI intéresse diverses communautés :

- ▶ AI,
- ▶ ML,
- ▶ IHM,
- ▶ Sciences cognitives,
- ▶ Sciences sociales,
- ▶ etc.

- ▶ Il n'y a actuellement pas de consensus dans les définitions,
- ▶ Les définitions vont changer d'une communauté à l'autre, mais également d'une application à l'autre.

Quelques définitions

Interprétabilité (Doshi-Velez & Kim, 2017)

L'**interprétabilité** est la capacité d'être expliqué ou présenté en des termes compréhensibles par un humain.

Explication (Guidotti et al, 2018)

Une **explication** est une interface entre un preneur de décision et un humain qui est à la fois une représentation fidèle du processus de décision et compréhensible par les humains.

Interprétabilité vs explicabilité en ML

- ▶ Au mieux, on va considérer que :
 - ▶ l'interprétabilité est une caractéristique du modèle,
 - ▶ l'explicabilité nécessite une procédure pour rendre le modèle plus clair.
- ▶ Mais très souvent, explicabilité et interprétabilité sont confondues

Exemples

- ▶ Recherche des features qui sont à l'origine du résultat
- ▶ Cartes de saillance
- ▶ etc.

Autres termes courants

Intelligibilité

Caractéristique d'un modèle à faire comprendre à un humain sa fonction - comment il fonctionne - sans qu'il soit nécessaire d'expliquer sa structure interne ou les moyens algorithmiques par lesquels il traite les données en interne (Montavon et al, 2018).

Compréhensibilité

Capacité d'un modèle à représenter ses connaissances inférées d'une manière compréhensible pour l'homme (Gleicher et al, 2016).

Transparence

Un modèle est dit transparent s'il est compréhensible par lui-même (Lipton, 2017).

Autre définition d'une explication

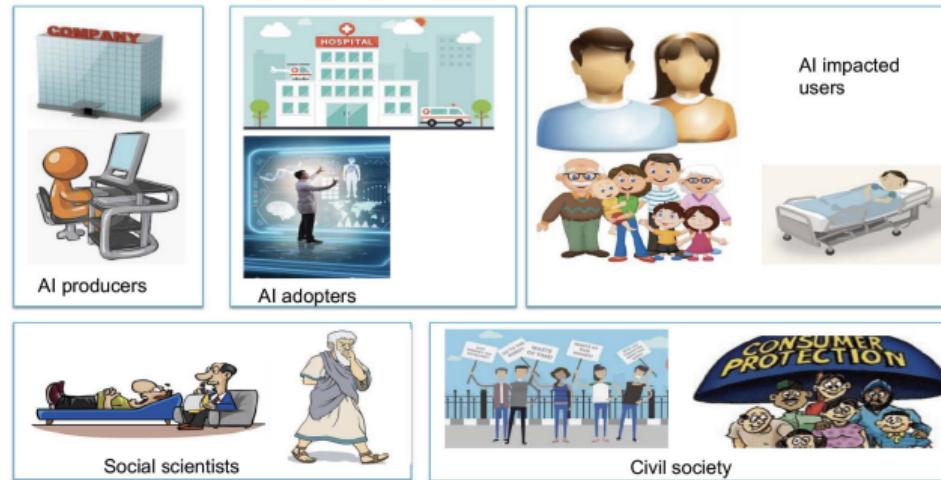
Miller propose une étude approfondie des explications, en s'appuyant sur les sciences sociales.

Explication (Miller, 2017)

Une explication est la conjonction des trois éléments suivants :

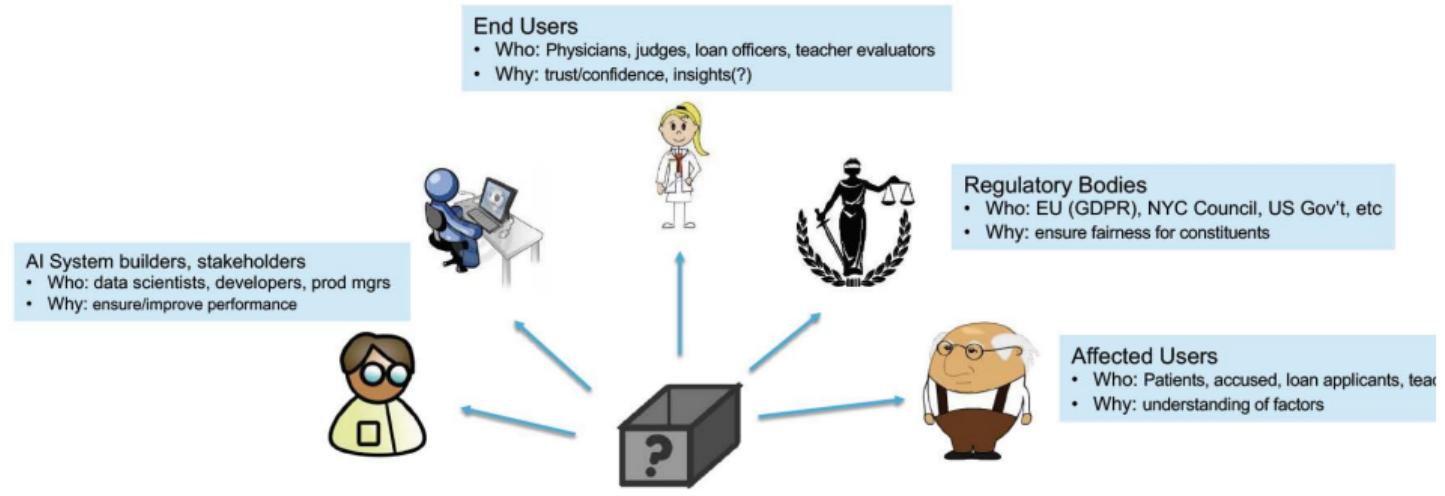
- ▶ Un processus abductif qui permet de déterminer une explication pour un événement donné dont les causes sont identifiées
Rappel abduction :
on observe b , on sait que $a \Rightarrow b$, on en déduit a
- ▶ Un produit, qui est le résultat du processus précédent
- ▶ Un processus qui permet de transférer cette connaissance de l'entité qui fournit l'explication à celle qui la reçoit.

Explicabilité pour qui ?



Source : IBM. Talk of Francesca Rossi

Explicabilité pour qui ?



Must match the **complexity capability** of the consumer
 Must match the **domain knowledge** of the consumer

Source : IBM. Talk of Francesca Rossi

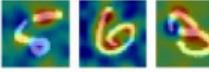
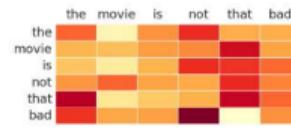
Types d'explications¹

- ▶ **Explication causale (why)** : la plus courante, la réponse à "pourquoi quelque chose s'est produit?", "Comment cela fonctionne?".
- ▶ **Explication par analogie/par l'exemple (what-else)** : il s'agit de trouver un exemple similaire qui génère des résultats identiques
- ▶ **Explication contrefactuelle (How-to)** : elle fournit des informations à propos de comment, quand, pourquoi la décision changerait ("Que se passerait-il si ?").
- ▶ **Explication contrastive (why-not)** : explore les possibilités de variation des entrées ("Vous m'avez dit que c'est X, mais pourquoi ce serait pas Y ?").

1. <https://arxiv.org/abs/1811.11839>

Types d'explications : exemples

Table 1: Examples of explanations divided for different data type and explanation

TABULAR	IMAGE	TEXT								
Rule-Based (RB) A set of premises that the record must satisfy in order to meet the rule's consequence. $r = \text{Education} \leq \text{College}$ $\rightarrow \leq 50k$	Saliency Maps (SM) A map which highlight the contribution of each pixel at the prediction. 	Sentence Highlighting (SH) A map which highlight the contribution of each word at the prediction. the movie is not that bad								
Feature Importance (FI) A vector containing a value for each feature. Each value indicates the importance of the feature for the classification. <table border="1"> <tr> <td>capitalgain</td> <td>0.00</td> </tr> <tr> <td>education-num</td> <td>14.00</td> </tr> <tr> <td>relationship</td> <td>1.00</td> </tr> <tr> <td>hoursperweek</td> <td>3.00</td> </tr> </table>	capitalgain	0.00	education-num	14.00	relationship	1.00	hoursperweek	3.00	Concept Attribution (CA) Compute attribution to a target "concept" given by the user. For example, how sensitive is the output (a prediction of zebra) to a concept (the presence of stripes)? 	Attention Based (AB) This type of explanation gives a matrix of scores which reveal how the word in the sentence are related to each other. 
capitalgain	0.00									
education-num	14.00									
relationship	1.00									
hoursperweek	3.00									

Source : Bodria et al, Benchmarking and Survey of Explanation Methods for Black Box Models, 2021²

2. <https://arxiv.org/abs/2102.13076>

Types d'explications : exemples

Prototypes (PR)

The user is provided with a series of examples that characterize a class of the black box

$$p = \text{Age} \in [35, 60], \text{Education} \in [\text{College}, \text{Master}] \rightarrow \geq 50k \quad p = \begin{array}{c} \text{dog} \\ \text{cat} \\ \text{mouse} \end{array} \rightarrow \begin{array}{l} p = \dots \text{not bad ...} \rightarrow \\ \text{"cat"} \\ \text{"positive"} \end{array}$$

Counterfactuals (CF)

The user is provided with a series of examples similar to the input query but with different class prediction

$$\begin{array}{l} q = \text{Education} \leq \text{College} \rightarrow \\ \quad \leq 50k \\ c = \text{Education} \geq \text{Master} \rightarrow \\ \quad \geq 50k \end{array} \quad q = \begin{array}{c} \text{S} \\ \text{B} \end{array} \rightarrow \begin{array}{l} "3" \\ "8" \end{array}$$

$q =$
The movie is not that bad \rightarrow "positive"

$c =$
The movie is that bad \rightarrow "negative"

Source : Bodria et al, Benchmarking and Survey of Explanation Methods for Black Box Models, 2021³

3. <https://arxiv.org/abs/2102.13076>

Sommaire

Introduction

Constat et motivations

Qu'est ce que l'IA eXplicable ?

Terminologie et définitions

Principales approches

Modèles explicables par conception (transparents)

Modèles de régression

Generalised Additive Models (GAM)

Modèles à base d'arbres

Explications post-hoc

Méthodes indépendantes du modèle

Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Dilemne performance vs interprétabilité

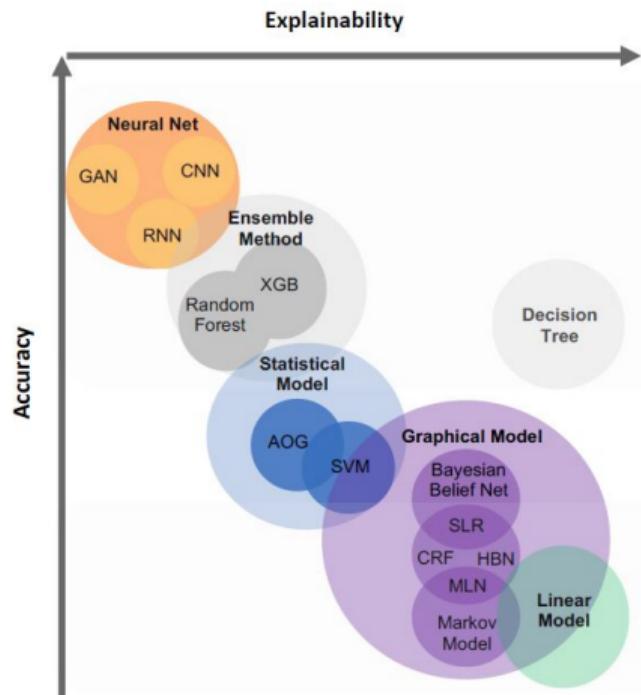


Figure – Modèles ML selon leur performance et leur interprétabilité

Principales approches

Approche 1 : Construire un modèle transparent

Utiliser un modèle qui est intrinsèquement explicable ou transparent

- ▶ Régression logistique, modèles à base d'arbres, GAM, ...

Approche 2 : Explicabilité Post-hoc

Accompagner un modèle boîte noire à l'aide d'un modèle complémentaire pour créer des interprétations.

- ▶ modèle **agnostique** vs modèle **spécifique**
- ▶ explication **locale** vs explication **globale** (inspection de modèle)

Principales approches

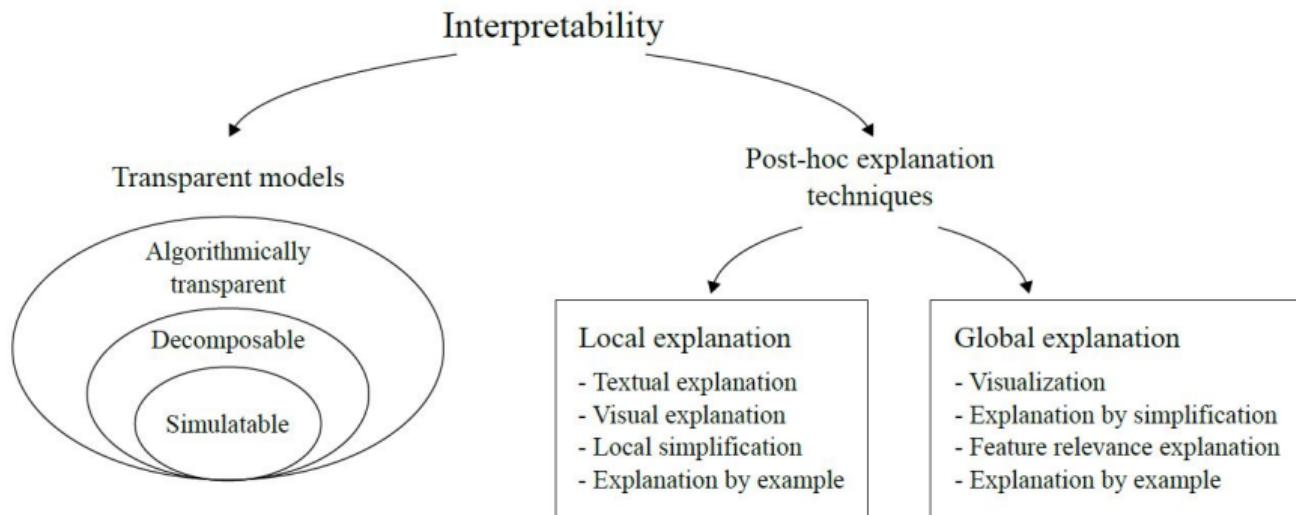


Figure – Les différents types d'interprétabilité selon Lipton et Arietta et al (Cherrier, 2021)

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Trois niveaux de transparence

Critères d'Arrieta et al

- ▶ **Simulable** : qui peut être exécuté “à la main” ou appréhendé par un humain (i.e. préférence pour les modèles parcimonieux)
- ▶ **Décomposable** : toutes les parties du modèle (entrées, hyperparamètres, calculs) sont explicables, de façon à pouvoir comprendre le comportement du modèle
- ▶ **Algorithmiquement transparent** : l'utilisateur peut comprendre le processus utilisé pour calculer les sorties à partir des entrées

Introduction

Différents modèles transparents

- ▶ régression linéaire
- ▶ arbres de décision
- ▶ k-plus proches voisins
- ▶ modèles à base de règles
- ▶ general additive models (GAM)
- ▶ modèles bayésiens
- ▶ ...

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Régression linéaire

Régression linéaire

La régression linéaire permet de prédire une sortie continue (régression) à partir d'une somme pondérée de ses entrées.

- ▶ Très utilisée en data science, permet d'étudier la dépendance d'une sortie par rapport à des entrées
- ▶ la relation linéaire exploitée est de la forme

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon$$

Régression linéaire

Régression linéaire

La régression linéaire permet de prédire une sortie continue (régression) à partir d'une somme pondérée de ses entrées.

- ▶ Très utilisée en data science, permet d'étudier la dépendance d'une sortie par rapport à des entrées
- ▶ la relation linéaire exploitée est de la forme

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon$$

Estimation

Les paramètres sont généralement déterminés par la méthode des moindres carrés :

$$\hat{\boldsymbol{\beta}} = \operatorname{argmin}_{\beta_0, \dots, \beta_p} \sum_{i=1}^n \left(y^i - \left(\beta_0 + \sum_{j=1}^p \beta_j x_j^i \right) \right)^2$$

Interprétation

Dataset bike rental

- ▶ On s'intéresse à la prédition du nombre de Vélib.
- ▶ On dispose d'un jeu de données avec un historique sur deux ans et les informations suivantes :
 - ▶ date, saison, vacances, jours ouvrés,
 - ▶ situation météo,
 - ▶ température, humidité, vitesse du vent.

Interprétation

Grandeurs d'intérêt

Pour interpréter la régression linéaire, nous allons nous intéresser aux valeurs suivantes :

- ▶ Squared Sum of Error (SSE) = combien de variance il reste après l'entraînement :

$$SSE = \sum_{i=1}^n (y^i - \hat{y}^i)^2$$

- ▶ Squared Sum of Data Variance (SST) = variance initiale de y :

$$SST = \sum_{i=1}^n (y^i - \bar{y})^2$$

- ▶ R-squared mesure combien de variance peut être expliquée par le modèle :

$$R^2 = 1 - \frac{SSE}{SST}$$

- ▶ La t-statistique : sa valeur absolue indique l'importance d'une entrée

$$t_{\hat{\beta}_j} = \frac{\hat{\beta}_j}{SE(\hat{\beta}_j)}$$

Interprétation

	Poids	SE	t
constante	2399.4	238.3	10.1
seasonSUMMER	899.3	122.3	7.4
seasonFALL	138.2	161.7	0.9
seasonWINTER	425.6	110.8	3.8
holiday	-686.1	203.3	3.4
workingday	124.9	73.3	1.7
weathersitMISTY	-379.4	87.6	4.3
weathersitBAD	-1901.5	223.6	8.5
temperature	110.7	7.0	15.7
humidity	-17.4	3.2	5.5
windspeed	-42.5	6.9	6.2
days_since_2011	4.9	0.2	28.5

Modèles apparentés

- ▶ ce que nous avons vu s'appelle régression linéaire multiple (x est un vecteur) mais par abus de langage on parle de régression linéaire
- ▶ régression non linéaire, polynomiale, etc.
- ▶ régression logistique : adaptée à la classification

Modèles de régression

Critères d'Arrieta et al

- ▶ Simulable : le modèle est simulable tant que les variables sont compréhensibles et que les interactions entre elles sont minimales
- ▶ Décomposable : le modèle n'est que décomposable si les entrées sont toujours lisibles mais les interactions sont devenues trop nombreuses et complexes et nécessitent d'être étudiées avec des outils mathématiques
- ▶ Algorithmiquement transparent : le modèle n'est qu'algorithmiquement transparent si les variables et les interactions sont trop complexes et nécessitent des outils mathématiques pour être étudiées.

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Generalised Additive Models

Motivations

- ▶ Régression linéaire : simple à interpréter car c'est un modèle additif
- ▶ Elle suppose que la sortie, connaissant les entrées, suit une distribution Gaussienne (en pratique, très rarement le cas)
- ▶ Elle suppose également que la relation entre la sortie et les entrées est linéaire

Generalised Additive Models

Motivations

- ▶ Régression linéaire : simple à interpréter car c'est un modèle additif
- ▶ Elle suppose que la sortie, connaissant les entrées, suit une distribution Gaussienne (en pratique, très rarement le cas)
- ▶ Elle suppose également que la relation entre la sortie et les entrées est linéaire
- ▶ Donc nécessité de corriger ces défauts

Generalised Additive Models

Generalised Linear Models

Pour corriger le problème sur la distribution, on propose les Generalized Linear Models :

$$g(E_Y(y|x)) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

Generalised Additive Models

Generalised Linear Models

Pour corriger le problème sur la distribution, on propose les Generalized Linear Models :

$$g(E_Y(y|x)) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

Avec 3 composantes :

- ▶ la somme pondérée (comme avant)
- ▶ une distribution de la famille des exponentielles ^a
- ▶ une fonction g qui lie la somme pondérée à la moyenne de la distribution

a. ex. : Normale, Bernouili, Poisson, Pareto, Laplace...

Generalised Additive Models

Generalised Linear Models

Pour corriger le problème sur la distribution, on propose les Generalized Linear Models :

$$g(E_Y(y|x)) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

Avec 3 composantes :

- ▶ la somme pondérée (comme avant)
- ▶ une distribution de la famille des exponentielles ^a
- ▶ une fonction g qui lie la somme pondérée à la moyenne de la distribution

a. ex. : Normale, Bernouili, Poisson, Pareto, Laplace...

En revanche, cela ne résout pas le problème de la linéarité.

Generalised Additive Models

Generalised Additive Models

Les GAM ont pour but de généraliser les GLM :

$$g(E_Y(y|x)) = \beta_0 + f_1(x_1) + f_2(x_2) + \dots + f_p(x_p)$$

Generalised Additive Models

Generalised Additive Models

Les GAM ont pour but de généraliser les GLM :

$$g(E_Y(y|x)) = \beta_0 + f_1(x_1) + f_2(x_2) + \dots + f_p(x_p)$$

Les termes linéaires ont été remplacés par des fonctions et on peut choisir des fonctions linéaires ou non-linéaires.

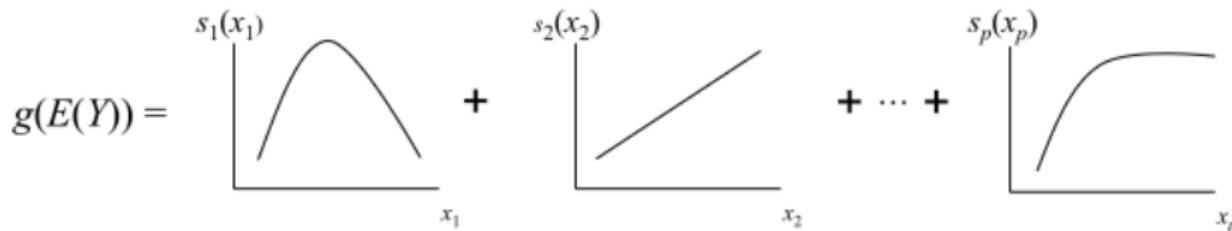
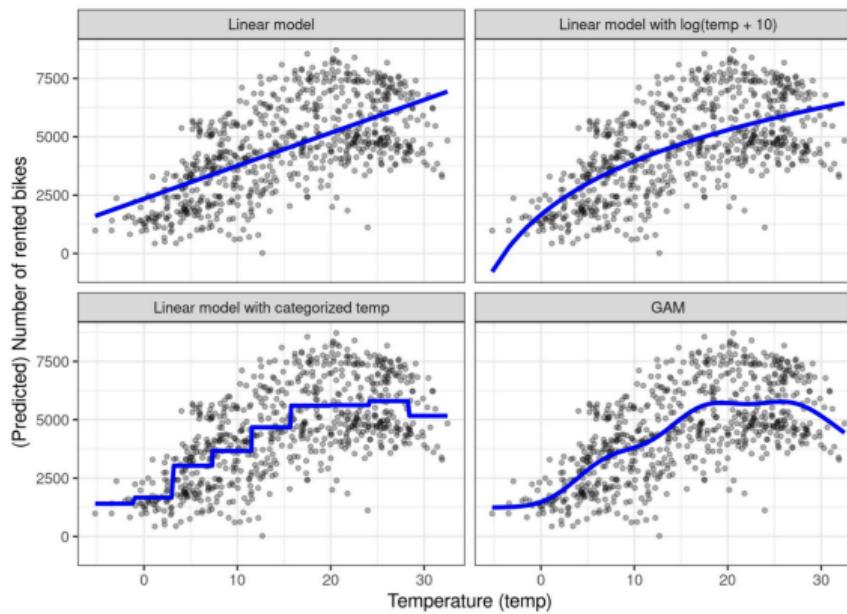


Figure – Decomposition of a GAM.

GAM - Example Rental Bike

La température a un effet positif linéaire sur le nombre de vélos de location, mais à un moment donné, elle s'estompe et a même un effet négatif lorsque les températures sont élevées.



Source : <https://christophm.github.io/interpretable-ml-book/extend-lm.html>

GAM - Exemple Rental Bike

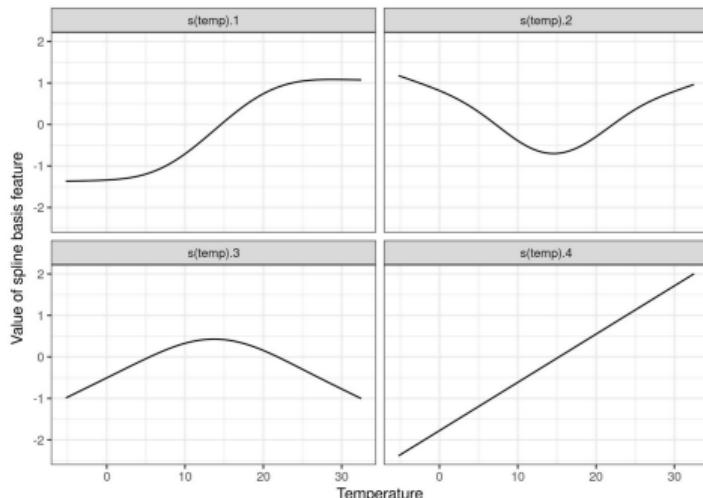


Figure – 4 spline basis functions

Source : <https://christophm.github.io/interpretable-ml-book/extend-lm.html>

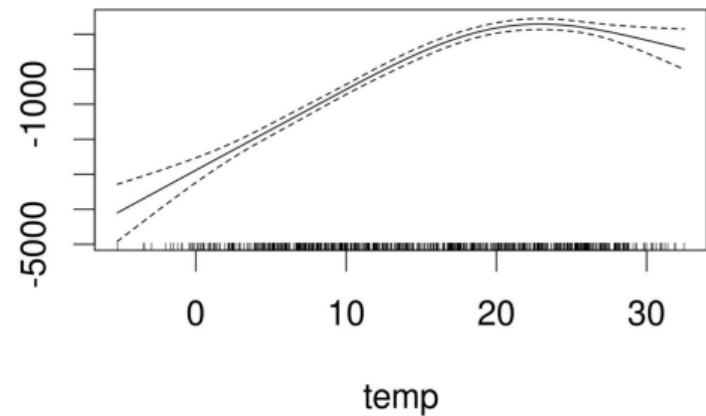


Figure – Effet GAM de la température sur la prédiction du nombre de vélos loués (température utilisée comme seule caractéristique).

Generalised Additive Models

Critères d'Arrieta et al

- ▶ Simulable : le modèle est simulable tant qu'on choisit des fonctions simples et un nombre de termes correspondant à la capacité de l'humain
- ▶ Décomposable : le modèle n'est plus décomposable si les interactions sont devenues trop complexes et qu'il faut des outils mathématiques pour les comprendre
- ▶ Algorithmiquement transparent : on tombe dans cette catégorie si les variables et les interactions sont trop complexes et nécessitent l'usage d'outils mathématiques extérieurs.

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

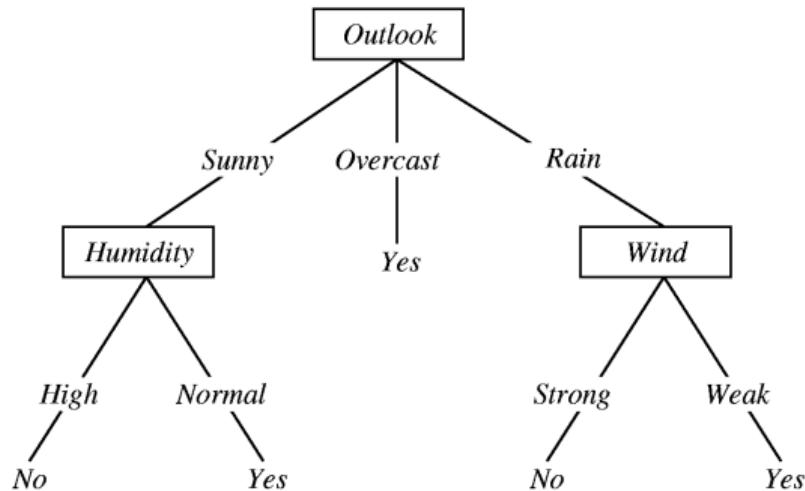
Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Arbres de décision

- ▶ Modèle symbolique
- ▶ Chaque noeud représente un test portant sur un attribut
- ▶ Chaque branche indique une valeur possible à un test
- ▶ Utilisables pour la classification ou la régression (on parle alors d'arbres de régression)



Arbres de décision

Induction d'un arbre de décision

- ▶ Plusieurs algorithmes, les plus connus étant ID3, C4.5, CART
- ▶ Apprentissage supervisé
- ▶ Ils diffèrent en général par leur capacité à gérer des attributs discrets ou continus, les valeurs manquantes, etc.
- ▶ Procèdent par division récursive de l'ensemble de données.

	Critère de split	Types d'attribut	Types de sortie	Valeurs manquantes	Elagage	Détection d'outlier
ID3	Info. Gain	Cat.	Cat.	Non	Aucun	Non
CART	Gini	Cat./Cont.	Cat./Cont.	Oui	Complexité	Oui
C4.5	Gain ratio	Cat./Cont.	Cat.	Oui	Erreur	Non

Arbres de décision

Induction d'un arbre de décision

```
Arbre ← ε
Noeud courant ← racine(Arbre)
repeat
    if Noeud courant est terminal then
        | Affecter une classe au noeud courant
    else
        | Sélectionner un test selon le critère de split
        | Créer les sous-arbres
        | Passer au noeud suivant non exploré
until arbre complet;
```

Arbres de décision

Interprétation

- ▶ L'attribut à la racine est le plus discriminant
- ▶ Une branche est une conjonction de tests
- ▶ L'arbre est une disjonction de branches
- ▶ Utilisation de la notion d'importance d'attribut :

$$\text{FI}(n) = \text{métrique}(n) \times \#\text{instances} - \text{métrique}(n_g) \times \#\text{instances à gauche} - \text{métrique}(n_d) \times \#\text{instances à droite}$$

puis on normalise.

Arbres de décision

Importance d'un attribut

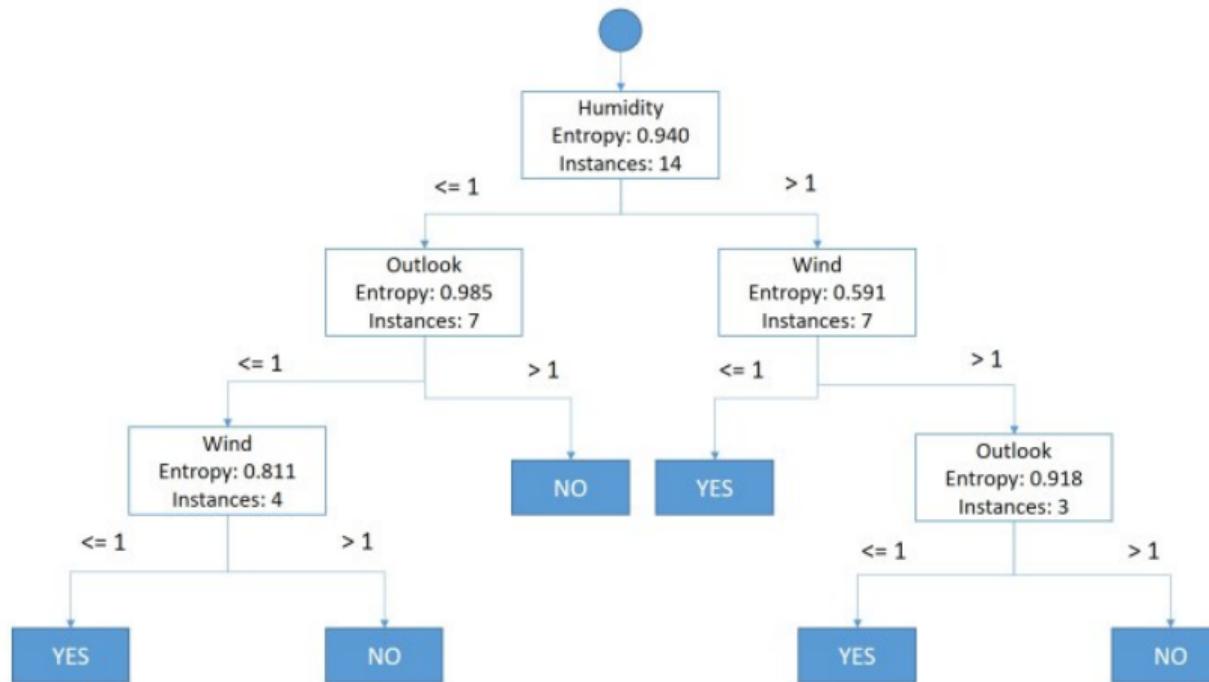
L'importance d'un attribut (feature importance) est un indicateur qui représente la réduction totale du critère de split utilisé (entropie, Gini...) et apportée par cet attribut. On utilise en général une valeur normalisée.

Plus ce critère est grand, plus l'attribut est important.

$$\text{FI}(n) = \text{métrique}(n) \times \#\text{instances} - \text{métrique}(n_g) \times \#\text{instances à gauche} - \text{métrique}(n_d) \times \#\text{instances à droite}$$

puis on normalise.

Arbres de décision



Arbres de décision

Importance d'attribut (feature importance)

$$FI(\text{humidity}) = 14 \times 0.940 - 7 \times 0.985 - 7 \times 0.591 = 2.121$$

$$FI(\text{Outlook}, \text{ 2e niv}) = 7 \times 0.985 - 4 \times 0.811 = 3.651$$

$$FI(\text{Wind}, \text{ 2e niv}) = 7 \times 0.591 - 3 \times 0.918 = 1.390$$

$$FI(\text{Wind}, \text{ 3e niv}) = 4 \times 0.811 = 3.244$$

$$FI(\text{Outlook}, \text{ 3e niv}) = 3 \times 0.918 = 2.754$$

$$FI(\text{Humidity}) = 2.121$$

$$FI(\text{Outlook}) = 3.651 + 2.754 = 6.405$$

$$FI(\text{Wind}) = 1.390 + 3.244 = 4.634$$

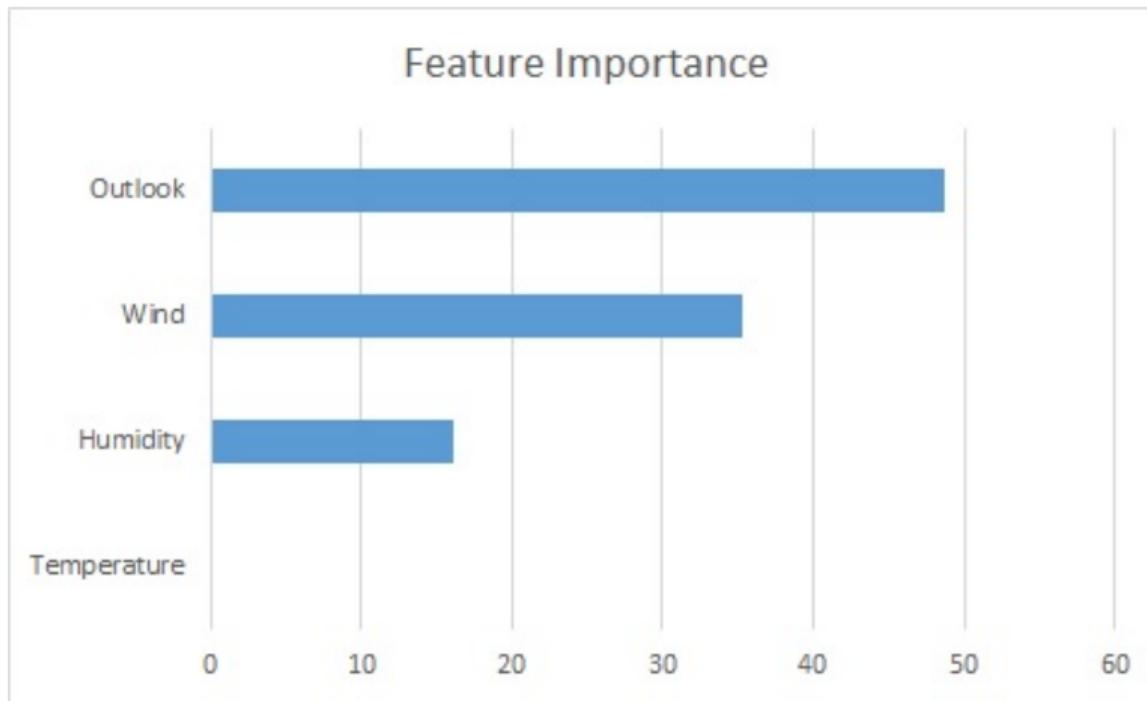
$$\text{somme} = 2.121 + 6.405 + 4.634 = 13.16$$

$$FI(\text{Humidity}) = 0.16$$

$$FI(\text{Outlook}) = 0.48$$

$$FI(\text{Wind}) = 0.35$$

Arbres de décision



Arbres de décision

Bilan

- ▶ Modèles très utilisés
- ▶ Interprétables
- ▶ Visualisables
- ▶ Induction rapide
- ▶ Frontières de décision parallèles aux axes
- ▶ L'espace des entrées est découpé en hyper-rectangles

Arbres de décisions

Critères d'Arrieta et al

- ▶ Simulable : un arbre est simulable si le modèle est exécutable par un humain sans connaissances mathématiques
- ▶ Décomposable : l'arbre tombe dans cette catégorie si les branches sont compréhensibles paquet par paquet
- ▶ Algorithmiquement transparent : les branches sont tellement longues et nombreuses qu'il faut des outils mathématiques pour les comprendre

Bilan

Les modèles transparents

- ▶ sont conçus pour des données tabulaires uniquement
- ▶ sont très souvent utilisés comme surrogates de modèles plus complexes
- ▶ sont investigués pour des méthodes hybrides : NodeGAM, Neural-Backed Decision Trees

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Vue d'ensemble

- ▶ Méthodes génériques (agnostiques) :
 - ▶ Explication par simplification
 - ▶ Explication par importance des features : fonctions d'influence, sensibilité, saillance
 - ▶ Explication locale
 - ▶ Explication visuelle : shapley, saillance
- ▶ Méthodes spécifiques à un modèle :
 - ▶ Explication par l'exemple
 - ▶ Explication visuelle / textuelle
 - ▶ Explication par importance des features
 - ▶ Explication par simplification
 - ▶ Explication locale

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Méthodes indépendantes du modèle

Différentes approches

- ▶ **Explication par simplification/locale** : il s'agit de la famille la plus grande, très souvent basée sur l'extraction de règles. Ex. : [LIME](#)
- ▶ **Explication par importance des features** : le but de ces méthodes est d'expliquer un modèle boîte noire en classant les features par contribution à la décision. Ex. : [SHAP](#)
- ▶ **Explication visuelle** : il s'agit bien souvent de graphes extraits des modèles et/ou des features. Cette famille est peu représentée car il est difficile de concevoir des méthodes de visualisation sans hypothèse sur le modèle.

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

LIME

Définition

Local Interpretable Model-agnostic Explanations est une méthode de simplification locale de modèles boîtes noires. L'idée est d'identifier un modèle interprétable qui reproduit fidèlement localement le modèle original.

LIME

Définition

Local Interpretable Model-agnostic Explanations est une méthode de simplification locale de modèles boîtes noires. L'idée est d'identifier un modèle interprétable qui reproduit fidèlement localement le modèle original.

Principe

A partir d'un modèle boîte noire et d'une instance à classer, on va générer des données dans un voisinage et entraîner un modèle interprétable.

<https://www.kdd.org/kdd2016/papers/files/rfp0573-ribeiroA.pdf>

LIME

Compromis fidélité/interprétabilité

On se donne :

- ▶ Une explication, vue comme un modèle $g \in G$, où G est la classe des modèles interprétables
- ▶ $\Omega(g)$ une mesure de complexité de g , Ex. :
 - ▶ si g est un arbre de décision, $\Omega(g)$ peut être sa profondeur
 - ▶ si g est un modèle linéaire, $\Omega(g)$ peut être le nombre de poids non nuls
- ▶ $\pi_x(z)$ une mesure de proximité d'une instance z à x , afin de définir la proximité autour de x
- ▶ enfin, $\mathcal{L}(f, g, \pi_x)$ qui mesure l'infidélité de g pour approximer f dans le voisinage défini par π_x .

LIME propose un modèle à la fois fidèle dans le voisinage d'une instance ($\mathcal{L}(f, g, \pi_x)$ minimal) et interprétable ($\Omega(g)$ minimal) :

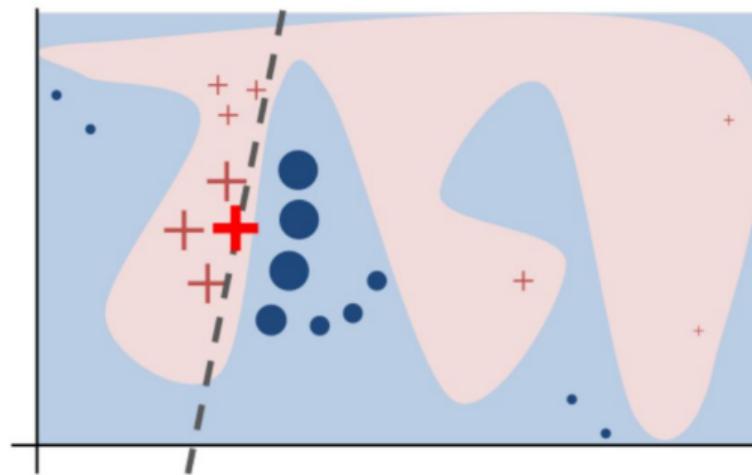
$$\xi(x) = \operatorname{argmin}_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g)$$



LIME

Algorithm

1. Générer N individus autour de l'instance x , pondérés par leur distance à x
2. Constituer un dataset
3. Entraîner un modèle g



LIME

Exemple voisinage

$$\pi_x(z) = e^{\frac{-D(x,z)^2}{\sigma^2}}$$

où σ est la largeur du voisinage, et avec D :

- ▶ distance cosinus pour du texte
- ▶ L2 pour des images
- ▶ etc.

Exemple mesure d'infidélité

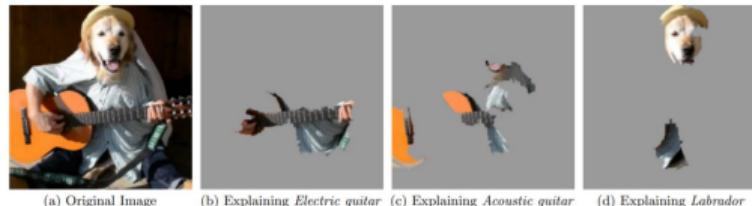
Très souvent, la mesure d'infidélité est la loss function :

$$\mathcal{L}(f, g, \pi_x) = \sum_{z, z' \in \mathcal{Z}} \pi_x(z) (f(z) - g(z'))^2$$

LIME

Avantages

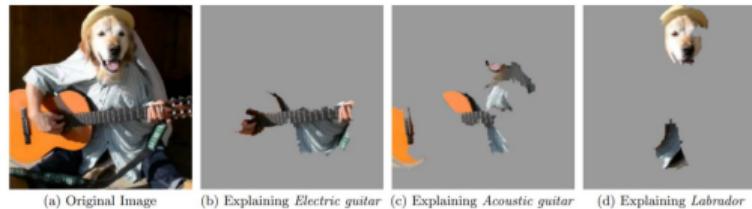
- ▶ agnostique
- ▶ rapide et simple
- ▶ applicable aux données tabulaires, images et texte



LIME

Avantages

- ▶ agnostique
- ▶ rapide et simple
- ▶ applicable aux données tabulaires, images et texte



Inconvénients

- ▶ approximation uniquement locale
- ▶ très dépendant du choix du voisinage
- ▶ instable
- ▶ si localement le modèle reste complexe, LIME ne fonctionnera pas

Shapley

Origines

- ▶ Théorie des jeux
- ▶ Dans un jeu coopératif, les joueurs collaborent pour obtenir un certain gain
- ▶ Lloyd Shapley s'est intéressé à la répartition équitable de ce gain entre les joueurs
- ▶ On parle de coalition pour désigner un sous-ensemble non vide de joueurs

Shapley

Origines

- ▶ Théorie des jeux
- ▶ Dans un jeu coopératif, les joueurs collaborent pour obtenir un certain gain
- ▶ Lloyd Shapley s'est intéressé à la répartition équitable de ce gain entre les joueurs
- ▶ On parle de coalition pour désigner un sous-ensemble non vide de joueurs

L'idée est donc d'appliquer ça aux features et savoir lesquels ont contribué à une décision en particulier.

Shapley

Shapley values

Etant donné un ensemble de features, il s'agit de trouver pour chaque feature sa contribution marginale à la prédiction.

Il faut donc imaginer une valeur prédictive de base, et comment chaque feature force la prédiction à s'éloigner de cette valeur de base.

Exemple

On considère un modèle qui calcule le prix d'un appartement. Pour un appartement donné, il prédit 300.000eur. L'appartement est au 2e étage, fait 50m², est situé près d'un parc et le propriétaire précédent n'avait pas de chat. Supposons que le prix moyen prédit est 310.000eur. On veut savoir comment chaque feature a joué un rôle.

Shapley

Calcul pour un modèle linéaire

Soit un modèle linéaire : $\hat{f}(x) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$.

La contribution ϕ_j du $j^{\text{ième}}$ feature sur la prédiction $\hat{f}(x)$ s'écrit :

$$\phi_j(\hat{f}) = \beta_j x_j - E(\beta_j X_j) = \beta_j x_j - \beta_j E(X_j)$$

avec $E(\beta_j X_j)$ l'effet moyen du feature j.

A présent, sommes les contributions pour cet exemple :

$$\begin{aligned} \sum_{j=1}^p \phi_j(\hat{f}) &= \sum_{j=1}^p (\beta_j x_j - E(\beta_j X_j)) \\ &= (\beta_0 + \sum_{j=1}^p \beta_j x_j) - (\beta_0 + \sum_{j=1}^p E(\beta_j X_j)) \\ &= \hat{f}(x) - E(\hat{f}(X)) \end{aligned}$$

On retrouve bien ce que l'on intuitait : il s'agit de la valeur prédictive moins la moyenne des valeurs prédictives.



Shapley (Bike rental Dataset)

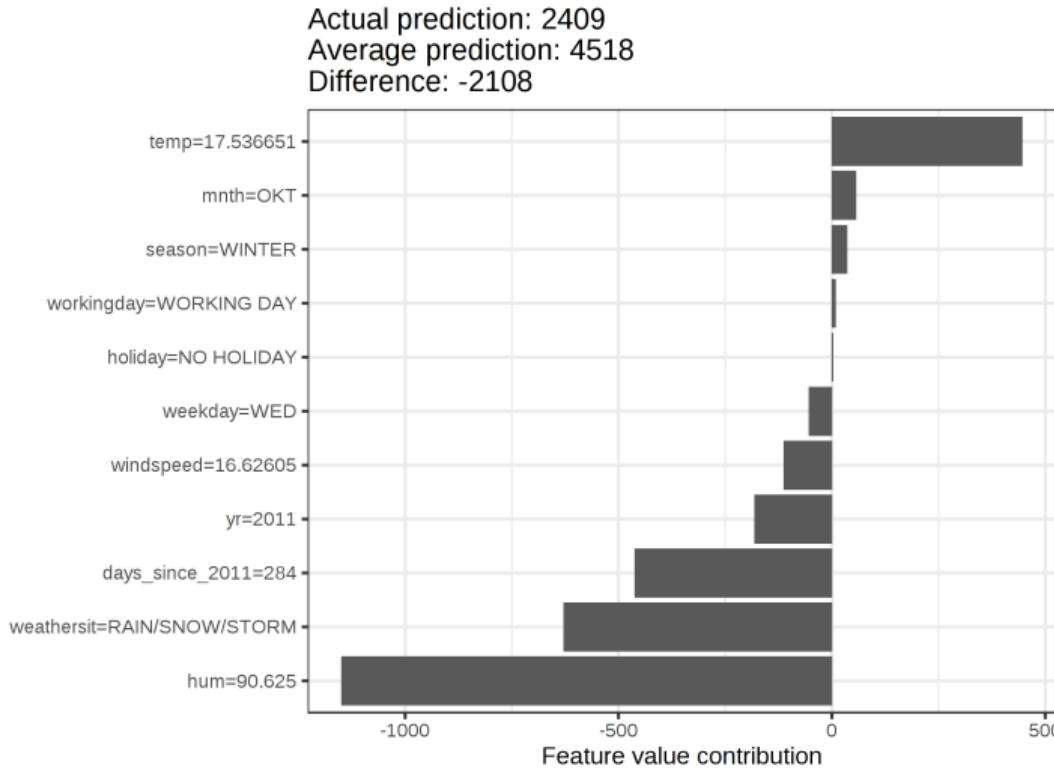
Cas général

Dans le cadre général, la contribution s'écrit :

$$\phi_j(\text{val}) = \sum_{S \subseteq \{x_1, \dots, x_p\} \setminus \{x_j\}} \frac{|S|!(p-|S|-1)!}{p!} (\text{val}(S \cup \{x_j\}) - \text{val}(S)) \text{ avec :}$$

- ▶ S , un sous-ensemble de features
- ▶ x , l'instance qui vient d'être classée
- ▶ p , le nombre de features
- ▶ $\text{val}(S) = E[\hat{f}(x)|x_S]$

Shapley



Shapley

En pratique

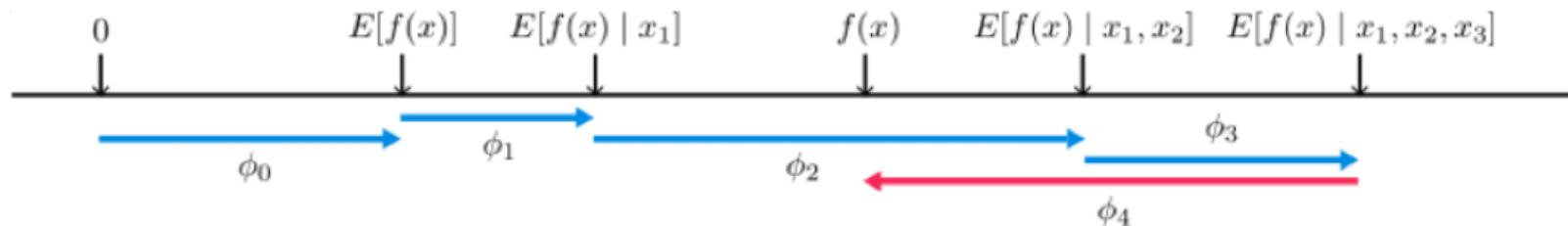
- ▶ pour les modèles linéaires, les calculs sont plus nombreux et plus coûteux
- ▶ notamment, il faut calculer la contribution de chaque coalition, or il y a 2^P coalitions
- ▶ on ne calcule pas les valeurs de Shapley exactes, on utilise des formules d'approximation
- ▶ il s'agit d'une méthode qui s'appuie sur une théorie sûre et très étudiée

SHAP

SHapley Additive exPlanation

- ▶ SHAP fait le lien entre les méthodes comme LIME et les valeurs de Shapley
- ▶ On considère une modèle additif défini à partir des coefficients de shapley et la valeur de base (ϕ_0) :

$$g(x') = \phi_0 + \sum_{j=1}^M \phi_j$$



SHAP

En pratique

- ▶ mêmes avantages que pour les valeurs de shapley
- ▶ mêmes inconvénients : temps de calcul
- ▶ cependant, il existe des versions spécifiques pour certains modèles : shapTree est une version très efficace pour les modèles à base d'arbres

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

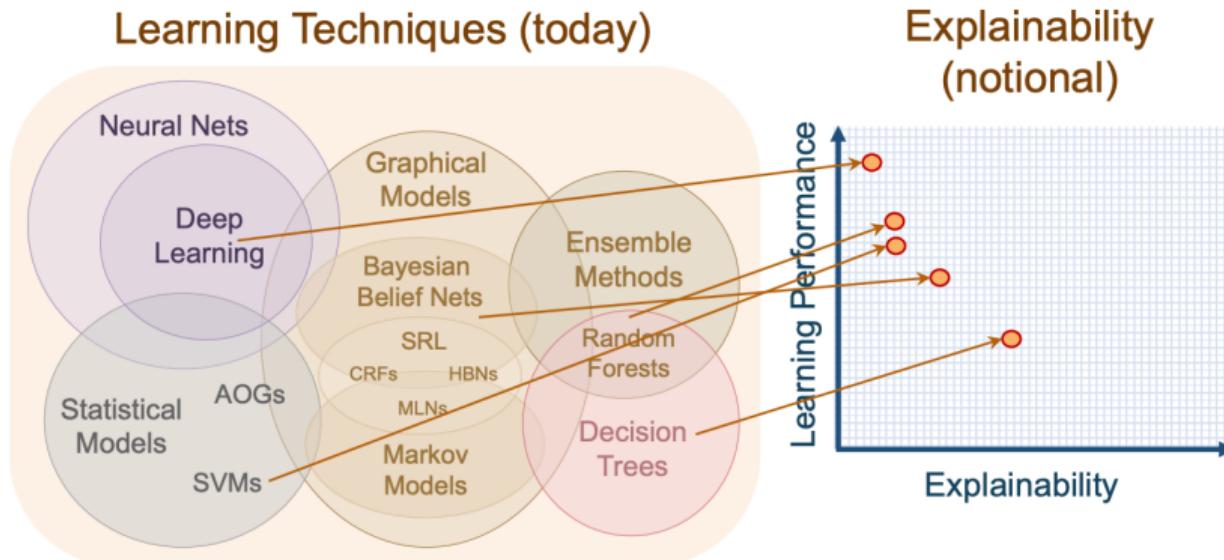
Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Motivations

Les systèmes d'IA basés sur des réseaux neuronaux profonds (DNNs) sont omniprésents mais sont des **modèles boîte noire**, c'est-à-dire des modèles dont les éléments internes sont soit inconnus de l'observateur, soit connus mais in-interprétables par l'homme⁴.



4. Guidotti et al - A Survey of Methods for Explaining Black Box Models
<https://dl.acm.org/doi/pdf/10.1145/3236009>

Motivations

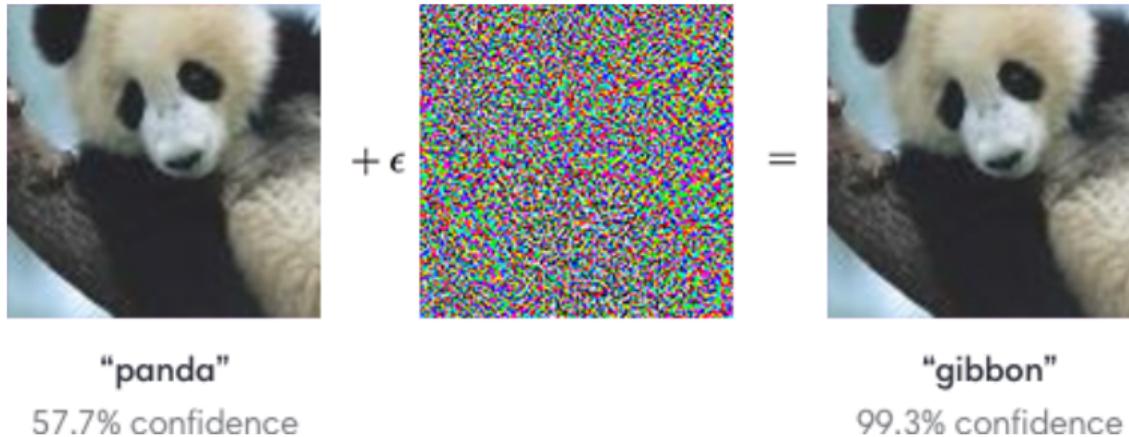


Figure – Goodfellow el al, Explaining and Harnessing adversarial examples. See <https://arxiv.org/pdf/1412.6572.pdf>

Ces modèles peuvent être trompés : ⇒ Besoin d'améliorer leur interprétabilité et explicabilité

Une large littérature, plusieurs taxonomies

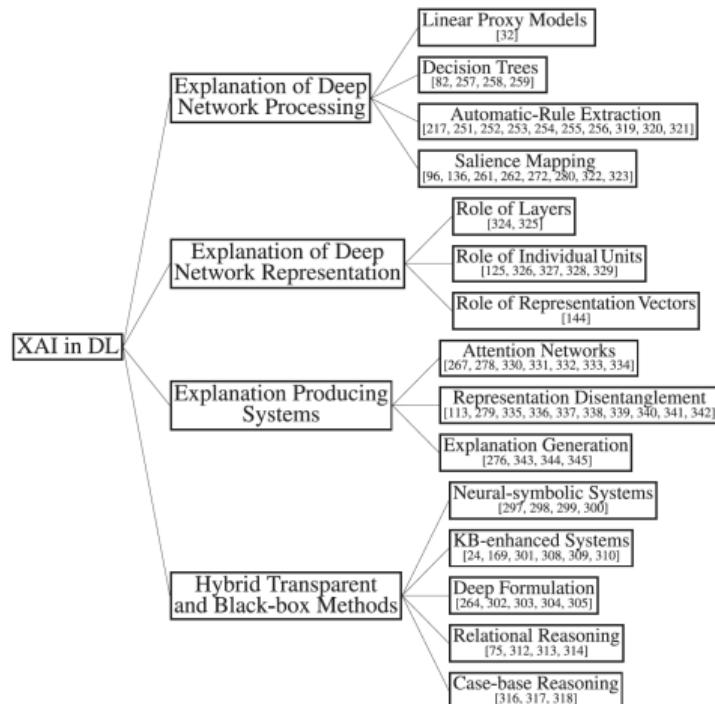


Figure – Source :Arrieta et al

Une large littérature, plusieurs taxonomies

Taxonomie de Xie et al^a Explainable Deep Learning : A Field Guide for the Uninitiated^b

a. (Xie et al,)

b. <https://arxiv.org/abs/2004.14545>

- ▶ **Méthode de visualisation** : Génère une explication en mettant en évidence, par le biais d'une **visualisation**, les caractéristiques d'une entrée qui influencent fortement la sortie d'un DNN.
 - ▶ Approches par back-propagation.
 - ▶ Approches par perturbation.
- ▶ **Distillation** : développe un modèle d'apprentissage machine distinct, transparent, qui est entraîné pour imiter le comportement d'entrée-sortie du réseau profond.
 - ▶ Approximations locales.
 - ▶ Traduction de modèles
- ▶ **Approches intrinsèques** : DNNs qui ont été spécifiquement créés pour donner une explication en même temps que le résultat.
 - ▶ Mécanismes d'attention
 - ▶ Réseaux auto-explicables

Une large littérature, plusieurs taxonomies

Taxonomie de Zhang et al⁵

Dimension 1 — Passive vs. Active Approaches	
Passive	Post-hoc explain trained neural networks
Dimension 2 — Type of Explanations (in the order of increasing explanatory power)	
To explain a prediction/class by	
Examples	Provide example(s) which may be considered similar or as prototype(s)
Attribution	Assign credit (or blame) to the input features (e.g. feature importance, saliency masks)
Hidden semantics	Make sense of certain hidden neurons/layers
Rules	Extract logic rules (e.g. decision trees, rule sets and other rule formats)
Dimension 3 — Local vs. Global Interpretability (in terms of the input space)	
Local	Explain network's <i>predictions on individual samples</i> (e.g. a saliency mask for a input image)
Semi-local	In between, for example, explain a group of similar inputs together
Global	Explain the network <i>as a whole</i> (e.g. a set of rules/a decision tree)

Figure – Source :Zhang et al

Dans ce tutorial

Un tour rapide et non exhaustif de la littérature :

- ▶ Explications par visualisation.
 - ▶ Basées sur la rétro-propagation.
 - ▶ Basées sur des perturbations.
- ▶ Explications par concepts.
- ▶ Explications par construction : Modèles interprétables par design.

Explications par visualisation

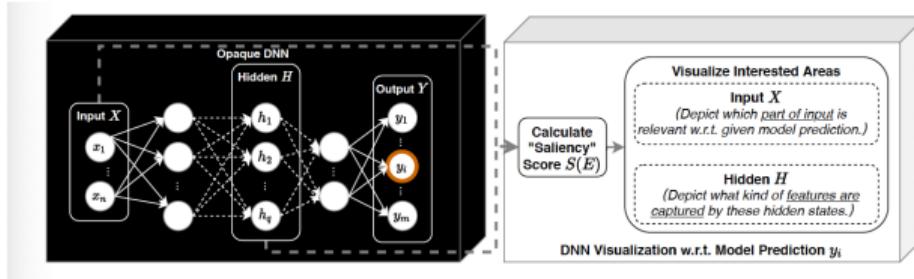


Figure – Source : Xie et al

Principe

- ▶ Méthodes qui associent le degré de prise en compte des caractéristiques (d'entrée ou cachées) par un réseau de neurones profond à une décision.
- ▶ Forme commune : [cartes de saillance](#)
 - ▶ Cartes qui identifient les caractéristiques d'entrée ou cachées les plus importantes pour la décision.

Explications par visualisation

Deux approches principales

- ▶ **Basées sur la rétro-propagation du gradient** : la pertinence des caractéristiques est fonction du volume du gradient passé à travers les couches du réseau au cours de l'apprentissage.
- ▶ **Basées sur des perturbations** : pertinence des caractéristiques en comparant la sortie du réseau et une copie modifiée de l'entrée.

Approches par rétro-propagation

- ▶ On considère un modèle de classification $f(\mathbf{x})$
- ▶ Entrée : vecteur $\mathbf{x} \in \mathbb{R}^d$
- ▶ $f_y(\mathbf{x})$ est la probabilité pour la classe y .
- ▶ On fait l'hypothèse que f est différentiable.

Approches par rétro-propagation

Les modèles profonds sont entraînés par descente de gradient stochastique (SGD)

- ▶ Entrée : Θ paramètres, η taux d'apprentissage, b taille du batch, \mathcal{D} jeu de données.
- ▶ Initialisation de $\theta_0 \in \Theta$.
- ▶ Pour un nombre donné d'itérations :

Approches par rétro-propagation

- ▶ On considère un modèle de classification $f(\mathbf{x})$
- ▶ Entrée : vecteur $\mathbf{x} \in \mathbb{R}^d$
- ▶ $f_y(\mathbf{x})$ est la probabilité pour la classe y .
- ▶ On fait l'hypothèse que f est différentiable.

Approches par rétro-propagation

Les modèles profonds sont entraînés par descente de gradient stochastique (SGD)

- ▶ Entrée : Θ paramètres, η taux d'apprentissage, b taille du batch, \mathcal{D} jeu de données.
- ▶ Initialisation de $\theta_0 \in \Theta$.
- ▶ Pour un nombre donné d'itérations :
 - ▶ Échantillonne $\mathcal{B} \sim \mathcal{D}$ tel que $|\mathcal{B}| = b$.

Approches par rétro-propagation

- ▶ On considère un modèle de classification $f(\mathbf{x})$
- ▶ Entrée : vecteur $\mathbf{x} \in \mathbb{R}^d$
- ▶ $f_y(\mathbf{x})$ est la probabilité pour la classe y .
- ▶ On fait l'hypothèse que f est différentiable.

Approches par rétro-propagation

Les modèles profonds sont entraînés par descente de gradient stochastique (SGD)

- ▶ Entrée : Θ paramètres, η taux d'apprentissage, b taille du batch, \mathcal{D} jeu de données.
- ▶ Initialisation de $\theta_0 \in \Theta$.
- ▶ Pour un nombre donné d'itérations :
 - ▶ Échantillonne $\mathcal{B} \sim \mathcal{D}$ tel que $|\mathcal{B}| = b$.
 - ▶ Calcul de la fonction de coût $\mathcal{L}^{\mathcal{B}}(\theta) := \frac{1}{b} \sum_{(x,y) \in \mathcal{B}} \ell(y, f_\theta(x))$.

Approches par rétro-propagation

- ▶ On considère un modèle de classification $f(\mathbf{x})$
- ▶ Entrée : vecteur $\mathbf{x} \in \mathbb{R}^d$
- ▶ $f_y(\mathbf{x})$ est la probabilité pour la classe y .
- ▶ On fait l'hypothèse que f est différentiable.

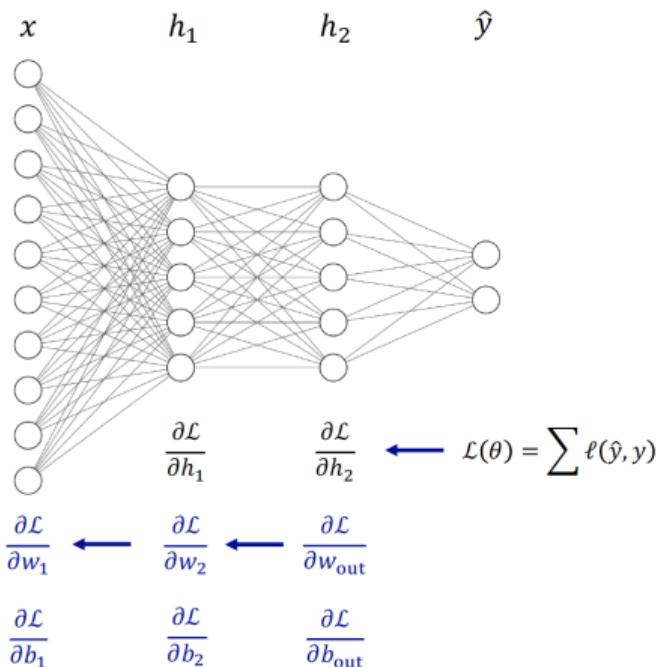
Approches par rétro-propagation

Les modèles profonds sont entraînés par descente de gradient stochastique (SGD)

- ▶ Entrée : Θ paramètres, η taux d'apprentissage, b taille du batch, \mathcal{D} jeu de données.
- ▶ Initialisation de $\theta_0 \in \Theta$.
- ▶ Pour un nombre donné d'itérations :
 - ▶ Échantillonne $\mathcal{B} \sim \mathcal{D}$ tel que $|\mathcal{B}| = b$.
 - ▶ Calcul de la fonction de coût $\mathcal{L}^{\mathcal{B}}(\theta) := \frac{1}{b} \sum_{(x,y) \in \mathcal{B}} \ell(y, f_\theta(x))$.
 - ▶ Mise à jour des paramètres selon :

$$\theta_{t+1} \leftarrow \theta_t - \eta \alpha(t) (\nabla_{\theta} \mathcal{L}^{\mathcal{B}})(\theta_t)$$

Rétro-propagation : règle en chaîne

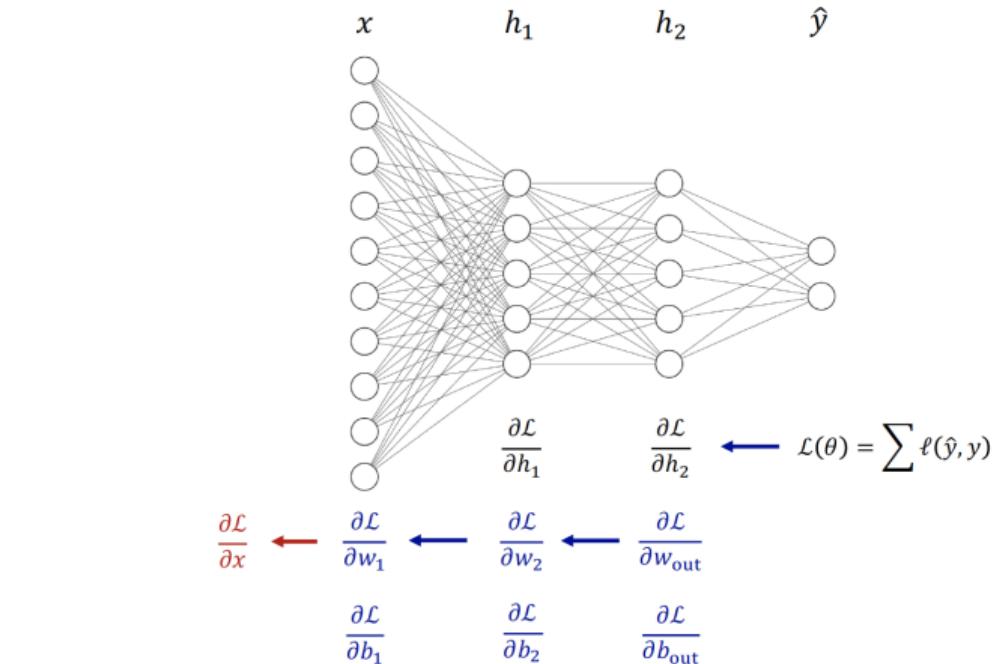


Explications par rétro-propagation

Idée principale

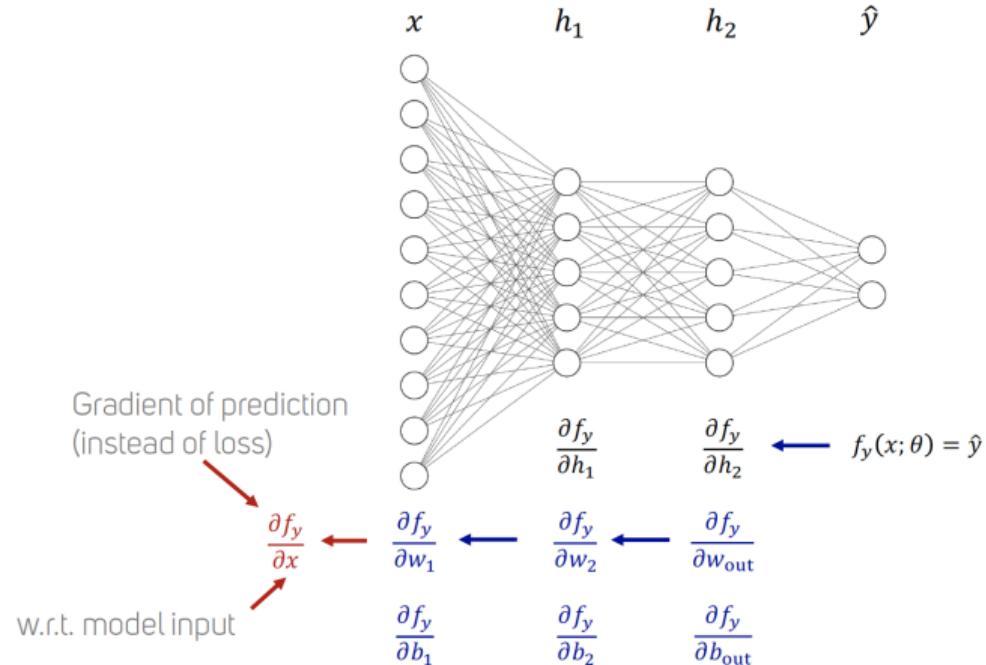
- ▶ Utiliser l'idée de rétro-propagation pour quantifier l'importance des caractéristiques
- ▶ Au lieu de calculer les gradients par rapport aux paramètres, calculer les gradients par rapport aux entrées ou aux caractéristiques cachées.

Explications par rétro-propagation



Source : Su-In Lee

Explications par rétro-propagation



Slide credit : Su-In Lee

Explications par rétro-propagation

Intuition :

- ▶ Les dérivées partielles représentent la sensibilité aux petites perturbations
- ▶ Soit, plus formellement :

$$\frac{\partial f_y}{\partial x_i}(x) = \lim_{\epsilon \rightarrow 0} \frac{f_y(x + e_i \cdot \epsilon) - f_y(x)}{\epsilon}$$

Limite lorsque le changement devient très faible.

$e_i \cdot \epsilon$: delta d'un petit changement dans la i -ième direction.

Explications par rétro-propagation

Différentes approches

- ▶ **Approches par gradients.**
- ▶ Activation Maximization.
- ▶ Deconvolution.
- ▶ Layer-Wise Relevance Propagation.
- ▶ DeepLift : Deep Learning Important FeaTures.

Explication par gradient

(Simonyan et al, 2014) Deep Inside Convolutional Networks : Visualising Image Classification Models and Saliency Maps.⁶

Visualisation de la saillance d'une classe spécifique sur une image : support spatial d'une classe dans une image donnée

Étant donnée la fonction de score d'un modèle pour une classe, laquelle des entrées a le plus d'influence sur le score ? : gradient du score par rapport à chaque entrée

- ▶ Pour l'entrée x , l'explication par le gradient est $E_{grad}(x) = \frac{\partial f_y}{\partial x}$
- ▶ Quantifie comment un changement dans chaque dimension d'entrée modifie la prédiction $f_y(x)$ dans un petit voisinage autour de l'entrée pour la classe y .

6. <https://arxiv.org/pdf/1312.6034.pdf>

Explication par gradient

Justification

- ▶ Pour les réseaux de neurones, le score pour la classe y , f_y est une fonction non-linéaire de x .
- ▶ Etant donnée x_0 , on peut approximer $f_y(x)$ avec une fonction linéaire dans le voisinage de x_0 avec un développement de Taylor

$$f_y(x) \approx w^T x + b$$

avec w la dérivée de f_y par rapport à l'image x au point (image) x_0 .

$$w = \frac{\partial f_y}{\partial x} \Big|_{x_0}$$

Explication par gradient : exemples

Quelques cartes de saillance obtenues avec ce principe



Explication par gradients

Beaucoup d'extensions

- ▶ **SmoothGrad^a** : Gradients moyens des entrées près de x , par ajout de bruit

$$E_{sg}(x) = \frac{1}{N} \sum_{i=1}^N E(x + g_i)$$

- ▶ **GradientInput^c** : multiplication de l'entrée et du gradient.

$$E_{\text{grad-input}} = x \odot \frac{\partial f}{\partial x}$$

- ▶ **Guided Backpropagation (GBP)^d** : Combinaison avec DeconvNet, i.e. les entrées négatives du gradient sont mises à zéro lors de la rétropropagation à travers une unité ReLU.
- ▶ **Integrated Gradient** : Intégration (moyenne) des gradients le long d'un ensemble d'images

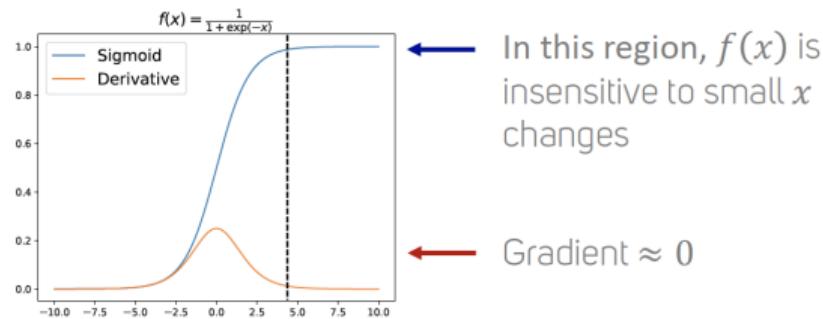
$$\phi_i^{IG}(f, x, x') = \underbrace{(x_i - x'_i)}_{\text{Difference from baseline}} \times \underbrace{\int_{\alpha=0}^1}_{\text{From baseline to input...}} \overbrace{\frac{\delta f(x' + \alpha(x - x'))}{\delta x_i} d\alpha}^{\dots \text{accumulate local gradients}}$$

- ▶ **VarGrad , Expected gradients, BlurIG ...**



Integrated Gradient

- ▶ Les gradients peuvent être saturés.
- ▶ Le modèle est sensible aux grands changements dans les entrées, mais pas aux petits.
- ▶ La saturation peut produire de petits gradients, même pour des entrées importantes.



Integrated Gradient

Idée

- ▶ Résoudre le problème de la saturation en calculant les gradients pour les images rescalées $\alpha \cdot x$

$$\frac{\partial S_c}{\partial x_i}(\alpha \cdot x) \text{ for } 0 \leq \alpha \leq 1$$

- ▶ Intégration (moyenne) des gradients sur toute une série d'images rescalées

$$\int_{\alpha=0}^1 \frac{\partial S_c}{\partial x_i}(\alpha \cdot x) d\alpha$$

- ▶ Multiplier par l'entrée

$$a_i = x_i \int_{\alpha=0}^1 \frac{\partial S_c}{\partial x_i}(\alpha \cdot x) d\alpha$$

- ▶ Repose implicitement sur une image de référence nulle (i.e. absence de la caractéristique dans l'entrée). Il est possible d'utiliser une image de référence non nulle x' .

Difference from baseline ...accumulate local gradients
 $\phi_i^{IG}(f, x, x') = \overbrace{(x_i - x'_i)}^{\text{From baseline to input...}} \times \underbrace{\int_{\alpha=0}^1 \frac{\delta f(x' + \alpha(x - x'))}{\delta x_i} d\alpha}_{\text{...}}$

Integrated Gradient

(Sundararajan et al,17) Axiomatic Attribution for Deep Networks⁷ Axiomatisation de l'attribution

Definition de l'attribution

Étant donné une fonction $F : \mathbb{R}^n \rightarrow [0, 1]$ représentant un réseau profond, une entrée $x = (x_1, \dots, x_n) \in \mathbb{R}^n$. Une attribution de la prédiction à l'entrée x par rapport à une entrée de base x_0 est un vecteur $a_F(a_1, \dots, a_n) \in \mathbb{R}^n$ où a_i est la contribution de x_i à la prédiction $F(x)$

Deux axiomes fondamentaux (caractéristiques souhaitables) pour les méthodes d'attribution

- ▶ **Sensitivity(a)** : Une méthode d'attribution satisfait à la Sensitivity(a) si, pour chaque image d'entrée et image de référence qui diffèrent par une caractéristique mais qui ont des prédictions différentes, il convient d'attribuer à la caractéristique qui diffère une attribution non nulle.
- ▶ **Implementation Invariance** : Les méthodes d'attribution doivent satisfaire à l'invariance de mise en œuvre, c'est-à-dire que les attributions sont toujours identiques pour deux réseaux fonctionnellement équivalents.

7. <https://arxiv.org/pdf/1703.01365.pdf>

Integrated Gradient

Axiome de complétude

- ▶ Proposition : SI $F : \mathbb{R}^n \rightarrow \mathbb{R}$ est différentiable presque partout

$$\sum_{i=1}^n \text{IntegratedGrads}_i(x) = F(x) - F(x')$$

- ▶ La somme des attributions doit être égale à la différence entre la sortie du DNN F pour l'entrée et celle pour l'image de référence.

Pour aller plus loin dans la compréhension voir :

<https://distill.pub/2020/attribution-baselines/>

Guided Integrated Gradient

Motivations et idées principales

- ▶ IG produit souvent des attributions de pixels fallacieuses ou bruitées dans des régions qui ne sont pas liées à la classe prédictive.
- ▶ L'une des causes est l'accumulation du bruit le long du chemin de l'IG.
- ▶ Idée : adapter le chemin IG lui-même : le chemin est conditionné non seulement par l'image mais aussi par le **modèle en cours d'explication**.
- ▶ Définir un chemin qui évite les régions d'entrée causant des anomalies : minimiser ℓ_{noise} à chaque caractéristique.

$$\gamma^{F*} = \arg \min_{\gamma^F \in \Gamma} \ell_{noise}$$

$$\ell_{noise} = \sum_{i=1}^N \int_{\alpha=0}^1 \left| \frac{\partial F(\gamma^F(\alpha))}{\partial \gamma_i^F(\alpha)} \frac{\partial \gamma_i^F(\alpha)}{\partial \alpha} \right| d\alpha$$

Méthode d'approximation gloutonne.

See : <https://pair-code.github.io/saliency/guided-ig>

(Kapishnikov et al,21) Guided Integrated Gradients : an adaptative Path Method for Removing Noise⁸

8. https://openaccess.thecvf.com/content/CVPR2021/papers/Kapishnikov_Guided_Integrated_2021_CVPR_paper.pdf

Guided Integrated Gradient

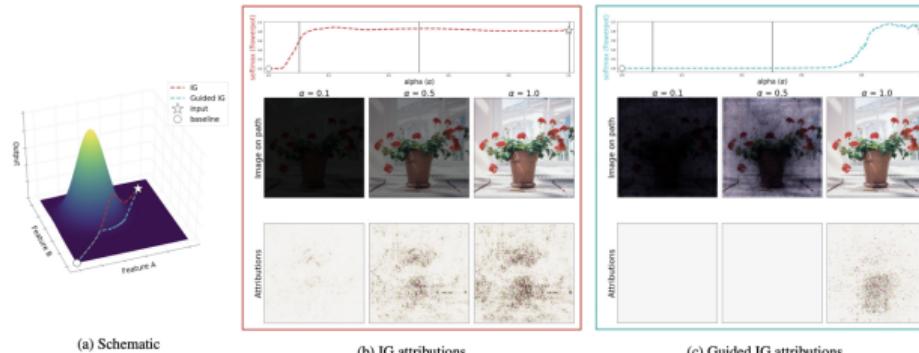


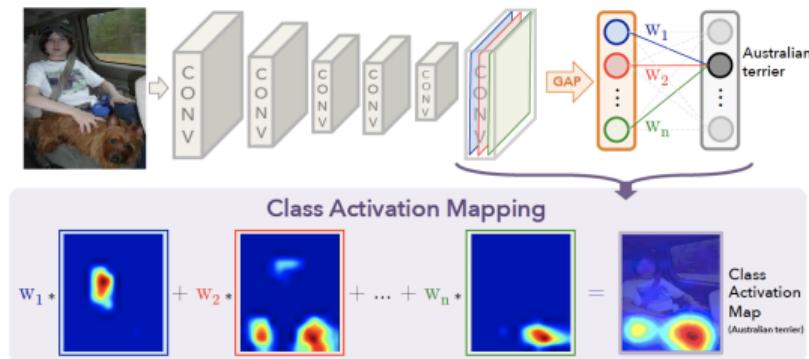
Figure 2: Comparing IG and Guided IG's paths and results. (a): For IG, a straight line path from baseline to input is followed (red dotted line), regardless of changes in gradients. For Guided IG, the path is chosen by selecting features that have the smallest absolute value of corresponding partial derivatives (cyan dotted line). Guided IG's goal is to reduce the accumulation of gradients caused by nearby very high/very low prediction examples. (b) and (c): Snapshots of attributions for the flower pot class for Integrated Gradients (center) and Guided IG (right) at alpha values of 0.1, 0.5, and 1.0. The top rows show graphs of the softmax prediction for flower pot as a function of alpha. The second row shows the input image produced by each technique at the three different alpha values. Note that IG's straight line path affects all pixels equally (e.g., see $\alpha = 0.5$), while Guided IG reveals the least important features, first. The third row shows each technique's attributions for each of the three alpha values, with Guided IG showing less noise outside the area of the image occupied by the flower pot.

(Kapishnikov et al,21) Guided Integrated Gradients : an adaptative Path Method for Removing Noise⁹

9. https://openaccess.thecvf.com/content/CVPR2021/papers/Kapishnikov_Guided_Integrated_Gradients_An_Adaptive_Path_Method_for_Removing_Noise_CVPR_2021_paper.pdf

CAM et GradCAM

Explication par des **Class Activation Maps** (CAM) basées sur le global average pooling dans les architectures de type CNNs



Global average pooling des cartes d'activation des dernières couches de convolution.

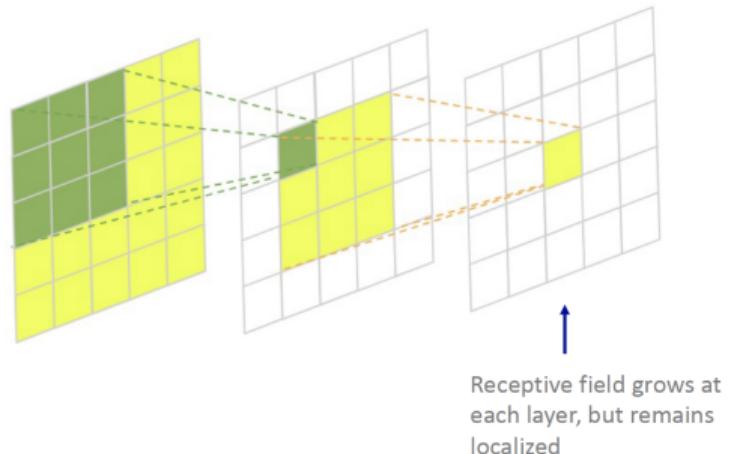
$$\text{map}_y = \sum_{i=1}^n w_{i,y} A_i$$

(Zhou et al, 2016) Learning Deep Features for Discriminative Localization¹⁰

10. <https://arxiv.org/pdf/1512.04150.pdf>

GradCAM : idées principales

- ▶ Dans les CNNs, les couches cachées peuvent représenter des concepts visuels de plus ou moins haut niveau.
- ▶ Les couches cachées portent une information spatiale à cause de la structure convolutionnelle (CNN receptive field)
- ▶ **Idée** : expliquer les modèles par la dernière couche de convolution plutôt que par les entrées.



GradCAM : procédure

- ▶ Soit A la représentation de la dernière couche cachée.
 - ▶ Taille de $A \in \mathbb{R}^{w \times h \times c}$
 - ▶ largeur w , longueur h et canaux c
 - ▶ Chaque canal $k = 1..c$ est noté $A^k \in \mathbb{R}^{w \times h}$
- ▶ La prédiction finale $f_y(x)$ peut être vue comme une fonction de A
 - ▶ E.g. $A \rightarrow \text{Global average pooling} \rightarrow \text{MLP}$.

GradCAM : procédure

- ▶ Gradients w.r.t. A

$$\frac{\partial f_y}{A_{ij}^k} \quad \forall(i, j, k)$$

- ▶ Moyenne des gradients dans chaque canal

$$\alpha_k^y = \frac{1}{wh} \sum_{ij} \frac{\partial f_y}{A_{ij}^k}$$

- ▶ Agréger les représentations cachées en utilisant α_k^y

$$a_{ij} = \sum_{k=1}^c \alpha_k^y A_{ij}^k$$

- ▶ Utilisation fréquente d'une fonction de seuillage (suppression des attributions négatives)

$$a_{ij} = \text{ReLU} \left(\sum_{k=1}^c \alpha_k^y A_{ij}^k \right)$$

GradCAM : Interprétation

- ▶ Les valeurs α_k^y représentent le gradient moyen ou lissé de la classe y w.r.t canal k .
- ▶ A chaque position, les activations A_{ij}^k sont multipliés par les gradients moyens et aggregés
- ▶ De manière similaire à *Grad – Input* (i.e. $\text{Grad} \times \text{Input}$), mais avec une couche cachée plutôt que l'entrée

CAM et GradCAM

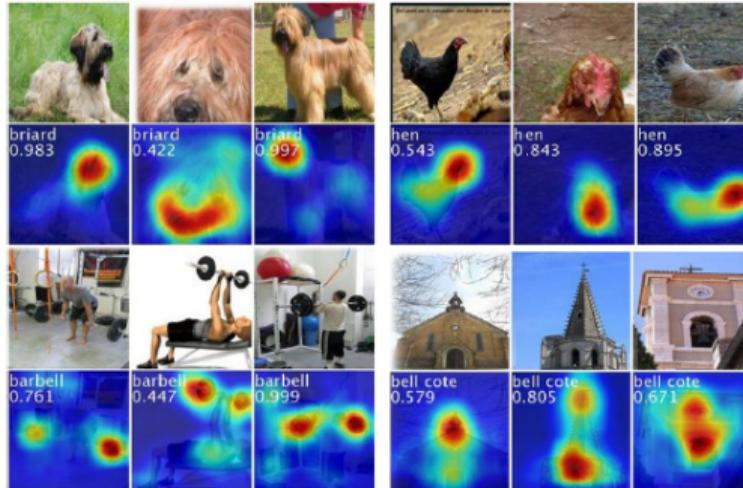
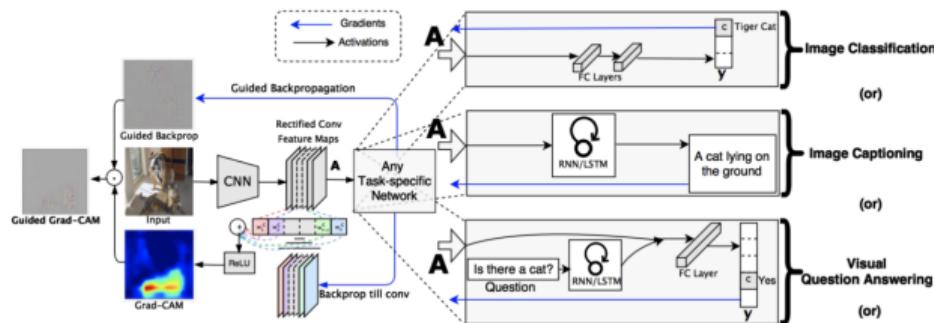


Figure 3. The CAMs of four classes from ILSVRC [20]. The maps highlight the discriminative image regions used for image classification e.g., the head of the animal for *briard* and *hen*, the plates in *barbell*, and the bell in *bell cote*.

CAM et GradCAM

Gradient-weighted Class Activation Map : généralisation¹¹ de CAM qui utilise les gradients de la sortie du réseau par rapport à la dernière couche convolutionnelle.



(Selvaraju et al., 2017)) Grad-cam : Why did you say that ?¹²

11. applicable à un grand nombre de CNNs, la dernière couche d'activation doit être différentiable

12. <https://arxiv.org/pdf/1611.07450.pdf>

CAM et GradCAM

Recap

- ▶ Gradient du score f_y (logit, avant softmax) de la classe y par rapport à chaque nœud de la couche de caractéristiques A_k dans la dernière couche de convolution est calculée et moyennée pour obtenir un score d'importance $\alpha_{k,y}$

$$\alpha_{k,y} = \frac{1}{m \cdot n} \sum_{i=1}^m \sum_{j=1}^n \frac{\partial S_y}{\partial A_{k,i,j}}$$

avec $A_{k,i,j}$ un neurone positionné à (i,j) dans la carte de caractéristiques A_k de taille $m \times n$

- ▶ Grad_CAM combine linéairement les scores d'importance dans chaque carte de caractéristique et passage dans une couche ReLU pour obtenir :

$$\text{map}_y = \text{ReLU}\left(\sum_k^c \alpha_{k,y} A_k\right)$$

Guided Grad-CAM = multiplication par points de Grad-CAM et Guided-Backpropagation.

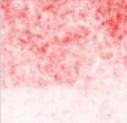
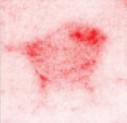
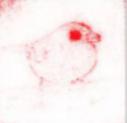
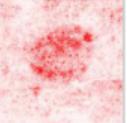
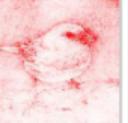
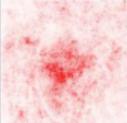
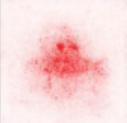
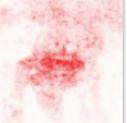
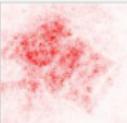
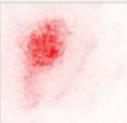
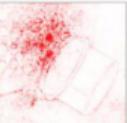
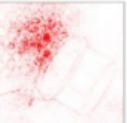
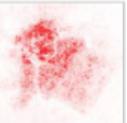
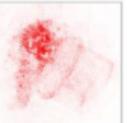
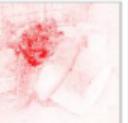
Rétro-propagation modifiée

- ▶ Les approches précédentes s'appuient sur la rétropropagation du gradient
- ▶ D'autres approches utilisent des variantes heuristiques de rétropropagation
 - ▶ L'importance des nœuds internes, propager vers les nœuds antérieurs.
 - ▶ Nécessité de justifier les différentes heuristiques de rétropropagation

Exemple : (Montavon et al, 2017) Layer-Wise Relevance Propagation : an overview¹³ : Calcul itératif des scores de pertinence pour chaque couche du modèle

13. <http://iphome.hhi.de/samek/pdf/MonXAI19.pdf>

Explication par rétro-propagation : exemples

	Original Image	Gradient	SmoothGrad	Guided BackProp	Guided GradCAM	Integrated Gradients	Integrated Gradients SmoothGrad	Gradient ⊙ Input	Edge Detector
Junco Bird									
Corn									
Wheaten Terrier									

Approches par perturbation

Principe

- ▶ Les méthodes basées sur la perturbation calculent la pertinence d'une caractéristique de l'entrée en **modifiant ou supprimant** la caractéristique d'entrée et en comparant la différence de sortie du réseau entre l'originale et la modifiée.
- ▶ Calcurent la pertinence de chaque caractéristique par rapport à la façon dont un réseau réagit à une entrée particulière.

Occlusion Sensitivity

(Zeiler et al, 2014) Visualizing and Understanding Convolutional Networks¹⁴

Principe

- ▶ Balayage d'un patch gris qui occulte des pixels de l'image et observation de la variation de la prédiction du modèle.
- ▶ Lorsque le patch couvre une zone critique, les performances de prédiction baissent de manière significative.
- ▶ La visualisation représente la sensibilité d'une zone d'une image par rapport à son étiquette de classification.

14. <https://csulb-ml.github.io/pdf/visualizing.pdf>

Approches par perturbation

(Fong et al, 2017) Interpretable Explanations of Black Boxes by Meaningful Perturbation¹⁵

Principe

- ▶ **Explication comme un meta-predicteur** : une explication est une règle qui prédit la réponse d'un modèle boîte noire f à certaines entrées.
- ▶ Exemple : on peut expliquer le comportement d'un classifieur de *rouge-gorge*(robin) par la règle :

$$Q_1(x; f) = \{x \in \mathcal{X}_c \iff f(x) = +1\}$$

où \mathcal{X}_c sous-ensemble d'images de rouge-gorge.

- ▶ f est imparfaite dont la règle ne s'applique que partiellement
- ▶ On peut mesurer la fidélité de l'explication par son erreur de prédiction attendue

$$\mathcal{L}_1 = \mathbb{E}[1 - \delta_{Q_1(x; f)}]$$

où δ_Q la fonction indicatrice de l'évènement Q .

- ▶ Utilisation de l'apprentissage pour découvrir les explications de manière automatique.

15. <https://arxiv.org/abs/1704.03296>

Approches par perturbation

(Fong et al, 2017) Interpretable Explanations of Black Boxes by Meaningful Perturbation¹⁶

Apprendre les explications

Trouver la meilleure explication Q est similaire à un problème d'apprentissage et peut être formulée par une minimisation de risque empirique régularisé.

$$\min_{Q \in \mathcal{Q}} \lambda \mathcal{R}(Q) + \frac{1}{n} \sum_{i=1}^n \mathcal{L}(Q(x_i; f), x_i, f), x_i \sim p(x)$$

La régularisation $\mathcal{R}(Q)$ a deux objectifs :

- ▶ permettre la généralisation des explications à plus de n exemples.
- ▶ choisir une explication Q qui est simple et interprétable

16. <https://arxiv.org/abs/1704.03296>

Approches par perturbation

Explication locale

- ▶ Une **explication locale** est une règle $Q(x; f, x_0)$ qui prédit la réponse de f au voisinage d'un point x_0
- ▶ Construction de Q à l'aide d'un développement de Taylor (f est lisse en x_0)
- ▶ La formulation fournit une interprétation des cartes de saillance comme des explications basées gradient
- ▶ La signification des perturbations dépend très largement de ce qui signifie **faire varier x pour le modèle boîte-noire**
- ▶ Trois principaux types de perturbations :
 - ▶ remplacer une région R par une valeur constante
 - ▶ injecter du bruit
 - ▶ flouter l'image

Types de perturbations

- ▶ Remplacer par une valeur constante μ

$$\Psi(x, m)_{ij} = m_{ij} \cdot x_{ij} + (1 - m_{ij}) \cdot \mu$$

- ▶ Remplacer avec du bruit $\epsilon \sim \mathcal{N}(0, \sigma^2)$

$$\Psi(x, m)_{ij} = m_{ij} \cdot x_{ij} + (1 - m_{ij}) \cdot \epsilon_{ij}$$

- ▶ Flouter avec un noyau gaussien

$$\Psi(x, m)_{ij} = \text{blur with kernel } g_{\sigma, m_{ij}}$$

Apprendre le floutage optimal

- ▶ Initialement, classe cible y a $f_y(\Psi(x, 1)) \approx 1$
- ▶ **But** : apprendre m telle que $f_y(\Psi(x, m)) \approx 0$
- ▶ Minimiser la loss suivante :

$$\min_m f_y(\Psi(x, m))$$

Autres considérations

1. Le flou doit être minimal
2. Le masque doit être lisse
3. L'optimisation doit être robuste à des perturbations adversariales

$$\min_m \mathbb{E}_\tau [f_y(\Psi(x(\cdot - \tau), m))] + \lambda_1 \|1 - m\|_1 + \lambda_2 \|\nabla m\|_\beta^\beta$$

premier terme : entrée ; terme du milieu : masque minimum ; dernier terme : masque lissé
 Déterminer le masque optimal par descente de gradient

$$m^+ = m - \alpha \cdot \frac{\partial L}{\partial m}(m)$$

Approches par perturbation

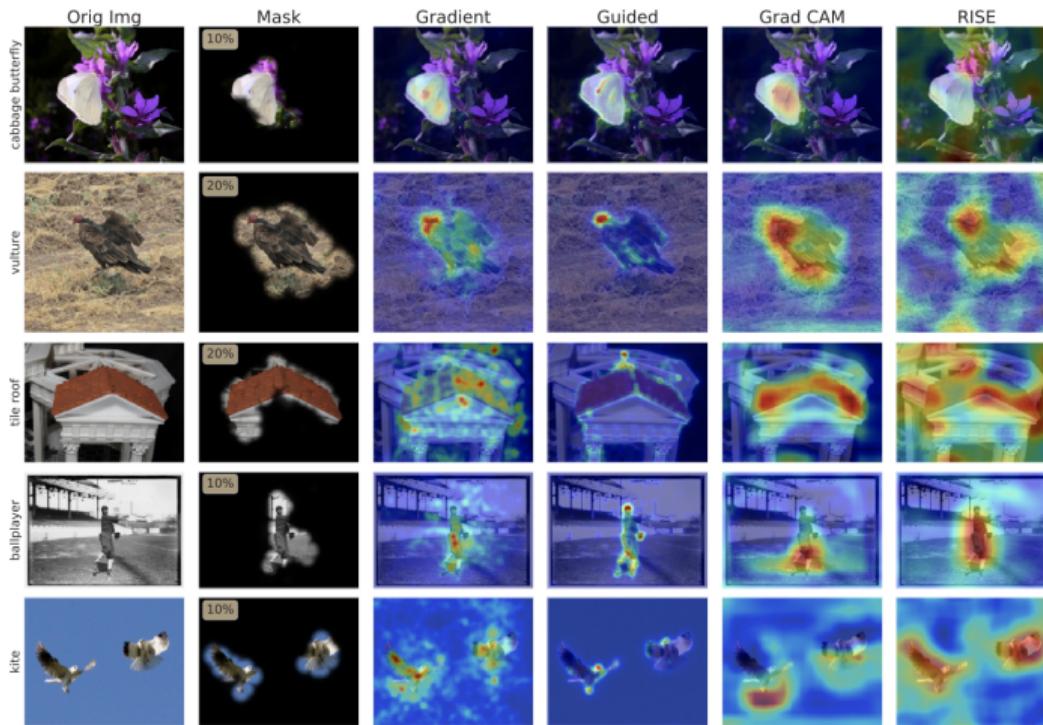
Re-visiter les cartes de saillance : suppression et préservation

- ▶ Étant donnée une image x_0 , l'objectif est d'avoir un *résumé* de l'effet de supprimer des parties de l'image pour le modèle pour expliquer le comportement du modèle, i.e. trouver les **régions à supprimer qui sont les plus informatives**
- ▶ Jeu de suppression : trouver le plus petit masque m qui impacte le plus la performance du modèle :

$$m^* = \arg \min_{m \in [0,1]^M} \lambda \|1 - m\|_1 + f_c(\Psi(x_0; m))$$

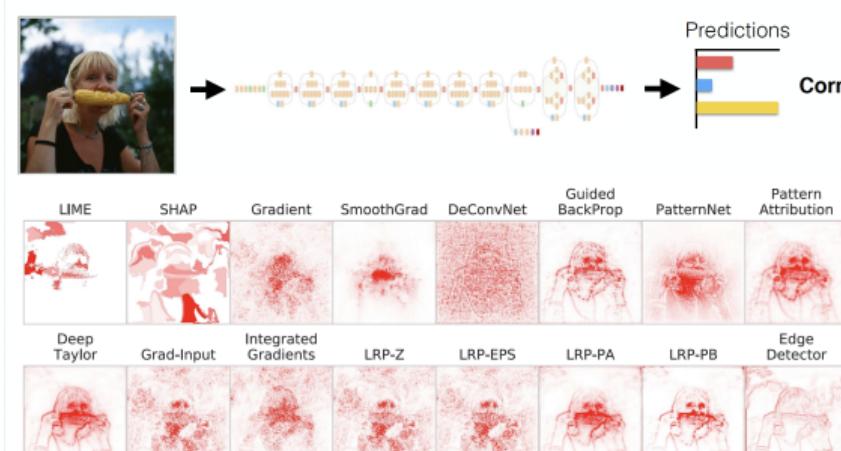
avec Ψ l'opérateur de perturbation.

Approches par perturbation



Contrôle (Sanity Checks) pour les cartes d'attribution

Méthodologie actionnable basée sur des tests de randomisation pour évaluer l'adéquation des approches d'explication.



Pour une tâche et un modèle particulier, comment un développeur/chercheur doit-il choisir la méthode à utiliser ?

(Adebayo et al, 2018) Sanity Checks for Saliency Maps¹⁷

17. <https://arxiv.org/abs/1810.03292>

Sanity Checks pour les cartes d'attribution

Propriétés souhaitées

- ▶ Sensibilité aux **paramètres** d'un modèle à expliquer.
- ▶ Dépendentes de l'étiquetage des données, c'est-à-dire qu'elles reflètent la relation entre les entrées et les sorties

Sanity Checks pour les cartes d'attribution

Principe

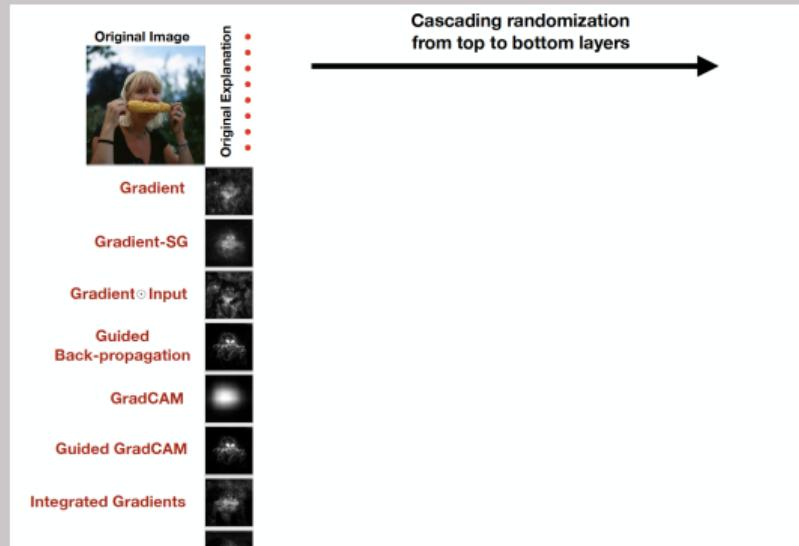
La randomisation comme moyen de tester les deux exigences

- ▶ **Test de randomisation des paramètres du modèle :** randomiser (réinitialiser) les paramètres d'un modèle et comparer les cartes d'attribution d'un modèle entraîné à celles dérivées d'un modèle randomisé.
- ▶ **test de randomisation des données :** comparer les cartes d'attribution d'un modèle entraîné avec des étiquettes correctes à celles dérivées d'un modèle entraîné avec des étiquettes aléatoires.

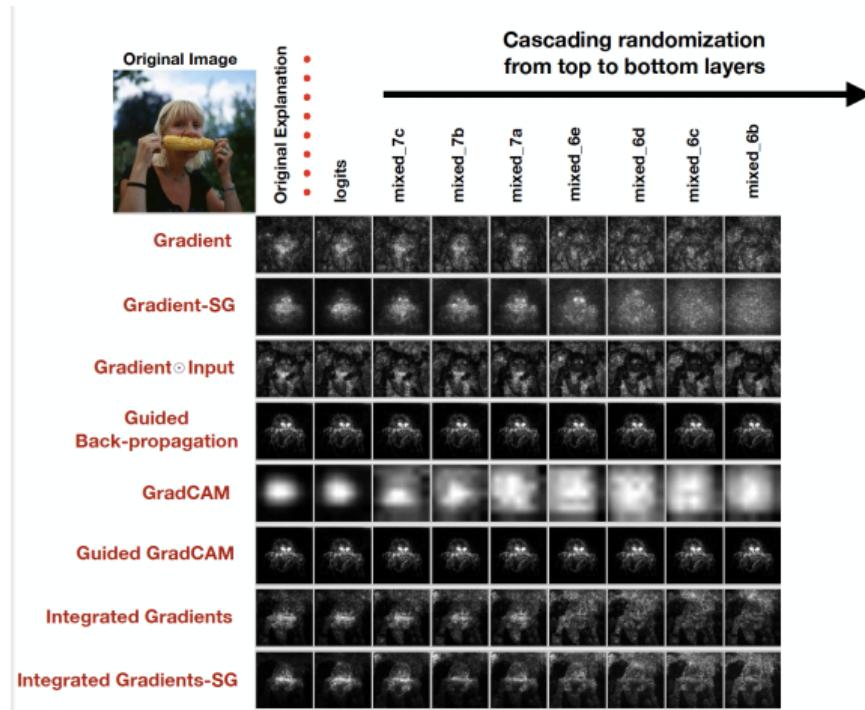
Sanity Checks pour les cartes d'attribution

andomisation des paramètres du modèle

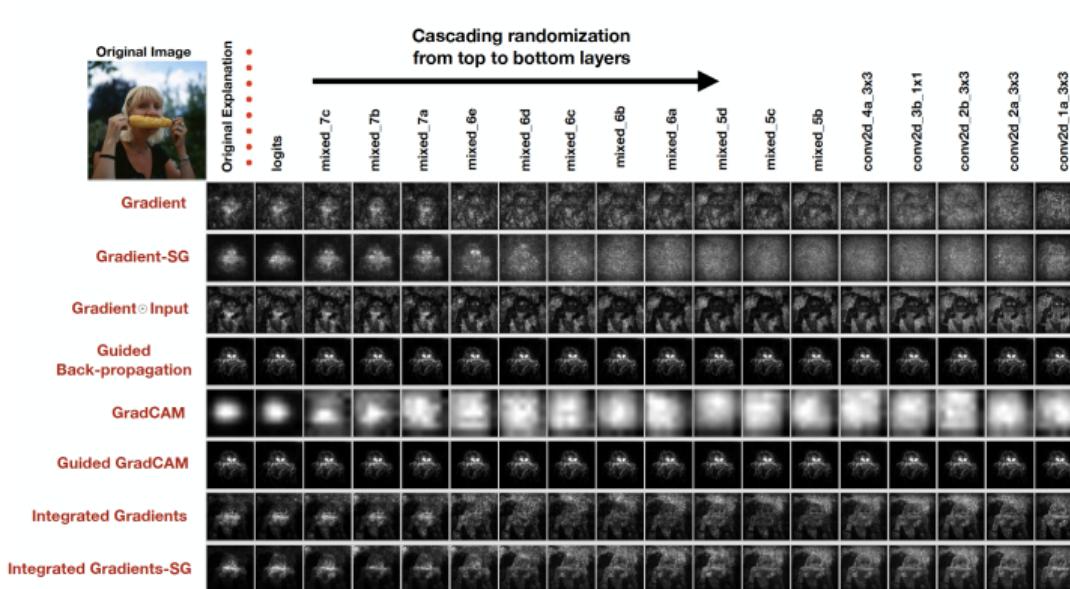
Conjoncture : Si un modèle capture des concepts de classe, les cartes de saillance devraient changer au fur et à mesure que le modèle est randomisé.



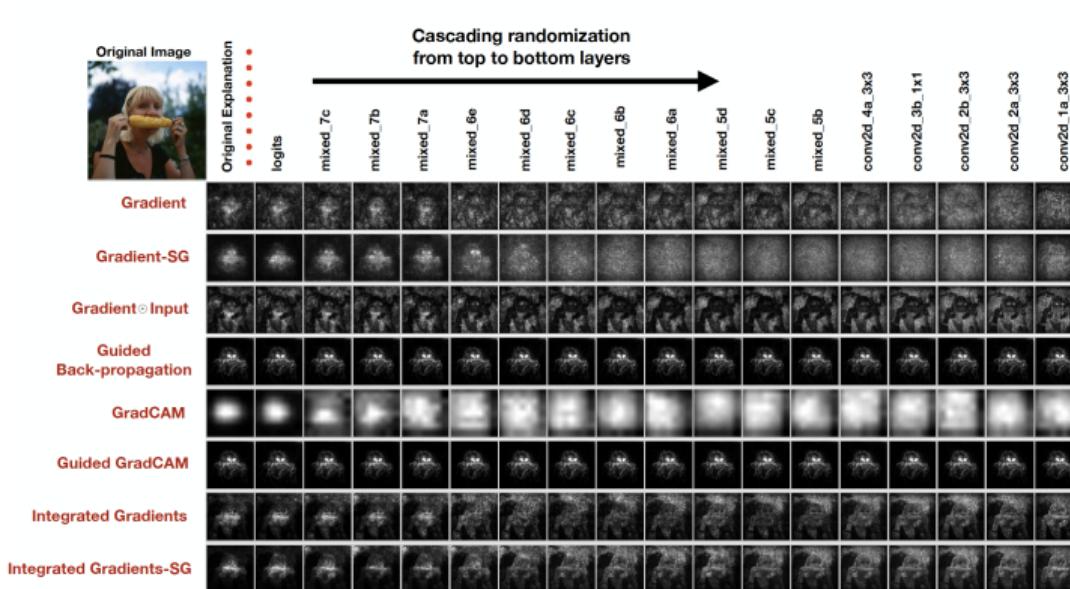
Sanity Checks pour les cartes d'attribution



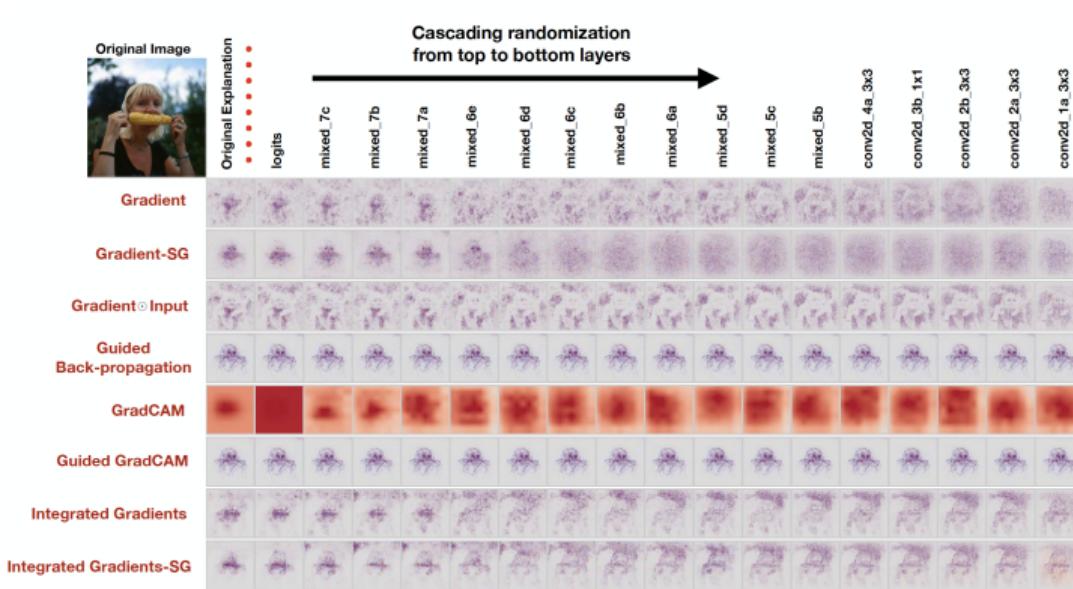
Sanity Checks pour les cartes d'attribution



Sanity Checks pour les cartes d'attribution

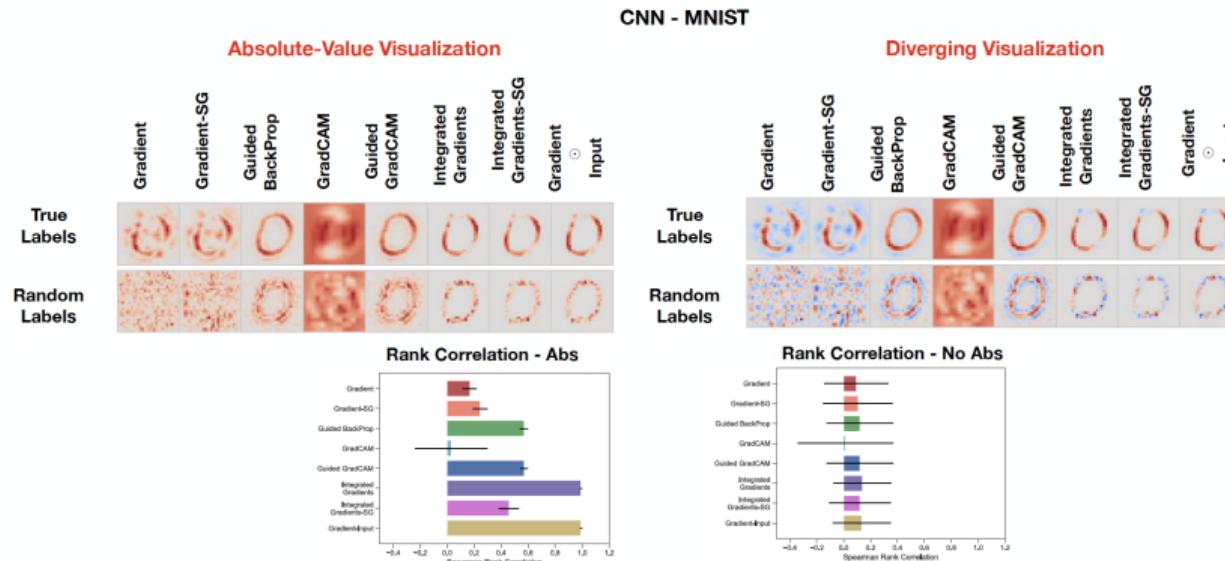


Sanity Checks pour les cartes d'attribution



Sanity Checks pour les cartes d'attribution

Randomisation des données :



Sanity Checks pour les cartes d'attribution

- ▶ Les contrôles d'intégrité ne permettent pas de savoir si une méthode est bonne, mais seulement si elle est invariante.
- ▶ Une simple inspection visuelle peut être trompeuse.

Explication basée concepts

Motivation

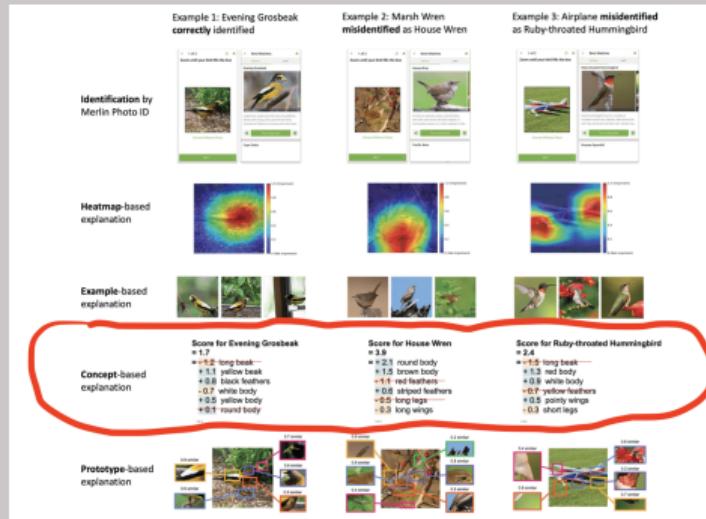


Figure – Source : Kim et al, 2022 - “Help Me Help the AI” : Understanding How Explainability Can Support Human-AI Interaction^b

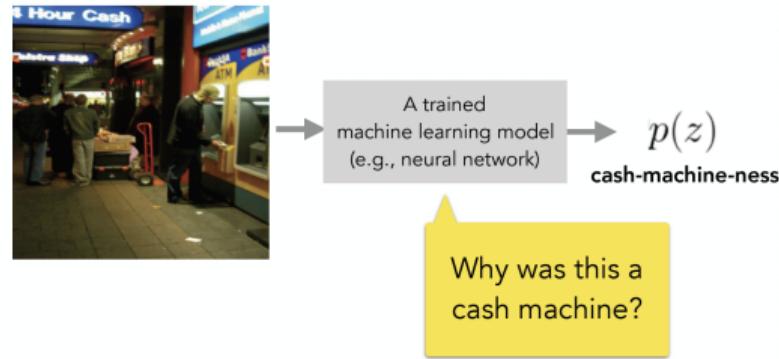
Principale motivation : **Explications centrées humain.**



Explication basée concepts : motivations

Exemple pris de Been Kim¹⁸

Problem: Post-training explanation



slides credit :

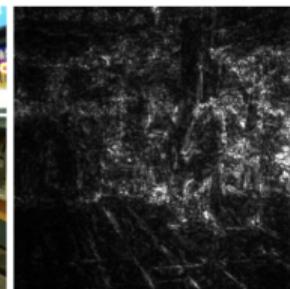
18. Kim et al - Interpretability beyond feature attribution : Testing with Concept Activation Vectors¹⁹

Explication basée concepts : motivations

Exemple pris de Been Kim²⁰



Caaaan do! we've got saliency maps to measure importance of each pixel!



a logit → $\frac{\partial p(z)}{\partial x_{i,j}}$
pixel i,j → $\frac{\partial p(z)}{\partial x_{i,j}}$

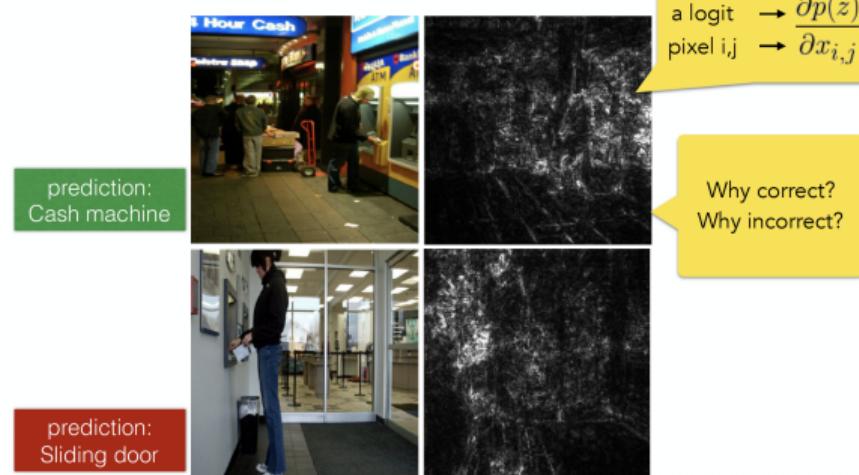
20. Kim et al - Interpretability beyond feature attribution : Testing with Concept Activation Vectors²¹

Explication basée concepts : motivations

Exemple pris de Been Kim²²

One of the most popular interpretability methods for images:

Saliency maps



22. Kim et al - Interpretability beyond feature attribution : Testing with Concept Activation Vectors²³

Explication basée concepts : motivations

Exemple pris de Been Kim²⁴

What we really want to ask...

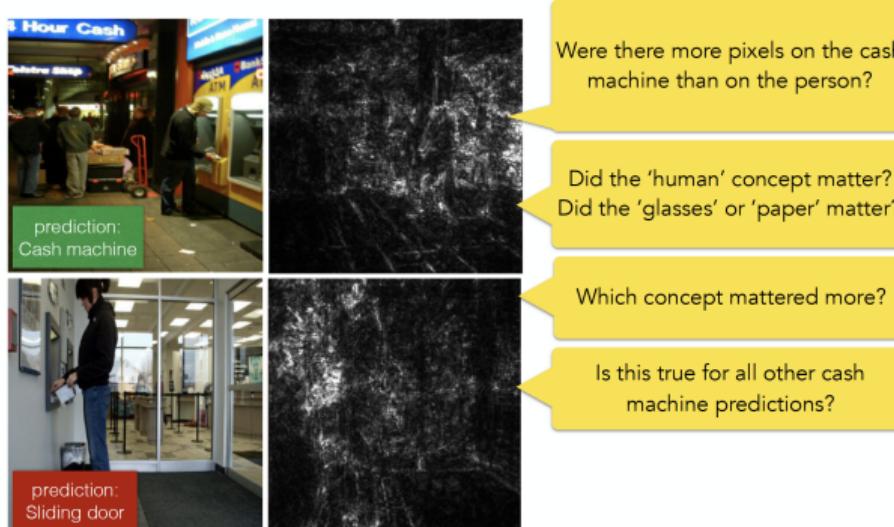


24. Kim et al - Interpretability beyond feature attribution : Testing with Concept Activation Vectors²⁵

Explication basée concepts : motivations

Exemple pris de Been Kim²⁶

What we really want to ask...

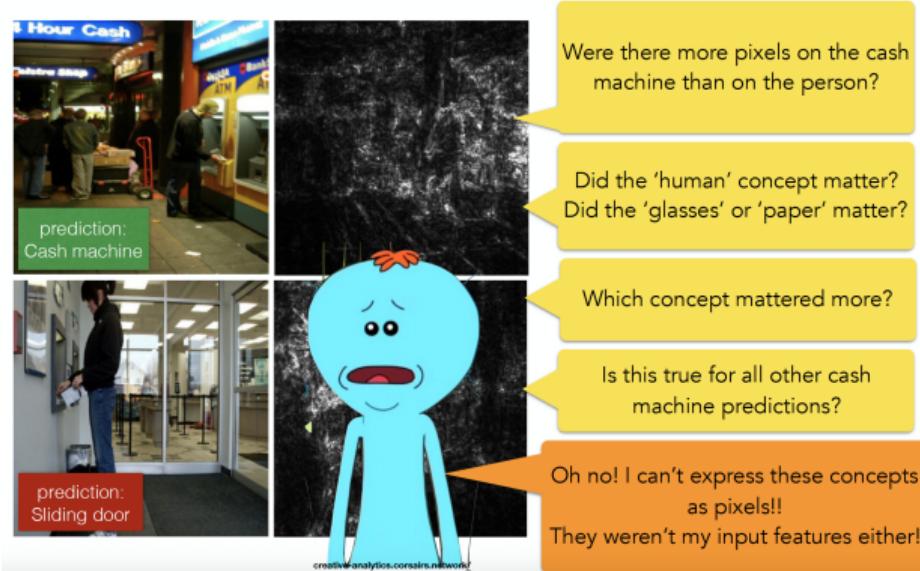


26. Kim et al - Interpretability beyond feature attribution : Testing with Concept Activation Vectors²⁷

Explication basée concepts : motivations

Exemple pris de Been Kim²⁸

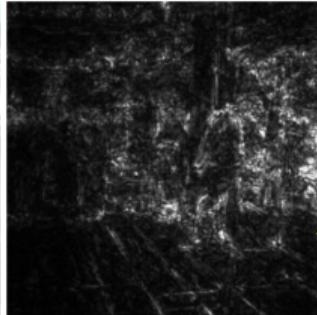
What we really want to ask...



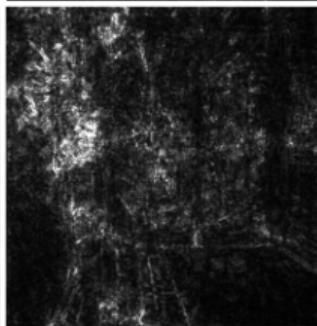
28. Kim et al - Interpretability beyond feature attribution : Testing with Concept Activation Vectors²⁹

Explication basée concepts : motivations

What we really want to ask...



Were there more pixels on the cash machine than on the person?



Did the 'human' concept matter?
Did the 'glasses' or 'paper' matter?

Which concept mattered more?

Is this true for all other cash machine predictions?

Wouldn't it be great if we can quantitatively measure how important *any* of these user-chosen concepts are?

Explication basée concepts : motivations

But

Fournir des explications sous forme de **concept humain de haut niveau** au lieu d'attribuer de l'importance à des caractéristiques ou des pixels individuels.

Propriétés voulues^a

a. (Ghorbani et al, 2019) Towards Automatic Concept-based Explanations
<https://arxiv.org/abs/1902.03129>

- ▶ **Significativité** : un exemple de concept est sémantiquement significatif en soi. (par exemple, pour les données image, les pixels individuels ne satisfont pas à cette propriété, mais un groupe de pixels peut être significatif).
- ▶ **Cohérence** : Les exemples d'un concept doivent être perceptuellement similaires les uns aux autres tout en étant différents des exemples d'autres concepts.
- ▶ **Importance** : Un concept est "important" pour la prédiction d'une classe si sa présence est nécessaire à la prédiction des échantillons de cette classe.

Explication basée concepts : les différentes approches

► **Explication en tant qu'ensemble d'éléments interprétables**

- (Zhou el al, 2018) Interpretable basis decomposition for visual explanation³⁰.
- Concept activation vectors : (Kim et al, 2018) Interpretability beyond feature attribution : Quantitative testing with concept activation vectors (tcav).³¹.

30. <https://people.csail.mit.edu/bzhou/publication/eccv18-IBD>

31. <https://arxiv.org/abs/1711.11279>

Décomposition en base interprétable d'une image

De l'attribution à l'interprétation : décomposer les preuves d'une prédiction en éléments sémantiquement interprétables.

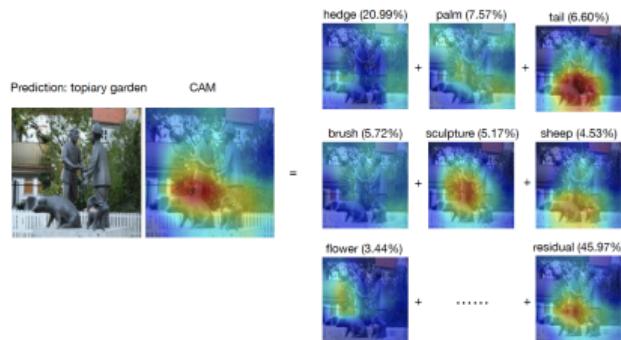


Fig. 1. Interpretable Basis Decomposition provides an explanation for a prediction by decomposing the decision into the components of interpretable basis. Top contributing components are shown with a label, contribution, and heatmap for each term.

(Zhou el al, 2018) Interpretable basis decomposition for visual explanation³².

32. url`https://people.csail.mit.edu/bzhou/publication/eccv18-IBD`

Décomposition en base interprétable d'une image

Idée : expliquer une couche en choisissant une base interprétable servant à la représenter.

Définition d'une base interprétable

- ▶ Nous voulons exprimer les propriétés de x qui déterminent le score $f_y(x)$ pour une classe particulière $y \in K$.
 - ▶ Le concept *foule de personnes* tend-il à faire classer une entrée comme *aéroport* ?
- ▶ On considère $f(x) = h(g(x))$ où $a = g(x)$ est la sortie de l'avant-dernière couche et $h(a)$ une simple opération linéaire effectuée par la dernière couche.

$$h(a) = W^{(h)} a + b^{(h)}$$

$$h_y(a) = w_y^T a + b_y$$

h_y est une fonction linéaire qui score a en fonction de l'angle entre a et w_y

- ▶ Expliquer w_y en la décomposant en une somme pondérée de composantes interprétables q_{c_i} avec q_{c_i} une direction dans l'espace de représentation qui correspond au concept interprétable élémentaire c_i .

$$w_y \approx s_{c_1} q_{c_1} + \dots + s_{c_n} q_{c_n}$$

- ▶ Trouver s_{c_i} pour minimiser $\|r\|$ où $w_y = s_{c_1} q_{c_1} + \dots + s_{c_n} q_{c_n} + r$ avec r l'erreur résiduelle dans la décomposition



Décomposition en base interprétable d'une image

Définition d'une base interprétable

- ▶ Les négations de concepts ne sont pas aussi compréhensibles que les concepts positifs : **rechercher une décomposition pour laquelle chaque coefficient $s_{ci} > 0$ est positif.**
- ▶ **Rechercher des décompositions avec un petit nombre de concepts.**
- ▶ Construction de la base de manière gloutonne : ajout incrémental de concepts pour réduire l'erreur résiduelle.
- ▶ On peut réduire l'erreur résiduelle en ajoutant un $(n + 1)$ -concept. Le meilleur concept est celui qui permet d'obtenir le résidu le plus faible tout en conservant un coefficient positif

$$\arg \min_{c \in C} \min_{s, s_i > 0} \|w_k - [C|q_c]s\|$$

avec $[C|q_c]$ la matrice qui ajoute le vecteur q_c pour le concept candidat c aux colonnes de C .

Décomposition en base interprétable d'une image

Apprendre la base interprétable à partir des annotations

- ▶ Broden (Broadly and Densely Labeled) dataset : Segmentations au niveau des pixels pour un grand nombre de concepts visuels de haut niveau, tels que les objets et les parties d'objets et de bas niveau, tels que les couleurs et les matériaux^a.
- ▶ L'univers des concepts candidats \mathcal{C} est construit à l'aide de l'ensemble de données Broden.
- ▶ Pour chaque concept c dans Broden, nous calculons un plongement q_c en utilisant :
 - ▶ un classificateur binaire logistique $h_c(a) = \text{sigmoïd}(w_c^T a + b_c)$ pour détecter la présence du concept c .
 - ▶ Entraînement sur un mélange d'images équilibrant c présent ou absent avec hard negative mining
 - ▶ $q_c = \frac{(w_c - \bar{w}_c)}{\|w_c - \bar{w}_c\|}$ (normalisation)

a. <https://github.com/CSAILVision/NetDissect>



Décomposition en base interprétable d'une image

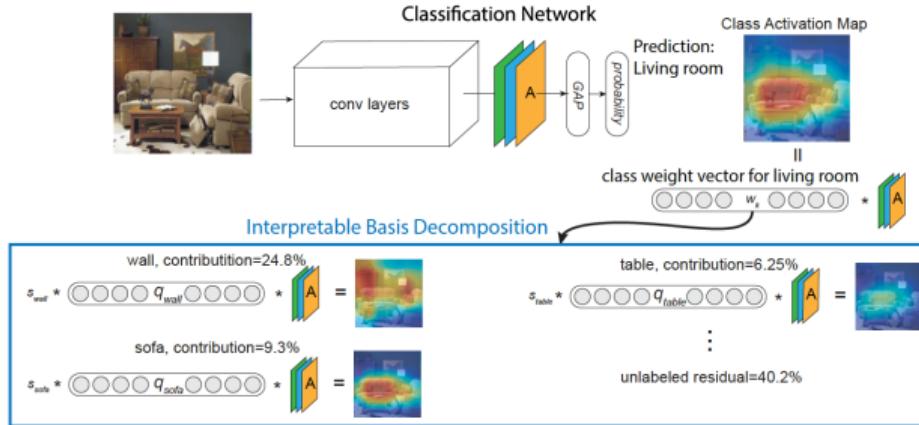
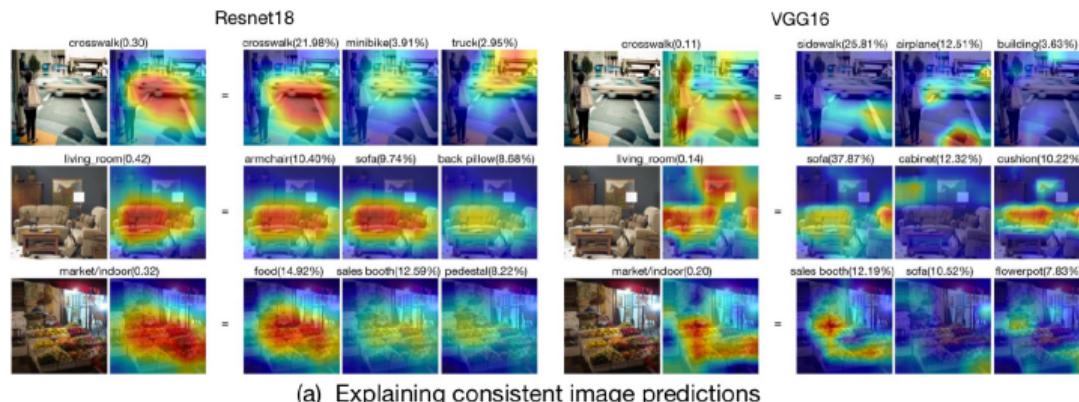


Fig. 2. Illustration of Interpretable Basis Decomposition. The class weight vector w_k is decomposed to a set of interpretable basis vectors $\sum s_{c_i} q_{c_i}$, each corresponding to a labeled concept c_i as well as a projection $q_{c_i}^T A$ that reveals a heatmap of the activations. An explanation of the prediction k consists of the concept labels c_i and the corresponding heatmaps for the most significant terms in the decomposition of $w_k^T a$. For this particular example, wall, sofa, table (and some others are not shown) are labels of the top contributing basis elements that make up the prediction of living room.

Décomposition en base interprétable d'une image

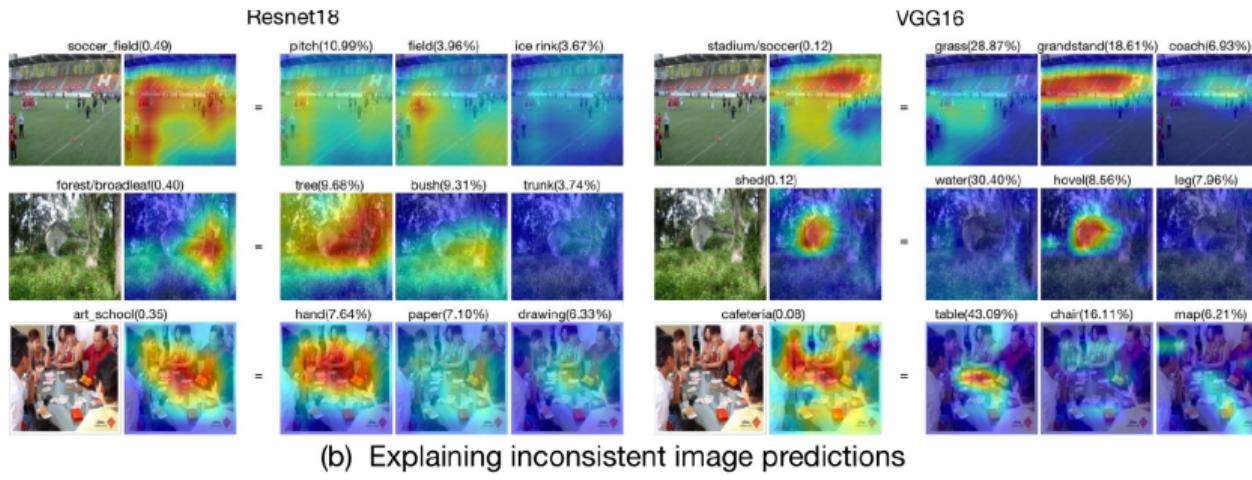
Explication d'une prédiction par la décomposition dans la base interprétable

- L'explication consiste en la liste des concepts c_i ayant les plus grandes contributions à $h_y(a)$ avec les cartes de saillance $q_{c_i}^T A$ pour chaque concept.



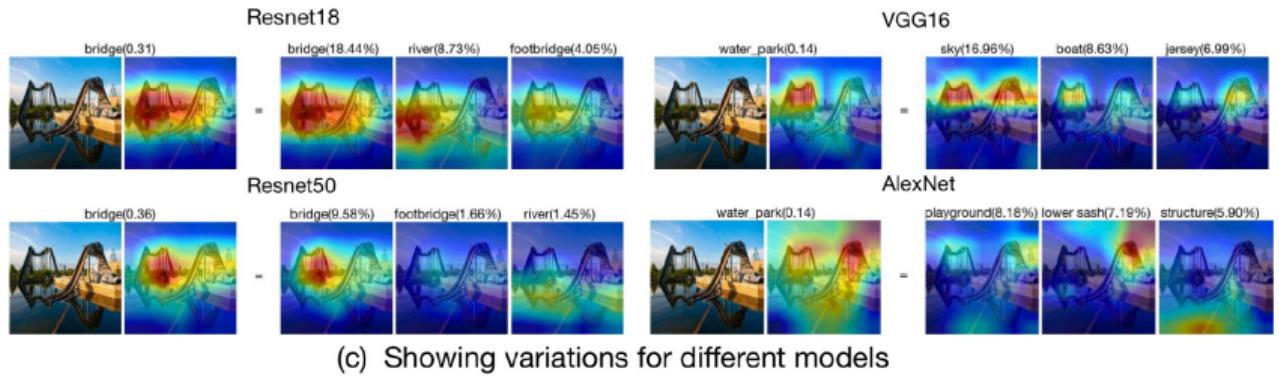
Dans ces deux exemples, les explications fournissent la preuve que la fiabilité de VGG16 peut être discutée dans certains cas : concept d'avion pour expliquer passage pour piétons ; concept de canapé pour la prédiction de marché. En revanche, ResNet18 semble être plus fiable.

Décomposition en base interprétable d'une image



ResNet18 classe l'image de la dernière ligne comme une école d'art parce qu'il voit des éléments décrits comme des mains, du papier et des dessins, tandis que VGG16 classe l'image comme une image de cafétéria parce que VGG16 est sensible aux caractéristiques de table et de chaise. Les deux réseaux sont incorrects car la table est recouverte de cartes à jouer, et non de dessins ou de cartes. L'étiquette correcte est la salle de jeux.

Décomposition en base interprétable d'une image



Décomposition en base interprétable d'une image

Explication de la limite de décision de la classification

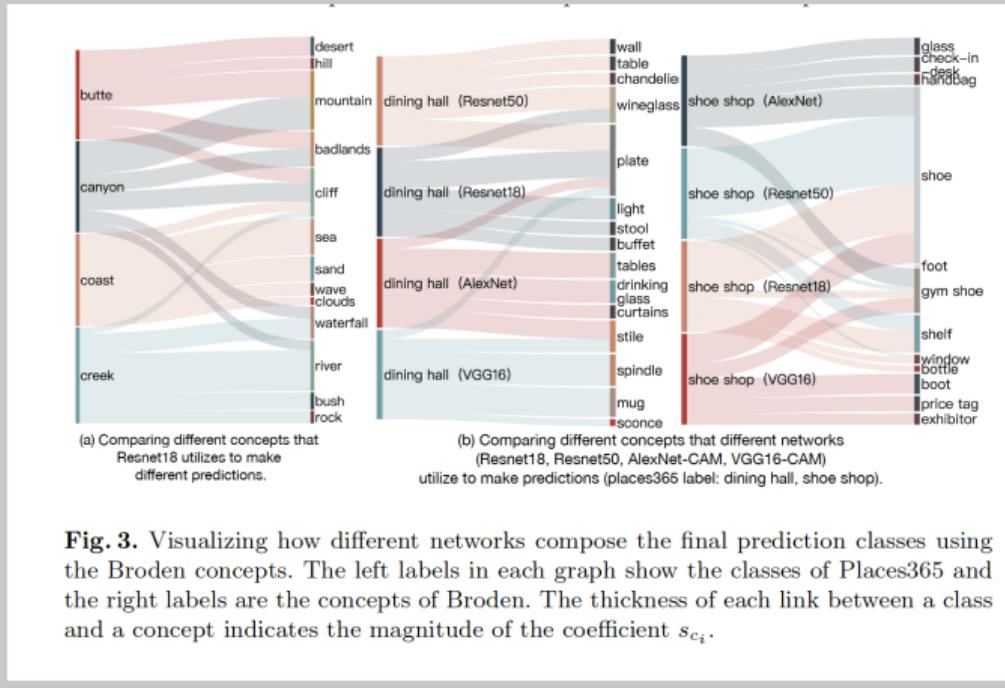
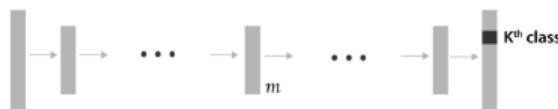


Fig. 3. Visualizing how different networks compose the final prediction classes using the Broden concepts. The left labels in each graph show the classes of Places365 and the right labels are the concepts of Broden. The thickness of each link between a class and a concept indicates the magnitude of the coefficient s_{ci} .

Autre approche : Concept Activation Vector

Kim et al - Interpretability beyond feature attribution : Testing with Concept Activation Vectors³³

Goal of TCAV:
Testing with Concept Activation Vectors



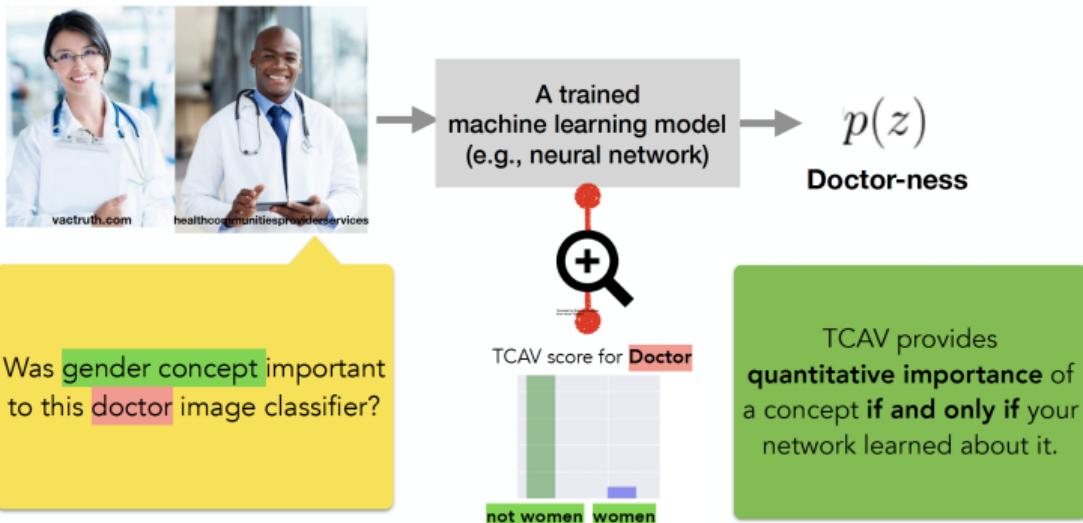
Quantitative explanation: how much a concept (e.g., gender, race) was important for a prediction in a trained model.

...even if the concept was not part of the training.

33. <https://arxiv.org/abs/1711.11279>

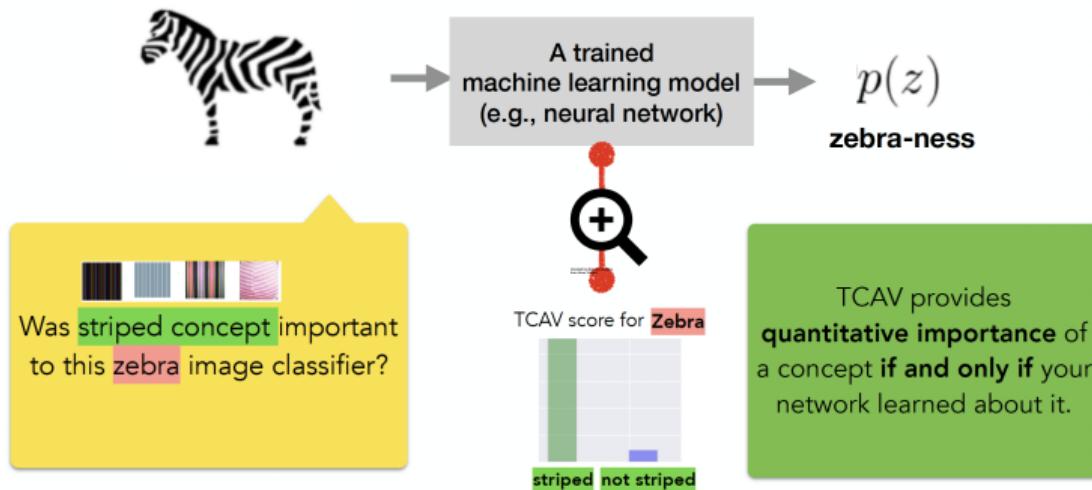
Concept Activation Vector : objectifs

Goal of TCAV: Testing with Concept Activation Vectors



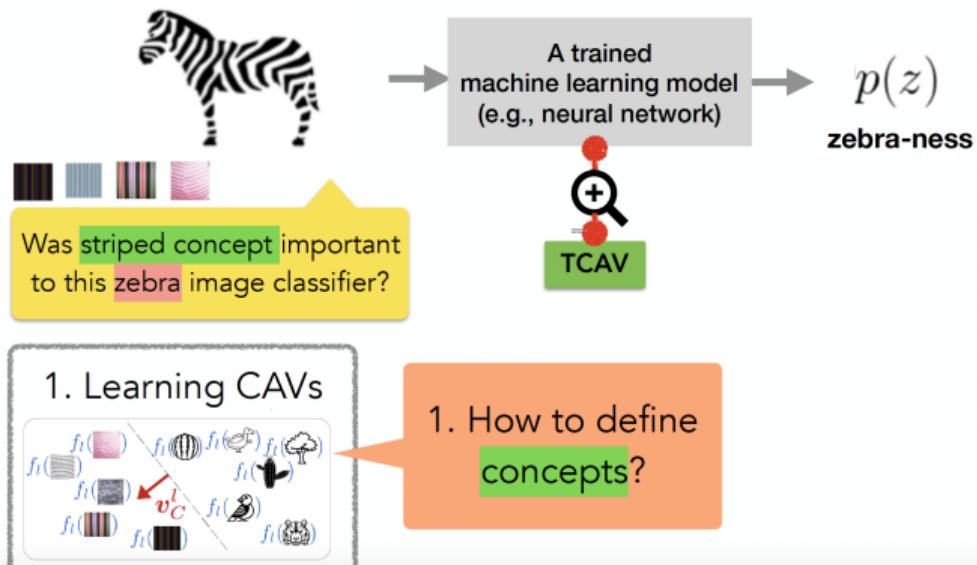
Concept Activation Vector : objectifs

Goal of TCAV: Testing with Concept Activation Vectors



Concept Activation Vector : apprentissage

TCAV: Testing with Concept Activation Vectors



Concept Activation Vector : apprentissage

- ▶ Idée principale : **interprétabilité linéaire** - les combinaisons linéaires de neurones ou d'activations peuvent coder des informations significatives et pertinentes.
- ▶ Étant donné un ensemble d'exemples représentant un concept d'intérêt humain, nous recherchons un vecteur dans l'espace des activations de la couche / qui représente ce concept.
- ▶ **Vecteur d'activation du concept** : vecteur normal à un hyperplan séparant les activations dans la couche / des exemples sans concept et des exemples avec un concept.

Concept Activation Vector : apprentissage

Defining concept activation vector (CAV)

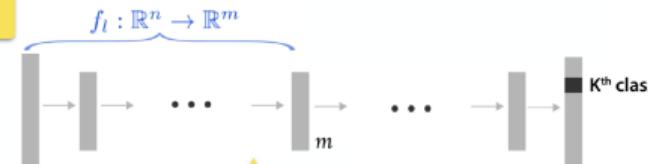
Inputs:

a



Examples of
concepts

Random
images

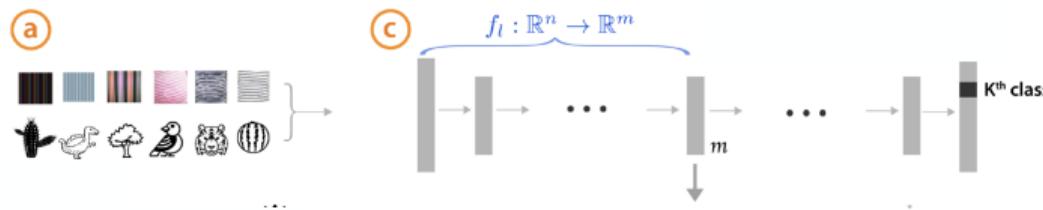


A trained network under investigation
and
Internal tensors

Concept Activation Vector : apprentissage

Defining concept activation vector (CAV)

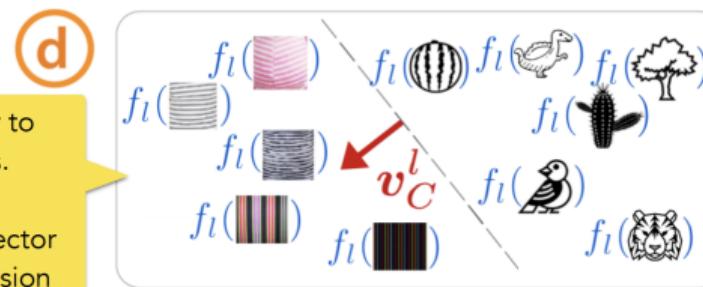
Inputs:



Train a linear classifier to separate activations.

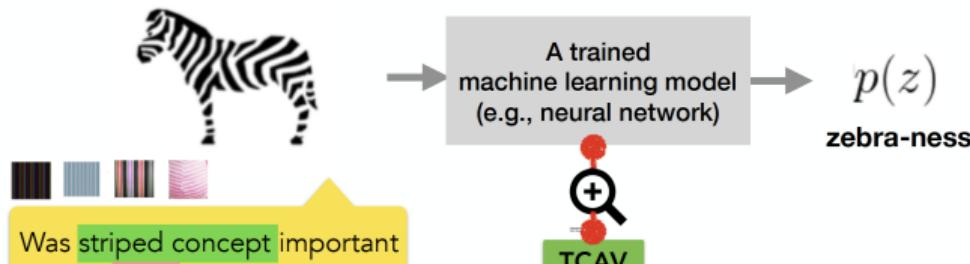
CAV (v_C^l) is the vector **orthogonal** to the decision boundary.

[Smilkov '17, Bolukbasi '16, Schmidt '15]

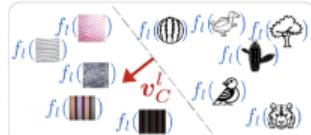


Concept Activation Vector : sensibilité conceptuelle

TCAV: Testing with Concept Activation Vectors



1. Learning CAVs



2. Getting TCAV score

$$S_{C,k,l}(\text{zebra})$$

$$S_{C,k,l}(\text{owl})$$

$$S_{C,k,l}(\text{zebra})$$

$$\rightarrow \text{TCAV}_{Q_{C,k,l}}$$

2. How are the CAVs useful to get explanations?

Concept Activation Vector : sensibilité conceptuelle

Conceptual sensitivity

- ▶ Récap : cartes de saillance - Utilisation des gradients des valeurs logit par rapport à des caractéristiques d'entrée individuelles pour évaluer la sensibilité.

$$\frac{\partial h_k(x)}{\partial x_{a,b}}$$

où $h_k(x)$ logit pour une donnée x pour une classe k et $x_{a,b}$ pixel à la position (a, b) dans x

- ▶ La sensibilité conceptuelle de la classe k au concept C et à la couche l est calculée comme la dérivée directionnelle $S_{C,k,l}(x)$:

$$S_{C,k,l}(x) = \lim_{\epsilon \rightarrow 0} \frac{h_{l,k}(f_l(x) + \epsilon v_C^l) - h_{l,k}(f_l(x))}{\epsilon}$$

$$S_{C,k,l}(x) = \nabla h_{l,k}(f_l(x)).v_C^l$$

Mesure quantitative de la sensibilité des prévisions du modèle par rapport aux concepts pour n'importe quelle couche du modèle

Concept Activation Vector : sensibilité conceptuelle

Test avec CAVs

- ▶ TACV : Testing with CAVs - Calcul de la sensibilité conceptuelle du modèle ML sur des classes entières d'entrées. Pour une classe k , avec X_k toutes les entrées avec cette étiquette donnée, le score TCAV est

$$TCAV_{C,k,I} = \frac{|\{x \in X_k : S_{C,k,I}(x) > 0\}|}{|X_k|}$$

Fraction des entrées avec I – le vecteur d'activation de la couche I positivement influencées par le concept C

- ▶ $TCAV_{C,k,I}$ dépend seulement du signe de $S_{C,k,I}$.
- ▶ Différentes mesures pourraient être utilisées pour évaluer la grandeur de la sensibilité conceptuelle.

Concept Activation Vector : sensibilité conceptuelle

TCAV core idea:
Derivative with CAV to get prediction sensitivity

TCAV



$$\begin{aligned} S_{C,k,l}(\text{zebra}) \\ S_{C,k,l}(\text{dotted}) \\ S_{C,k,l}(\text{striped}) \\ S_{C,k,l}(\text{zig-zagged}) \end{aligned} \quad \left. \right\}$$

zebra-ness $\rightarrow \frac{\partial p(z)}{\partial \mathbf{v}_C^l} = S_{C,k,l}(\mathbf{x})$

striped CAV $\rightarrow \frac{\partial}{\partial \mathbf{v}_C^l} = S_{C,k,l}(\mathbf{x})$

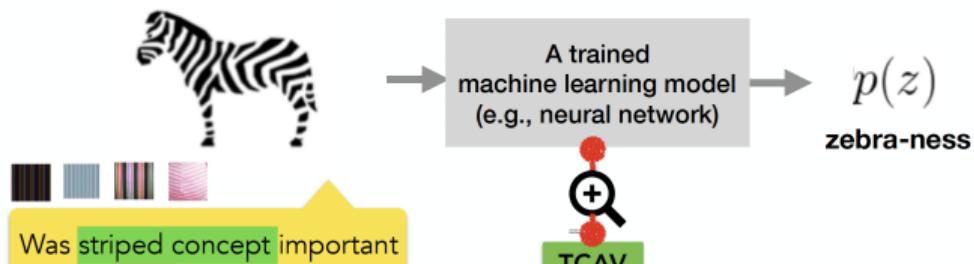
$$\text{TCAV}_{Q_{C,k,l}} = \frac{|\{x \in X_k : S_{C,k,l}(x) > 0\}|}{|X_k|}$$

Directional derivative with CAV

Concept Activation Vector :validation

TCAV:

Testing with Concept Activation Vectors



1. Learning CAVs



2. Getting TCAV score

$$S_{C,k,l}(\text{zebra})$$

$$S_{C,k,l}(\text{blue})$$

$$S_{C,k,l}(\text{pink})$$

$$S_{C,k,l}(\text{wavy})$$

} $\rightarrow \text{TCAV}_{Q_{C,k,l}}$

3. CAV validation

Qualitative
Quantitative

Concept Activation Vector :validation

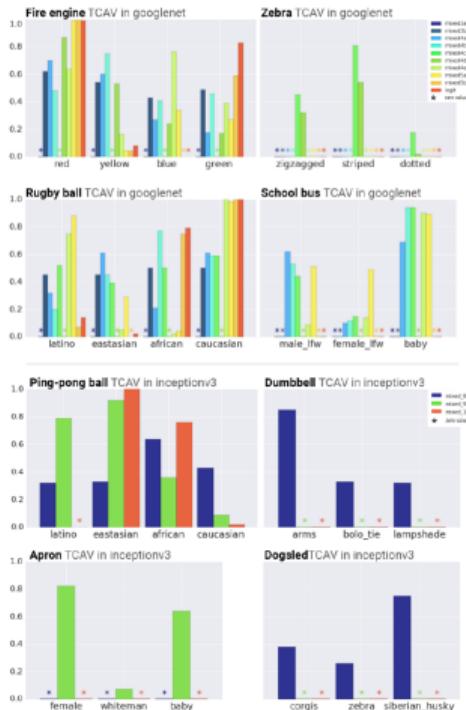


Figure 4. Relative TCAV for all layers in GoogleNet (Szegedy et al., 2015) and last three layers in Inception V3 (Szegedy et al., 2016) for confirmation (e.g., fire engine), discovering biases (e.g., rugby, apron), and quantitative confirmation for previously qualitative findings in (Mordvintsev et al., 2015; Stock & Cisse, 2017) (e.g., dumbbell, ping-pong ball). TCAVs in layers close to the logit layer (red) represent more direct influence on the prediction than lower layers. *'s mark CAVs omitted after statistical testing.

Concept Activation Vector :validation

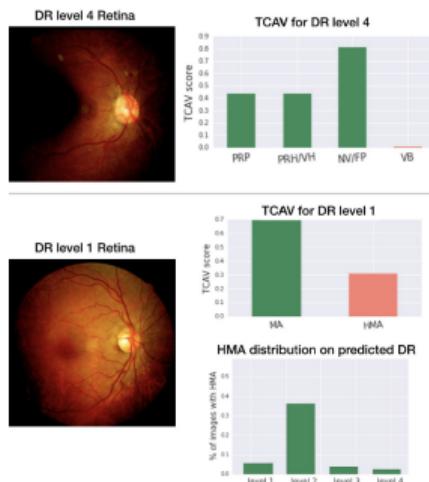


Figure 10. Top: A DR level 4 image and TCAV results. TCAVQ is high for features relevant for this level (green), and low for an irrelevant concept (red). Middle: DR level 1 (mild) TCAV results. The model often incorrectly predicts level 1 as level 2, a model error that could be made more interpretable using TCAV: TCAVQs on concepts typically related to level 1 (green, MA) are high in addition to level 2-related concepts (red, HMA). Bottom: the HMA feature appears more frequently in DR level 2 than DR level 1.

Concept Activation Vector :Avantages

- ▶ Une méthode globale d'explicabilité pour interpréter les états internes d'un modèle de boîte noire (modèle profond) dans un domaine conceptuel.
- ▶ Concepts compréhensibles par l'homme
- ▶ De nombreuses extensions :
 - ▶ ACE (Automatic Concept-based Explanations)³⁴ : fournit automatiquement des explications sur les concepts sans étiquetage humain, par k-means des activations des DNN
 - ▶ Causal Concept Effet³⁵ : effet causal (de la présence ou de l'absence) d'un concept interprétable par l'homme sur les prédictions d'un réseau neuronal profond.
 - ▶ ConceptSHAP³⁶ : se concentre sur la notion de complétude, qui quantifie la suffisance d'un ensemble particulier de concepts pour expliquer le comportement de prédiction d'un modèle.

34. <https://arxiv.org/pdf/1902.03129.pdf>

35. <https://arxiv.org/pdf/1907.07165.pdf>

36. <https://openreview.net/pdf?id=BylWYC4KwH>

Concept Activation Vector :Inconvénients

- ▶ Peut générer des CAVs sans signification si les concepts d'entrée ne sont pas sélectionnés correctement.
- ▶ Sujet à un biais humain dans la sélection des concepts.

Modèles interprétables par design

Modèles interprétables par design

Concept Bottleneck models

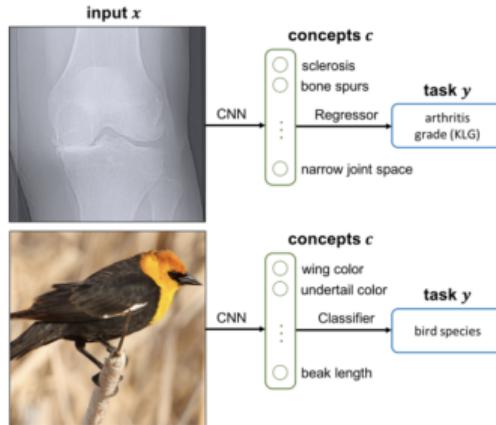


Figure 1. We study concept bottleneck models that first predict an intermediate set of human-specified concepts c , then use c to predict the final output y . We illustrate the two applications we consider: knee x-ray grading and bird identification.

[Koh et al, 2020] Concept Bottleneck Models³⁷

37. <https://arxiv.org/pdf/2007.04612.pdf>

Concept Bottleneck models

Principe

- ▶ Les Bottleneck models sont de la forme $f(g(x))$ où $g : \mathbb{R}^d \rightarrow \mathbb{R}^k$ fait correspondre un espace d'entrée x à l'espace conceptuel et $f : \mathbb{R}^k \rightarrow \mathbb{R}$ fait correspondre les concepts à la prédiction finale (ici une régression).
- ▶ Précision de la tâche : comment $f(g(x))$ prédit y .
 - ▶ $L_{C_j} : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}_+$ une fonction de perte qui mesure l'écart entre le concept prédit et le vrai j -th concept.
- ▶ Précision du concept : précision avec laquelle $g(x)$ prédit c (moyenne pour chaque concept).
 - ▶ $L_Y : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}_+$ une fonction de perte qui mesure l'écart entre la cible prédite et la cible réelle.

Concept Bottleneck models

Schémas d'apprentissage

- ▶ **Independent bottleneck** : \hat{f} et \hat{g} sont appris indépendamment

$$\hat{f} = \arg \min_f \sum_i L_Y(f(c^{(i)}); y^{(i)})$$

$$\hat{g} = \arg \min_g \sum_{i,j} L_{C_j}(g_j(x^{(i)}); c_j^{(i)})$$

\hat{f} est appris en utilisant c . A test time, prendre $\hat{g}(x)$ comme entrée

- ▶ **Sequential bottleneck** : apprend d'abord \hat{g} de la même manière que ci-dessus et utilise ensuite les prédictions conceptuelles $\hat{g}(x)$ pour apprendre

$$\hat{f} = \arg \min_f \sum_i L_Y(f(\hat{g}(x^{(i)}); y^{(i)})$$

- ▶ **Joint bottleneck** : minimise la somme pondérée

$$\hat{f}, \hat{g} = \arg \min_{f,g} \sum_i [L_Y(f(c^{(i)}); y^{(i)}) + \sum_j \lambda L_{C_j}(g(x^{(i)}); c_j^{(i)})]$$

Concept Bottleneck models

Test-time intervention

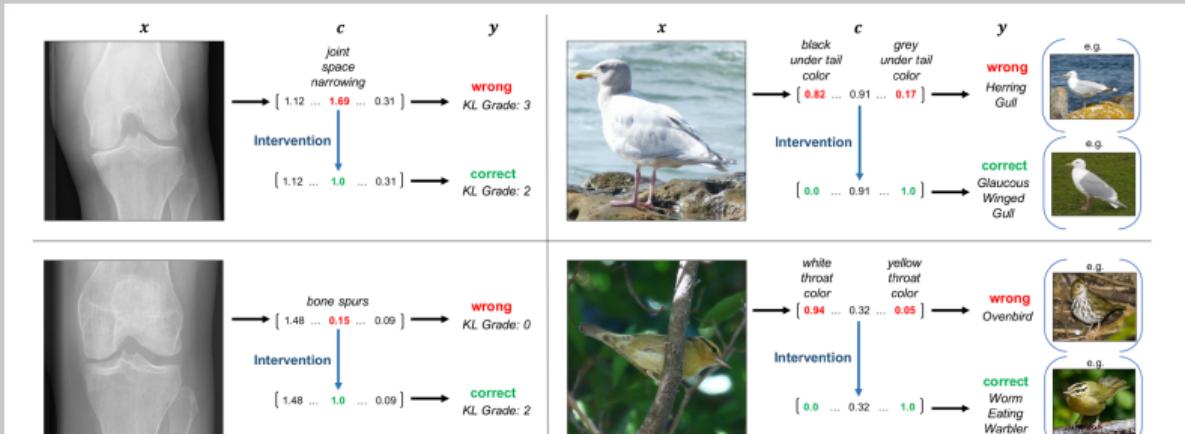


Figure 3. Successful examples of test-time intervention, where intervening on a single concept corrects the model prediction. Here, we show examples from independent bottleneck models. **Right:** For CUB, we intervene on concept groups instead of individual binary concepts. The sample birds on the right illustrate how the intervened concept distinguishes between the original and new predictions.

ProtoPNet

Prototypical part network

Le réseau dissèque l'image en trouvant des parties prototypiques et combine les preuves provenant des prototypes pour établir une classification finale.

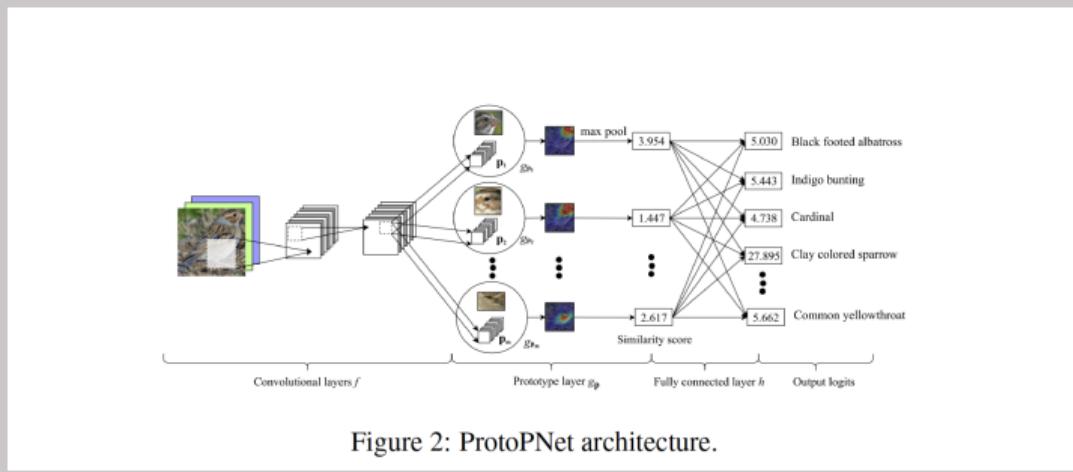


Figure 2: ProtoPNet architecture.

[Chen et al, 2021] This Looks Like That : Deep Learning for Interpretable Image Recognition³⁸

38. <https://arxiv.org/pdf/1806.10574.pdf>

ProtoPNet

Principe d'apprentissage

L'entraînement de ProtoPNet se divise en deux parties :

1. stochastic gradient descent (SGD) de couches avant la dernière couche
2. projection des prototypes
3. optimisation convexe de la dernière couche

Self explaining neural networks

SENN

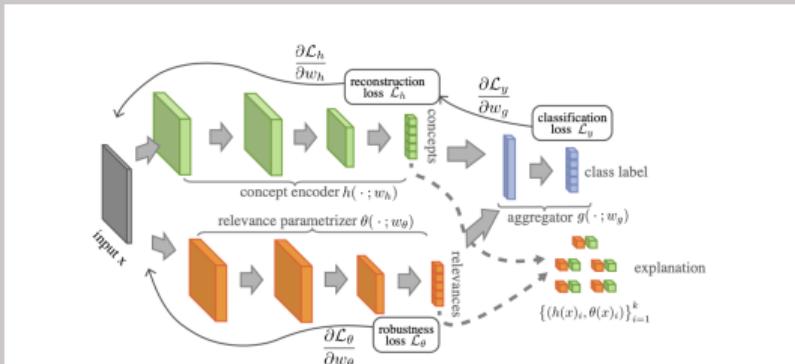


Figure 1: The proposed architecture consists of three components: a concept encoder (green) that transforms the input into a small set of interpretable basis features; an input-dependent parametrizer (orange) that generates relevance scores; and an aggregation function that combines them. Concepts and their relevance parameters are used by the aggregation function to produce the final label prediction. The robustness loss on the parametrizer encourages f to behave locally as a linear function on $h(x)$ with parameters $\theta(x)$, allowing for immediate interpretation of both both concepts and relevances.

Modèles interprétables par design

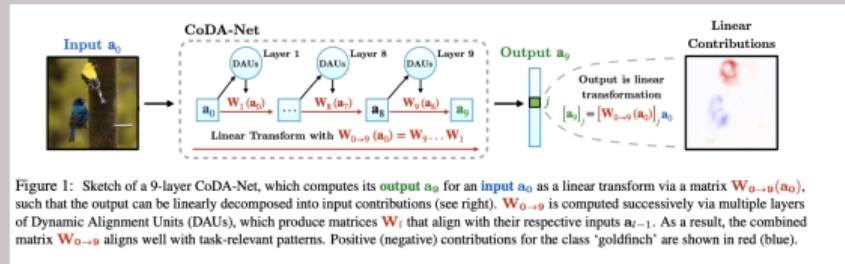
- ▶ (Koh et al, 2020) Concept Bottleneck Models³⁹
- ▶ (Alvarez melis et al, 2021) Self explaining models⁴⁰
- ▶ ...

39. url<https://arxiv.org/pdf/2007.04612.pdf>

40. <https://people.csail.mit.edu/davidam/docs/SENN.pdf>

De nombreuses autres approches

coDA-Nets



[Böhle et al , 2021] Convolutional Dynamic Alignment Networks for Interpretable Classifications⁴¹

[Böhle et al , 2022] B-cos Networks : Alignment is All We Need for Interpretability⁴²

41. <https://arxiv.org/pdf/2104.00032.pdf>

42. <https://arxiv.org/pdf/2205.10268.pdf>

Sommaire

Introduction

- Constat et motivations
- Qu'est ce que l'IA eXplicable ?
- Terminologie et définitions
- Principales approches

Modèles explicables par conception (transparents)

- Modèles de régression
- Generalised Additive Models (GAM)
- Modèles à base d'arbres

Explications post-hoc

- Méthodes indépendantes du modèle
- Explicabilité et réseaux de neurones profonds

Conclusion et questions ouvertes

Conclusion

- ▶ XAI n'est pas un nouveau problème, mais une reformulation d'un problème déjà abordé
- ▶ Domaine multidisciplinaire : IA, IHM, sciences sociales, sciences cognitives
- ▶ Crucial pour l'application de l'IA dans des systèmes critiques

Conclusion

- ▶ XAI n'est pas un nouveau problème, mais une reformulation d'un problème déjà abordé
 - ▶ Domaine multidisciplinaire : IA, IHM, sciences sociales, sciences cognitives
 - ▶ Crucial pour l'application de l'IA dans des systèmes critiques
-
- ▶ Pas de consensus sur les définitions de base
 - ▶ Pas de formalisme pour représenter une définition
 - ▶ Pas de méthodes pour quantifier l'interprétabilité, la compréhensibilité des explications, etc.
 - ▶ L'évaluation : pas de méthodes systématiques, il manque des données pour faire des benchmarks,...

Quelques outils I

- ▶ LIME
 - ▶ <https://github.com/marcotcr/lime>
- ▶ SHAP
 - ▶ <https://shap.readthedocs.io/en/latest/>
- ▶ AI Explainability 360 toolkit : intrinsic, post-hoc, local and global explainers.
 - ▶ <https://github.com/Trusted-AI/AIX360>
- ▶ Shapash : Python library which aims to make machine learning interpretable and understandable by everyone, focus on visualizations.
 - ▶ <https://github.com/MAIF/shapash>
- ▶ Google Pair : People + AI Guidebook with tools such as What-If Tool, Facets,LIT...
 - ▶ <https://pair.withgoogle.com/tools/>
- ▶ InterpretML : intrinsic and post-hoc methods for Python and R.
 - ▶ <https://github.com/interpretml/interpret>
- ▶ Captum : library built for PyTorch models, focus on attribution methods.
 - ▶ <https://github.com/pytorch/captum>

Quelques outils II

- ▶ DALEX : post-hoc and model-agnostic explainers that allow local and global explanations.
 - ▶ <https://github.com/ModelOriented/DALEX>
- ▶ Fat Forensics : Python toolkit for evaluating Fairness, Accountability and Transparency of Artificial Intelligence systems.
 - ▶ <https://fat-forensics.org/>

Contacts

- ▶ C. Hodelot : celine.hodelot@centralesupelec.fr
- ▶ W. Ouerdane : wassila.ouerdane@centralesupelec.fr
- ▶ J-P Poli : jean-philippe.poli@cea.fr

Inscrivez vous au GT EXPLICON (GDR RADIA) : <https://gt-explicon.github.io>