www.cnrs.fr

Explainable machine learning: Motivation and main strategies

Laurent Risser

Institut de Mathématiques de Toulouse (UMR 5219)

Artificial and Natural Intelligence Toulouse Institute (3IA ANITI)

Introduction

Machine learning models are functions with parameters optimised on a training set:

Input variables X_i | **Trained black-box model** f_θ | **Prediction** $\hat{Y}_i = f_\theta(X_i)$

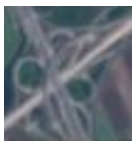
Examples:

Bank loan:

Bank profile

Access or not to the loan

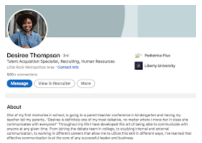
Shape detection:



Remote sensing image

Highway detected

HR pre-screening:

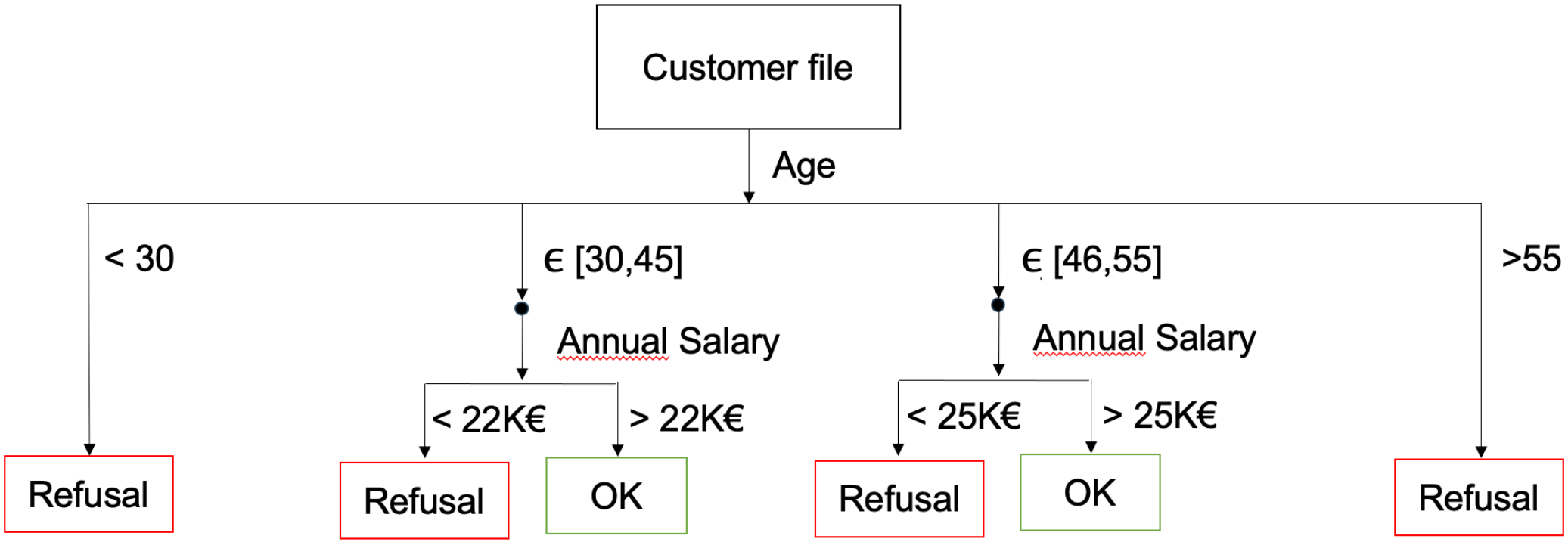


Profile out of LinkedIn

Interesting profile or not

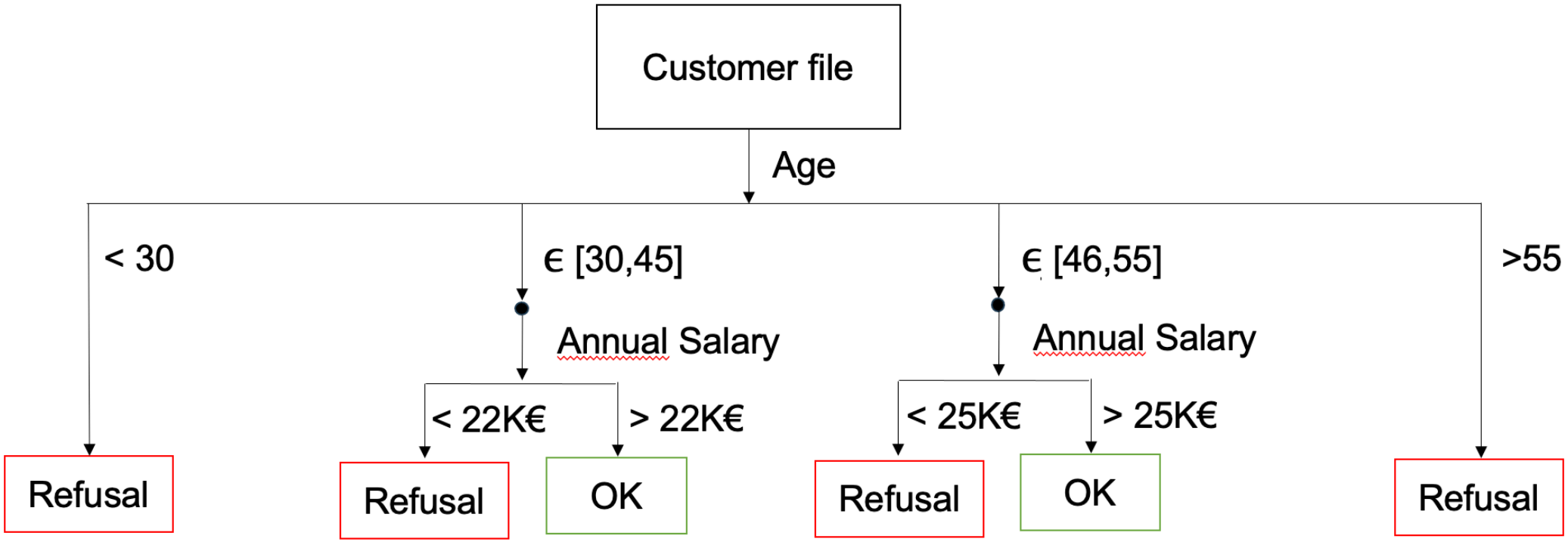
Introduction

Model for granting or not granting a bank loan (purely imaginary):



Introduction

Model for granting or not granting a bank loan (purely imaginary):

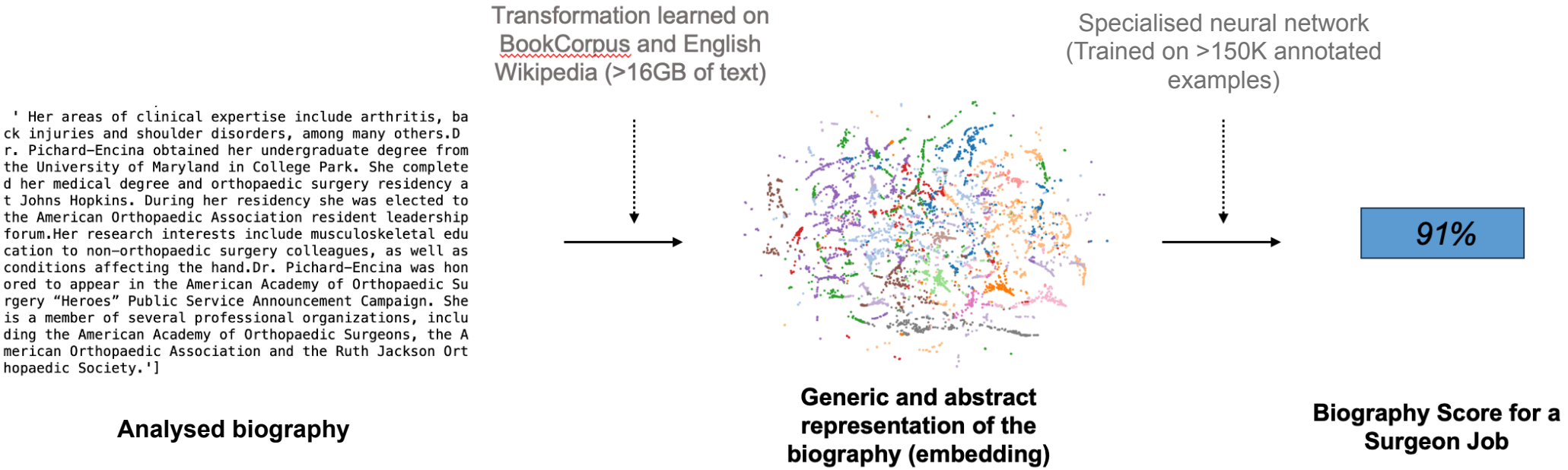


Whether we agree with the decision rules or not, they are explainable

- The person using the model can explain the decision
- The person to whom the decision is applied can understand how to have a positive decision
- It is possible to check whether all decision rules are legal

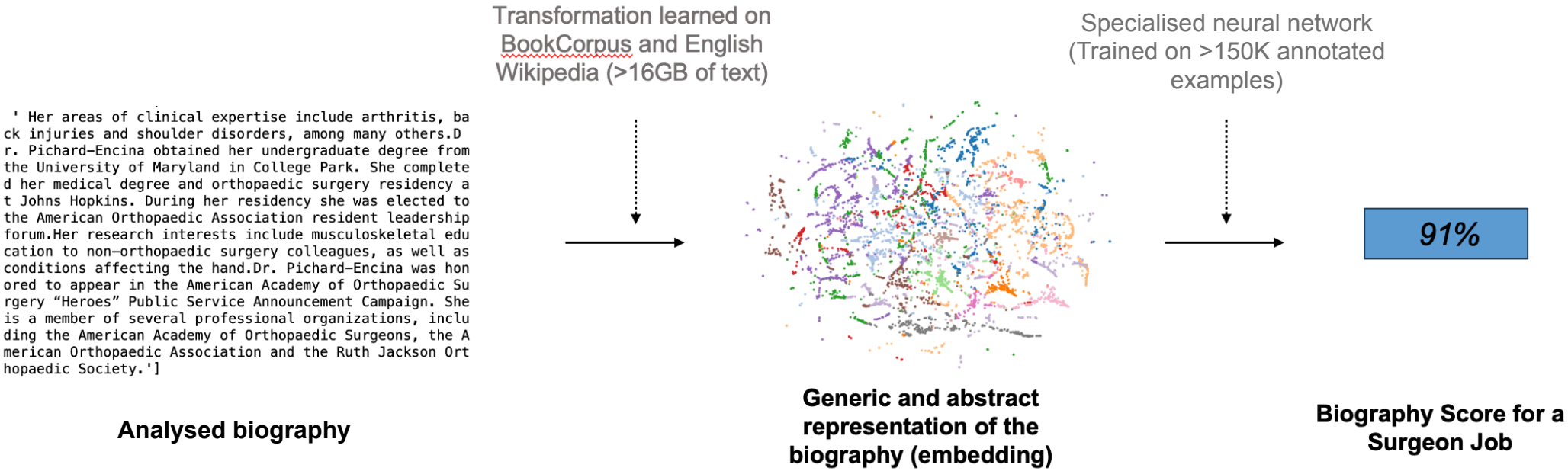
Introduction

Model for *pre-screening candidates*, from a very large database of *biographies*, for a specific job.



Introduction

Model for *pre-screening candidates*, from a very large database of *biographies*, for a specific job.

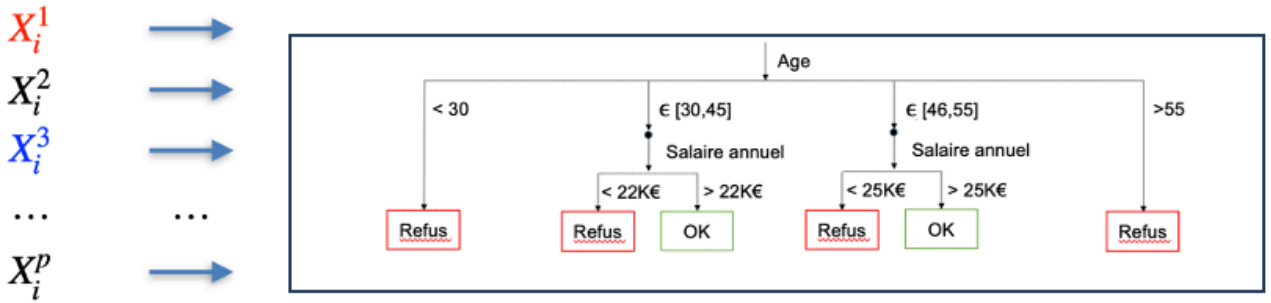


Visualisation of information processing possible (if the model is available)
… but it is humanly impossible to understand it because much more than millions of operations are carried out.

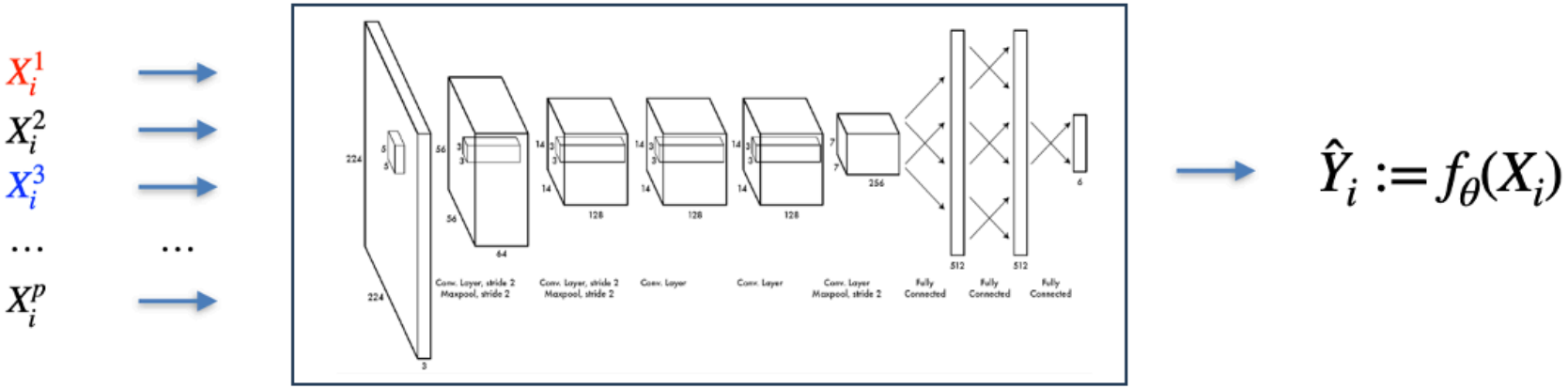
→ In different contexts, we may have on average fewer errors with a neural network than with a human, but the set of decision rules cannot be understood.

Introduction

- **White Boxes:** The Decision Rules Are Known



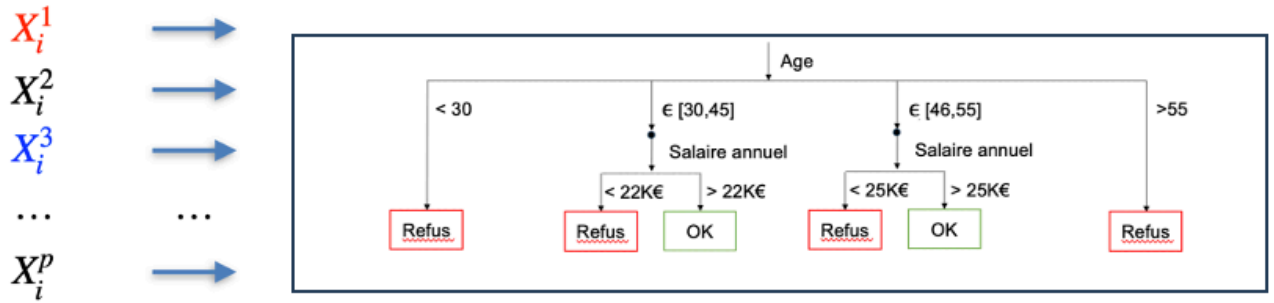
- **Grey boxes:** Partial access to decision rules (type of architecture, access to code and parameters, etc.)



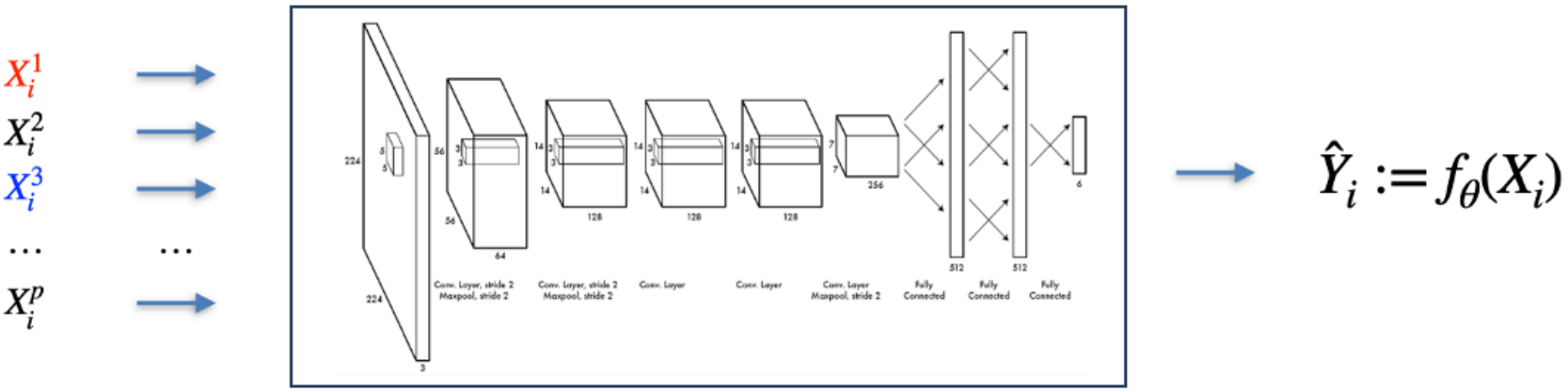
- **Black boxes:** One can only test the model without knowing what's in it

Introduction

- **White Boxes:** The Decision Rules Are Known



- **Grey boxes:** Partial access to decision rules (type of architecture, access to code and parameters, etc.)



- **Black boxes:** One can only test the model without knowing what's in it

Introduction

Part 1: Why XIA has become an important field of applied research?

- 1.1: How DNNs treat the information?
- 1.2: Robustness and hidden confounding variables
- 1.3: Legal requirements w.r.t. to explainability

Part 2 : Three solutions to explain machine learning decisions

- 2.1: LIME
- 2.2: GradCAM
- 2.3: GEMS-AI

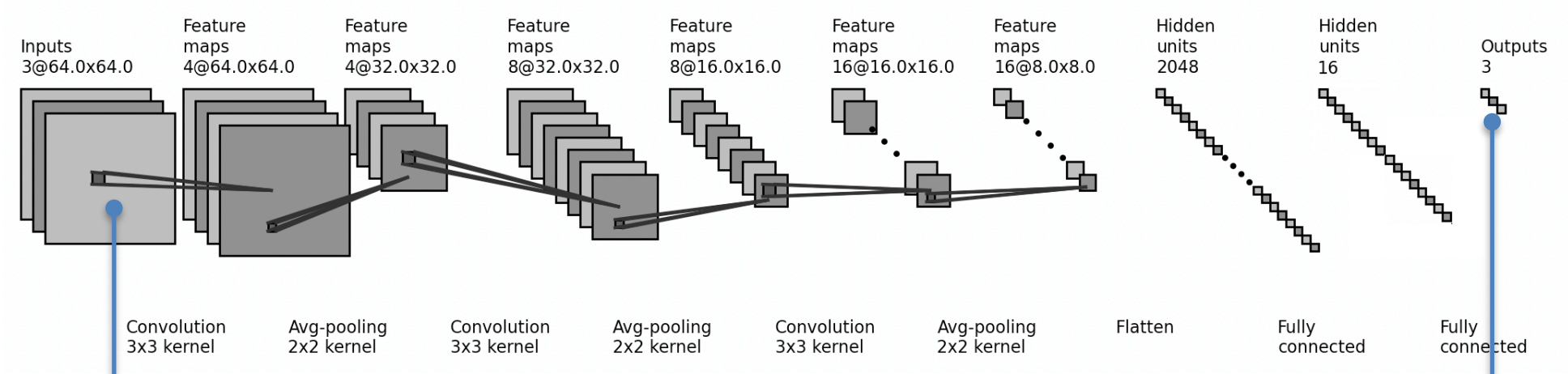
1.1 How DNNs treat the information

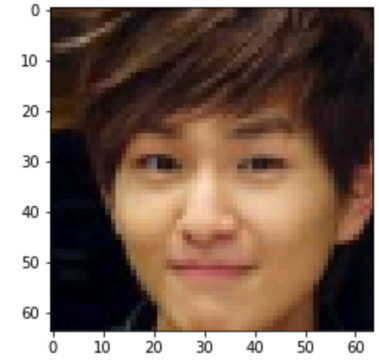
Illustration: CelebA dataset (https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html)

Inputs X_i : RGB images of 64×64 pixels

Outputs $\widehat{Y_i}$: Predicted observed features $\left(\mathbb{P}(eyeglasses), \mathbb{P}(smiling), \mathbb{P}(young)\right)$

1.1 How DNNs treat the information — Predictions



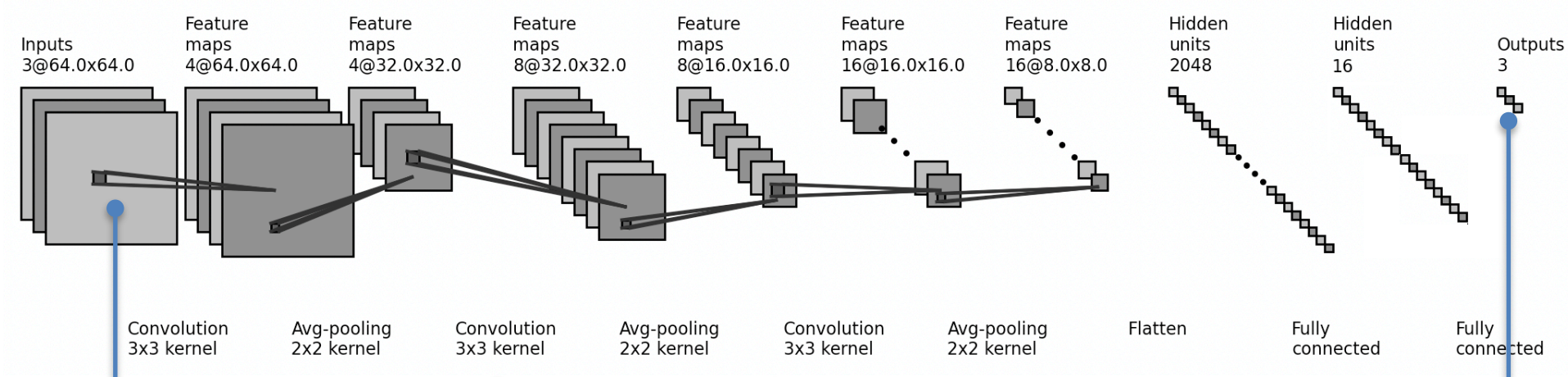


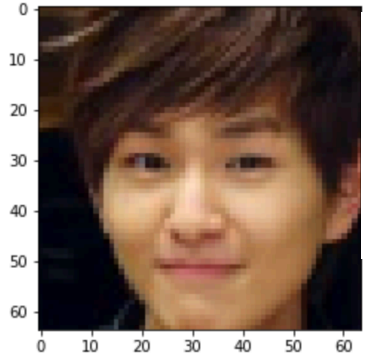
$$\mathbb{P}(eyeglasses) = 0.02$$

$$\mathbb{P}(smiling) = 0.93$$

$$\mathbb{P}(young) = 0.85$$

1.1 How DNNs treat the information — Predictions





$$\begin{cases} s_{i,j}^{l+1,c} = \sum_{c'=0}^{C_l} \sum_{v=0}^{2} \sum_{w=0}^{2} K_{v,w}^{l,c,c'} o_{i+v,j+w}^{l,c'} + b^{l+1,c} \\ o_{i,j}^{l+1,c} = max\left(0, s_{i,j}^{l+1,c}\right) \end{cases}$$

$$\begin{cases} s_i^{l+1} = \sum_{j=1}^{S_l} M_{i,j} o_j^l + b^{l+1} \\ o_i^{l+1} = max\left(0, s_i^{l+1}\right) \\ \text{ou} \quad o_i^{l+1} = \dfrac{1}{1 + e^{-s_i^{l+1}}} \end{cases}$$

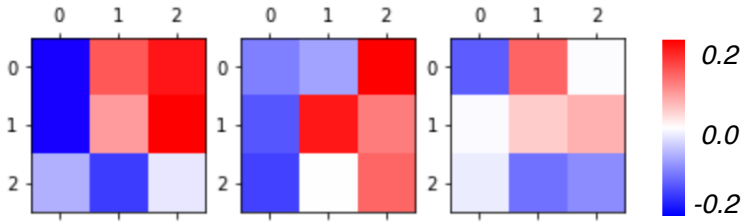
$$\mathbb{P}(eyeglasses) = 0.02$$

$$\mathbb{P}(smiling) = 0.93$$

$$\mathbb{P}(young) = 0.85$$

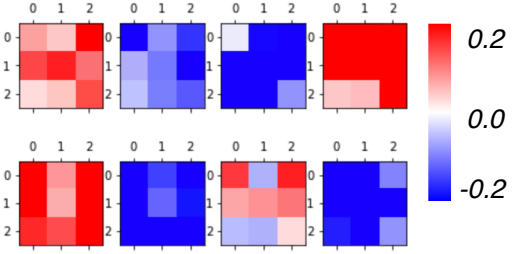
1.1 How DNNs treat the information — Predictions

$K_{v,w}^{l=1,c=0,c'=0:2} =$

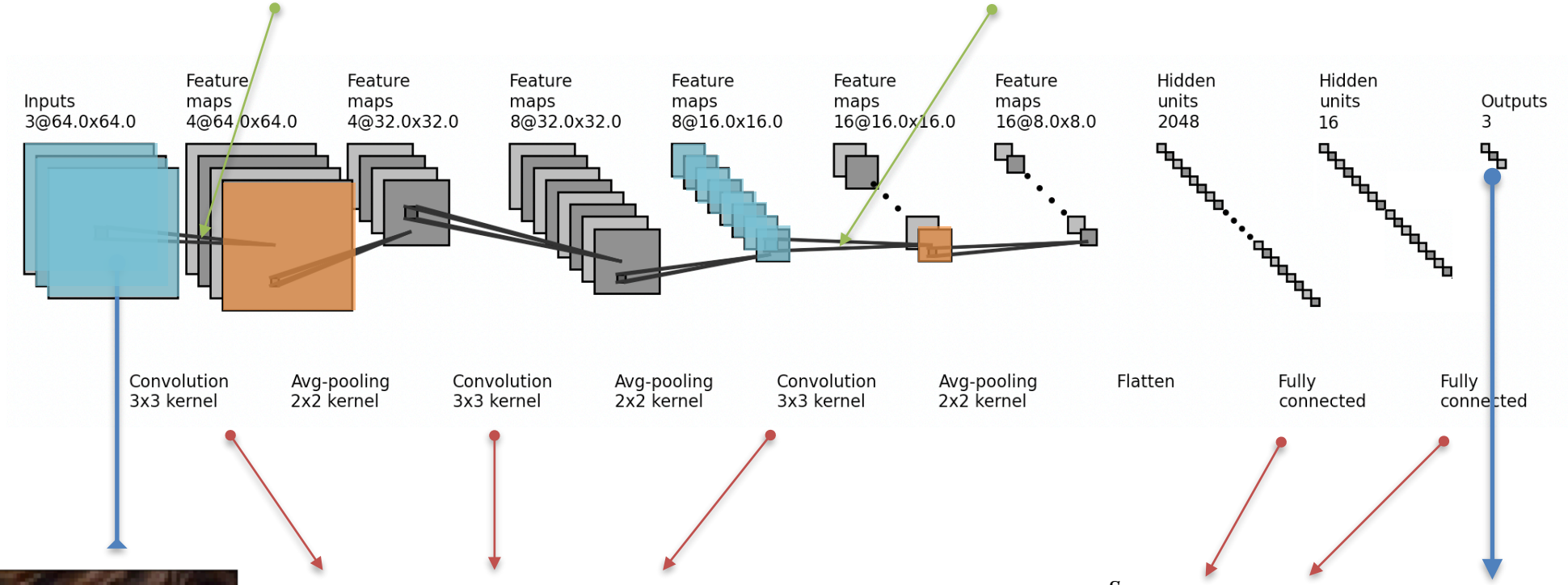


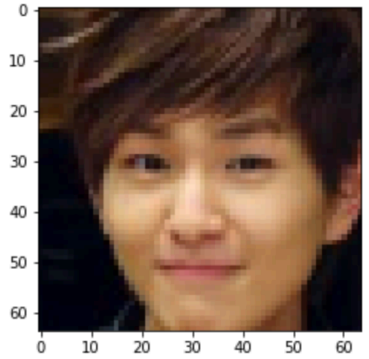
$b^{l=2,c=0} = 0.16$

$K_{v,w}^{l=3,c=0,c'=0:8} =$



$b^{l=4,c=0} = -0.01$





$$\begin{cases} s_{i,j}^{l+1,c} = \sum_{c'=0}^{C_l} \sum_{v=0}^{2} \sum_{w=0}^{2} K_{v,w}^{l,c,c'} o_{i+v,j+w}^{l,c'} + b^{l+1,c} \\ o_{i,j}^{l+1,c} = max\left(0, s_{i,j}^{l+1,c}\right) \end{cases}$$

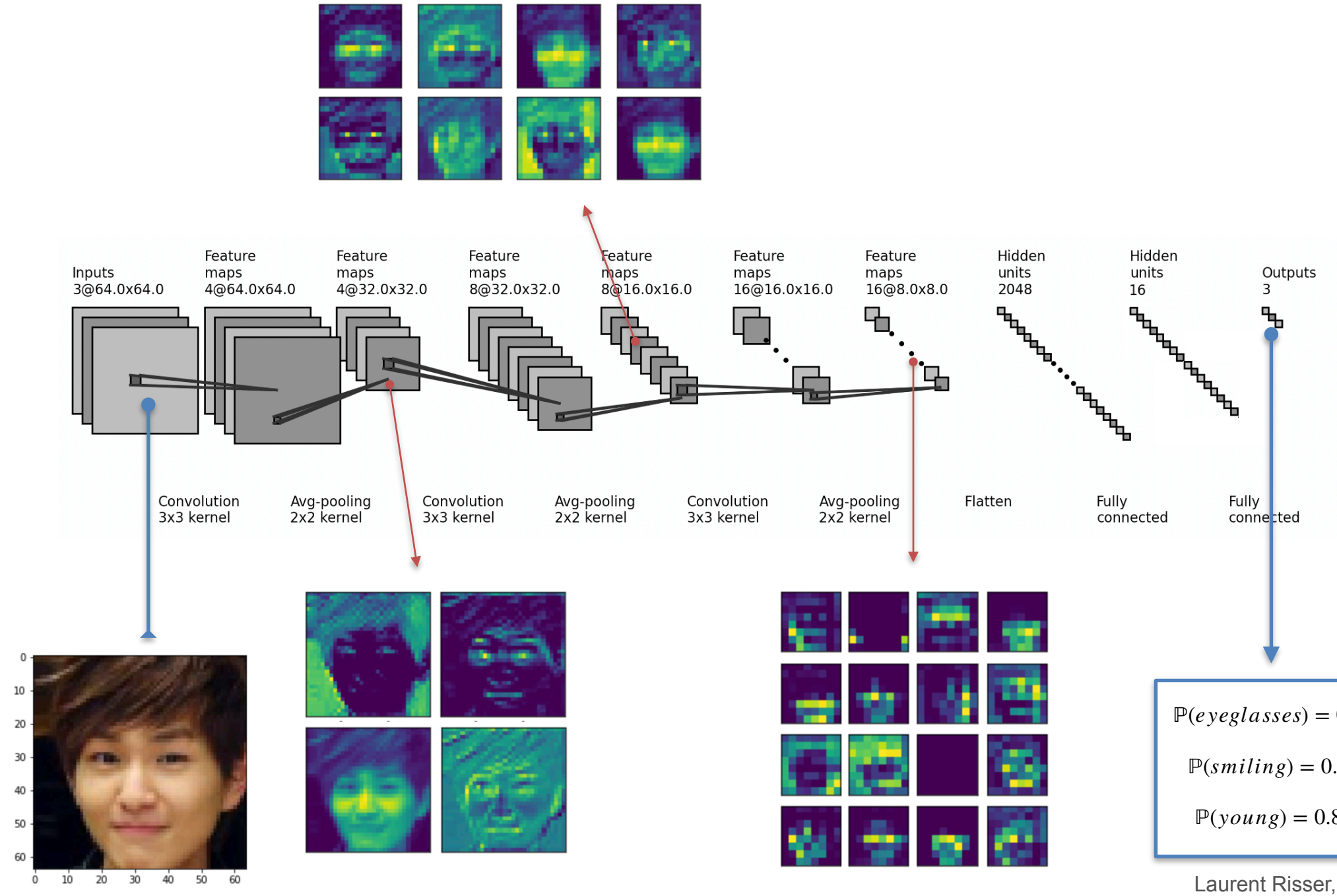
$$\begin{cases} s_i^{l+1} = \sum_{j=1}^{S_l} M_{i,j} o_j^l + b^{l+1} \\ o_i^{l+1} = max\left(0, s_i^{l+1}\right) \\ \text{ou} \quad o_i^{l+1} = \dfrac{1}{1+e^{-s_i^{l+1}}} \end{cases}$$

$\mathbb{P}(eyeglasses) = 0.02$

$\mathbb{P}(smiling) = 0.93$

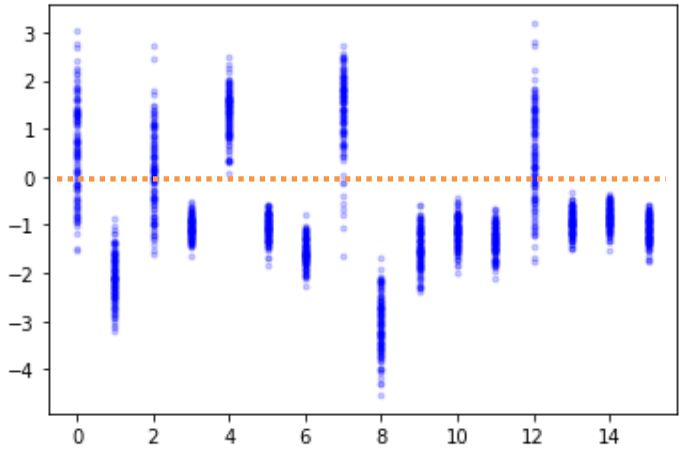
$\mathbb{P}(young) = 0.85$

1.1 How DNNs treat the information — Predictions

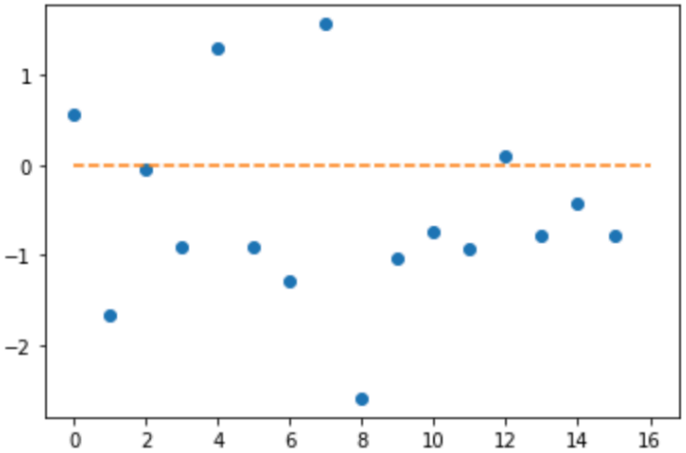


Laurent Risser, 2024

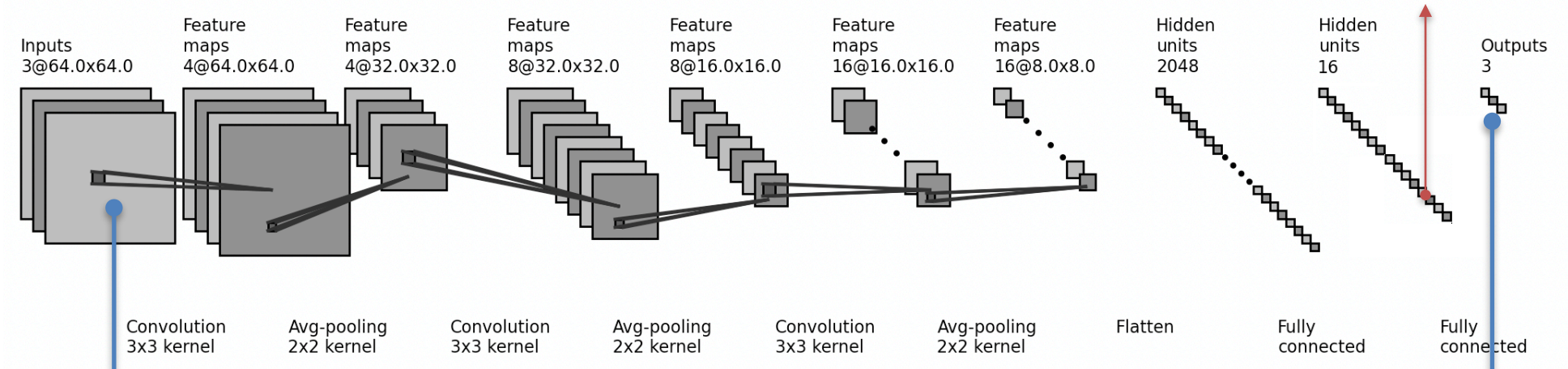
1.1 How DNNs treat the information — Predictions

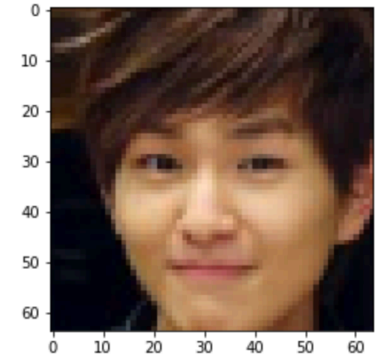


Variability of latent data



Latent data / feature space





$\mathbb{P}(eyeglasses) = 0.02$

$\mathbb{P}(smiling) = 0.93$

$\mathbb{P}(young) = 0.85$

1.1 How DNNs treat the information — Training

Training set: $\{X_i, y_i\}_{i=1,\dots,n}$

Predictions: $\widehat{y_i} = f_\theta(X_i)$

Neural network parameters: θ

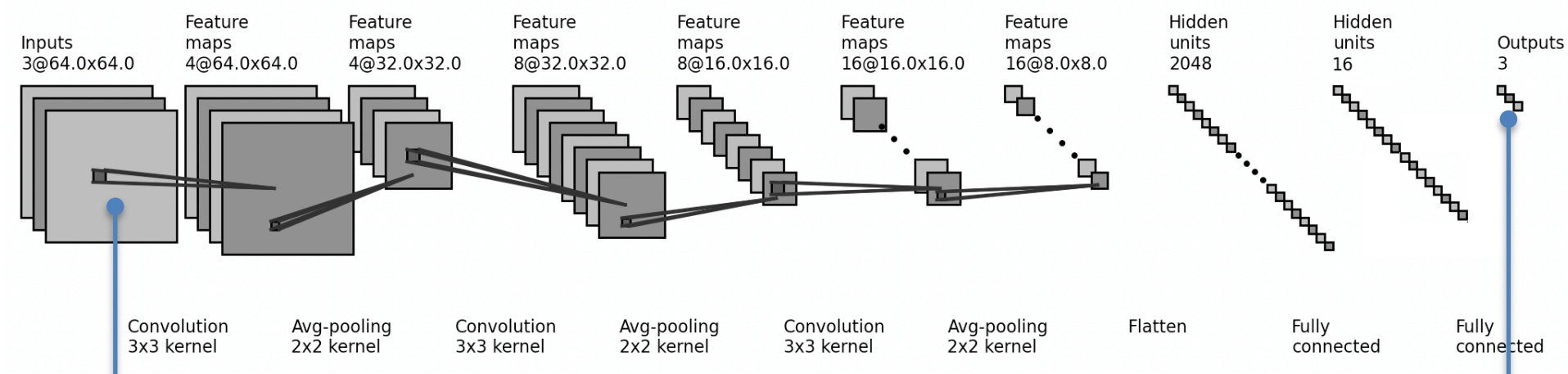
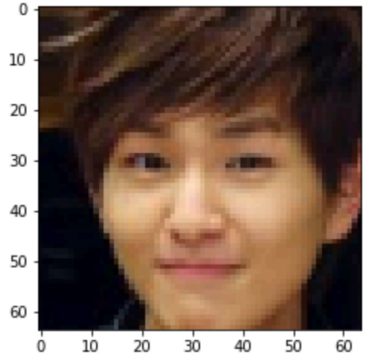


Image X_i



True labels y_i

$eyeglasses = 0$

$smiling = 1$

$young = 1$

Predictions $f_\theta(X_i)$

$\mathbb{P}(eyeglasses) = 0.02$

$\mathbb{P}(smiling) = 0.93$

$\mathbb{P}(young) = 0.85$

1.1 How DNNs treat the information — Training

Training set: $\{X_i, y_i\}_{i=1,\dots,n}$

Predictions: $\widehat{y_i} = f_\theta(X_i)$

Neural network parameters: θ

$$\hat{\theta} = \arg\min_\theta \frac{1}{n}\sum_{i=1}^{n} loss_i = \arg\min_\theta \underbrace{\frac{1}{n}\sum_{i=1}^{n} loss(f_\theta(X_i), y_i)}_{R_\theta}$$

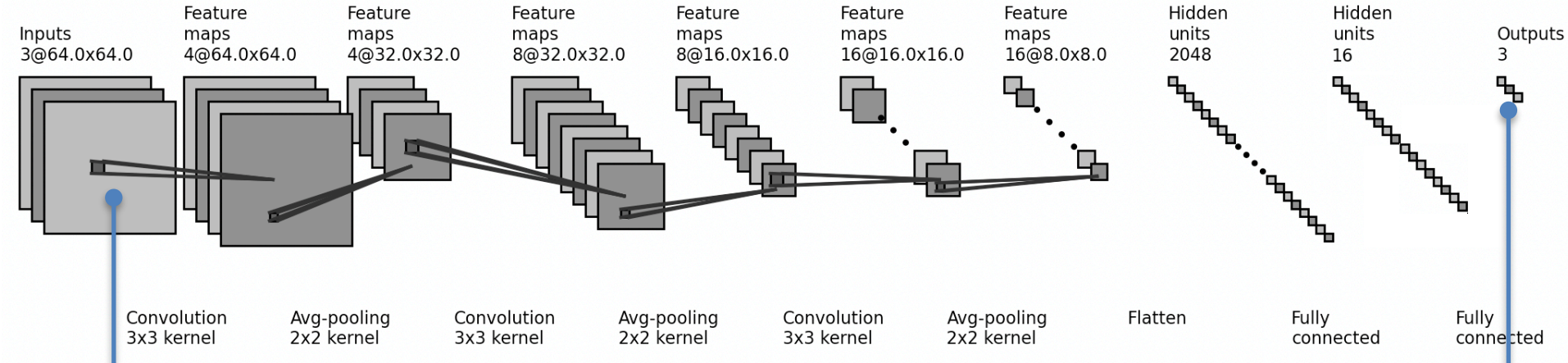
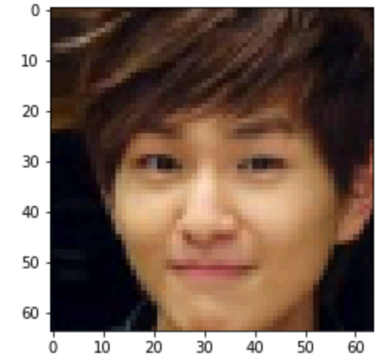


Image X_i



$$loss_i = (0.02-0)^2 + (0.93-1)^2 + (0.85-1)^2$$

True labels y_i

$eyeglasses = 0$

$smiling = 1$

$young = 1$

Predictions $f_\theta(X_i)$

$\mathbb{P}(eyeglasses) = 0.02$

$\mathbb{P}(smiling) = 0.93$

$\mathbb{P}(young) = 0.85$

1.1 How DNNs treat the information — Training

- Gradient descent strategy : Incremental updates of θ to minimise R_θ
- Need to compute the derivatives of R_θ w.r.t. all neural-network parameters θ: $\nabla_\theta R_\theta = \left(\frac{\partial R_\theta}{\partial \theta_1}, \frac{\partial R_\theta}{\partial \theta_2}, \ldots, \frac{\partial R_\theta}{\partial \theta_K} \right)$
- Gradient computed by using the back-propagation algorithm

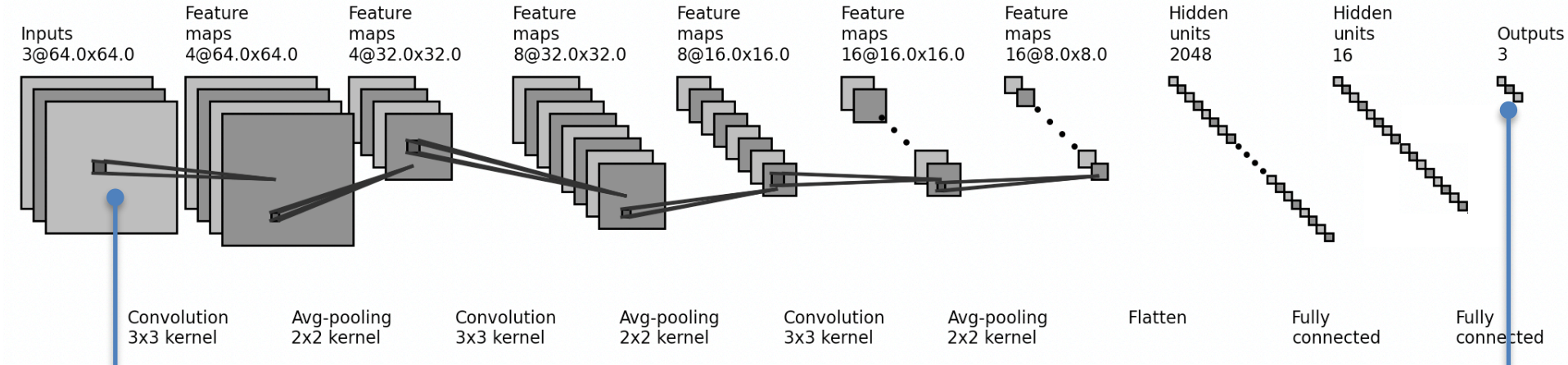
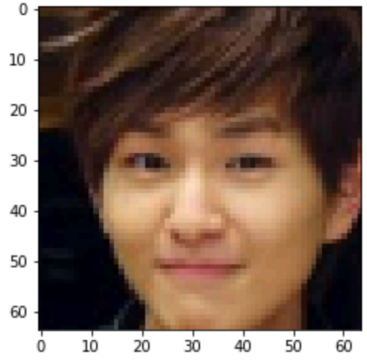


Image X_i



$loss_i = (0.02 - 0)^2 + (0.93 - 1)^2 + (0.85 - 1)^2$

True labels y_i

$eyeglasses = 0$

$smiling = 1$

$young = 1$

Predictions $f_\theta(X_i)$

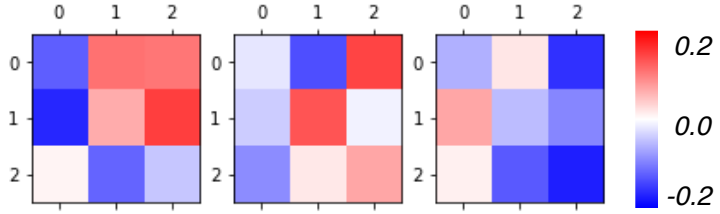
$\mathbb{P}(eyeglasses) = 0.02$

$\mathbb{P}(smiling) = 0.93$

$\mathbb{P}(young) = 0.85$

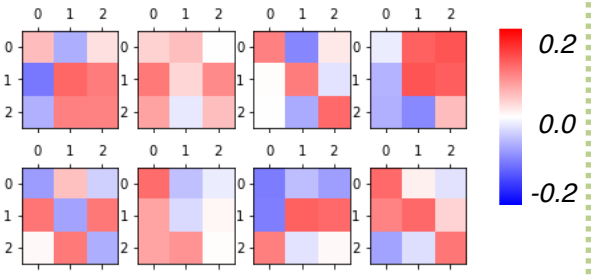
1.1 How DNNs treat the information — Training

$K^{l=1,c=0,c'=0:2}_{v=0:2,w=0:2} =$

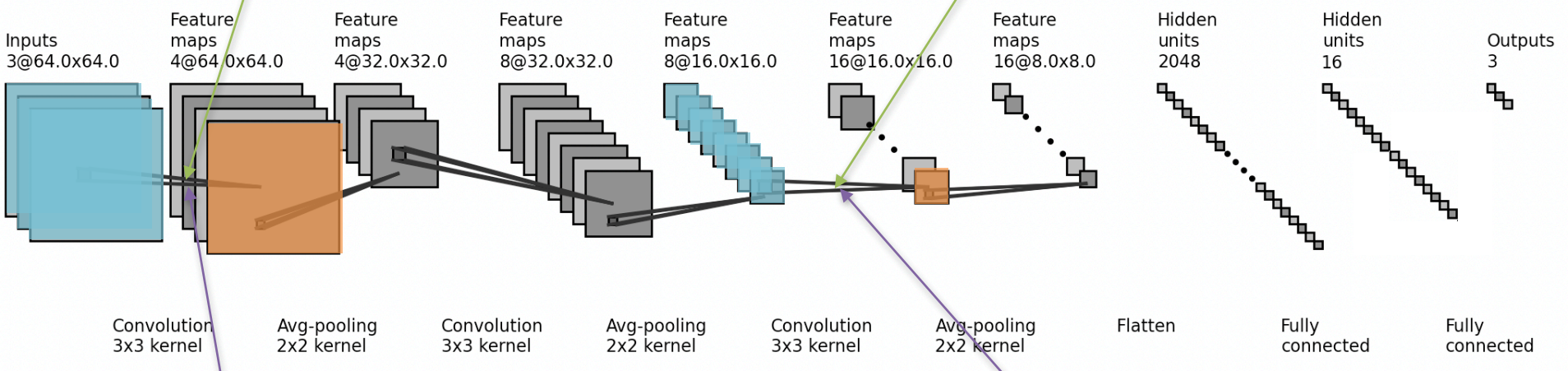


$b^{l=2,c=0} = 0.16$

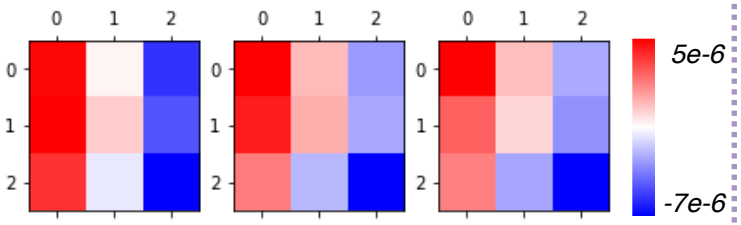
$K^{l=3,c=0,c'=0:7}_{v=0:2,w=0:2} =$



$b^{l=4,c=0} = -0.01$

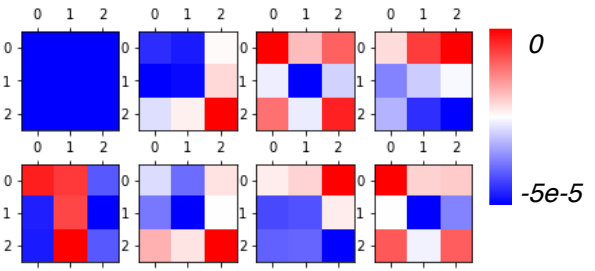


$\frac{\partial R_\theta}{\partial K^{l=1,c=0,c'=0:2}_{v=0:2,w=0:2}} =$



$\frac{\partial R_\theta}{\partial b^{l=2,c=0}} = -5.6e-5$

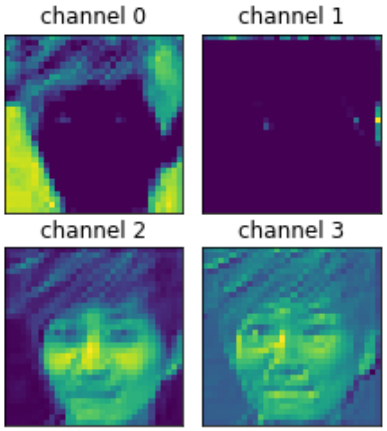
$\frac{\partial R_\theta}{\partial K^{l=3,c=0,c'=0:7}_{v=0:2,w=0:2}} =$

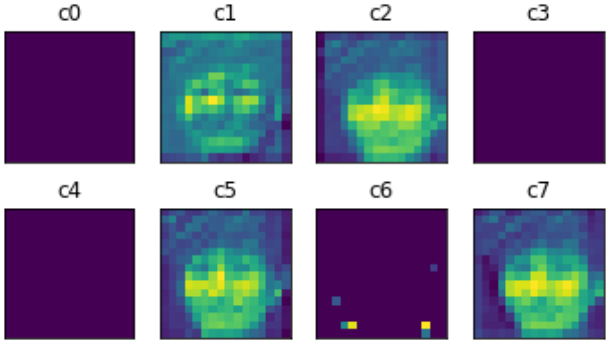


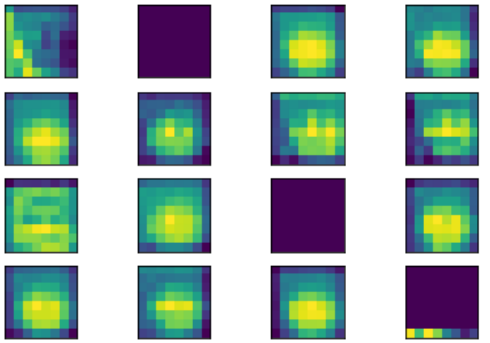
$\frac{\partial R_\theta}{\partial b^{l=4,c=0}} = -5.9e-5$

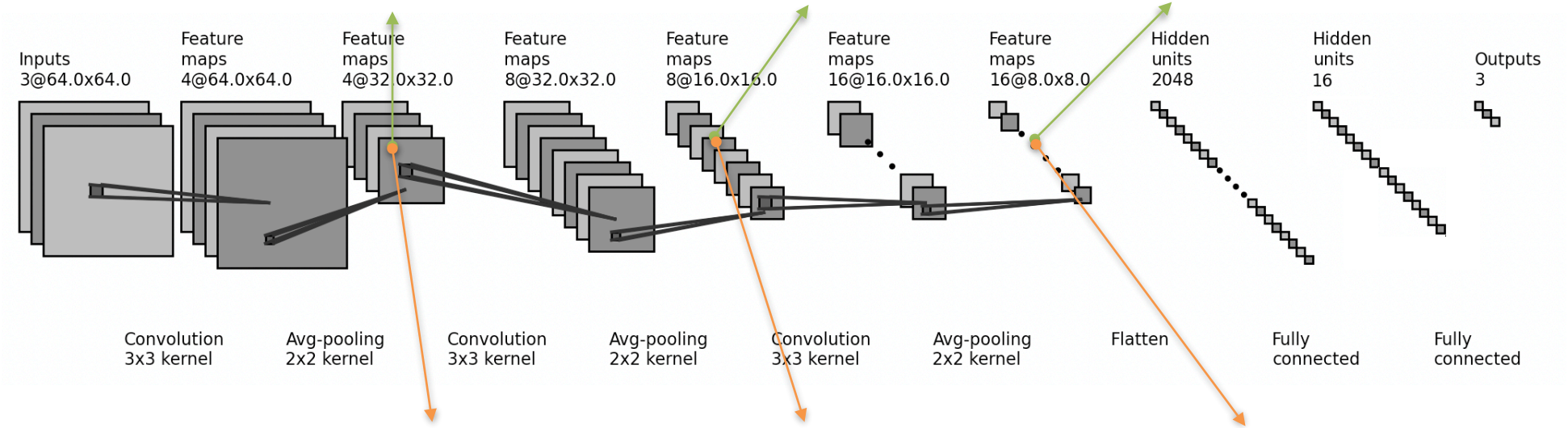
1.1 How DNNs treat the information — Training

Hidden representations of X_i

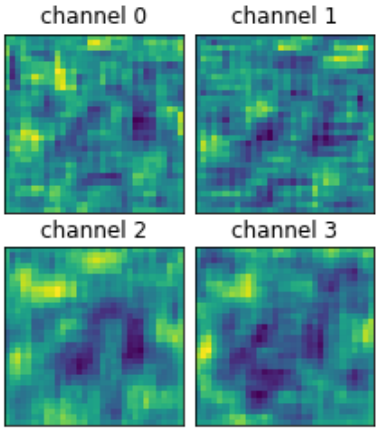


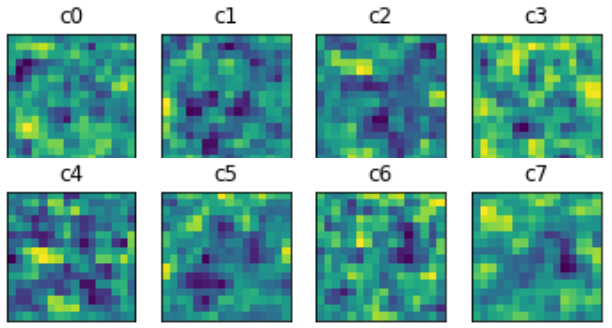


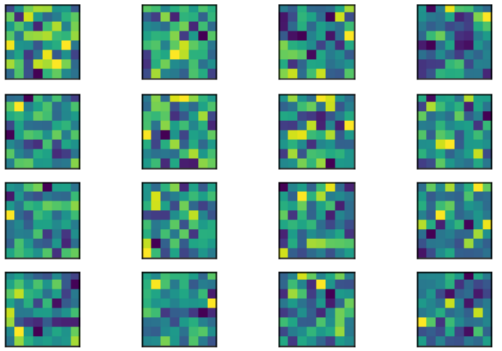




Derivatives of R_θ w.r.t. all intensities in the hidden representations of X_i

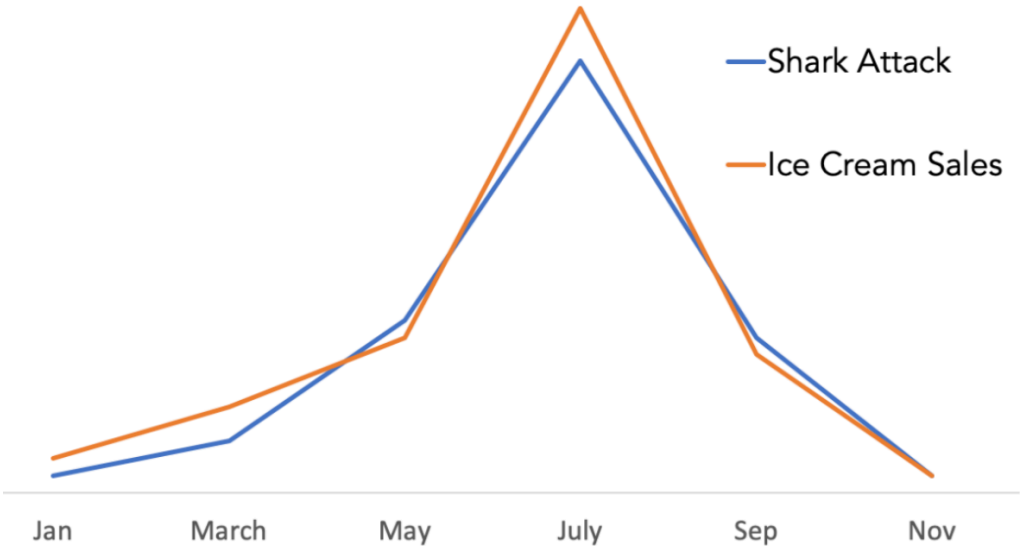




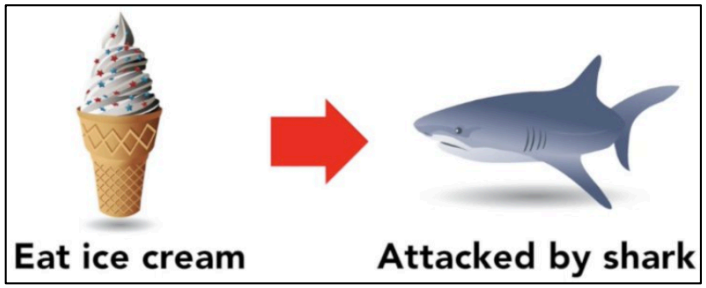


1.2 Robustness in DNNs and hidden confounding variables

ICE CREAM AND SHARKS EXAMPLE

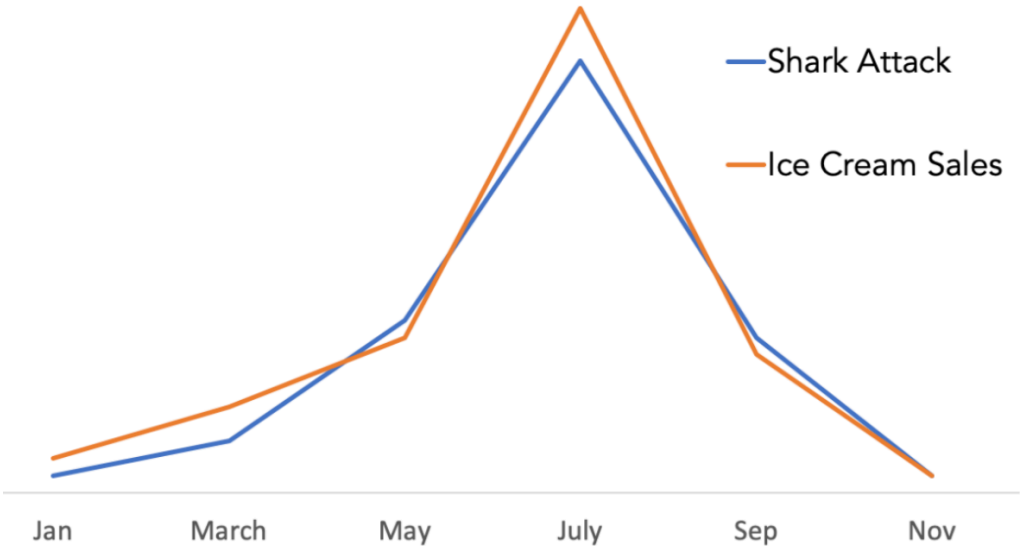


???

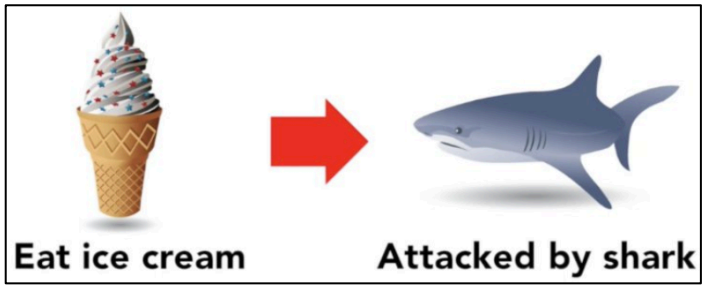


1.2 Robustness in DNNs and hidden confounding variables

ICE CREAM AND SHARKS EXAMPLE



???



⇢ Correlation is <u>not</u> causality

⇢ Here the *hot temperature* is the confounding variable

1.2 Robustness in DNNs and hidden confounding variables

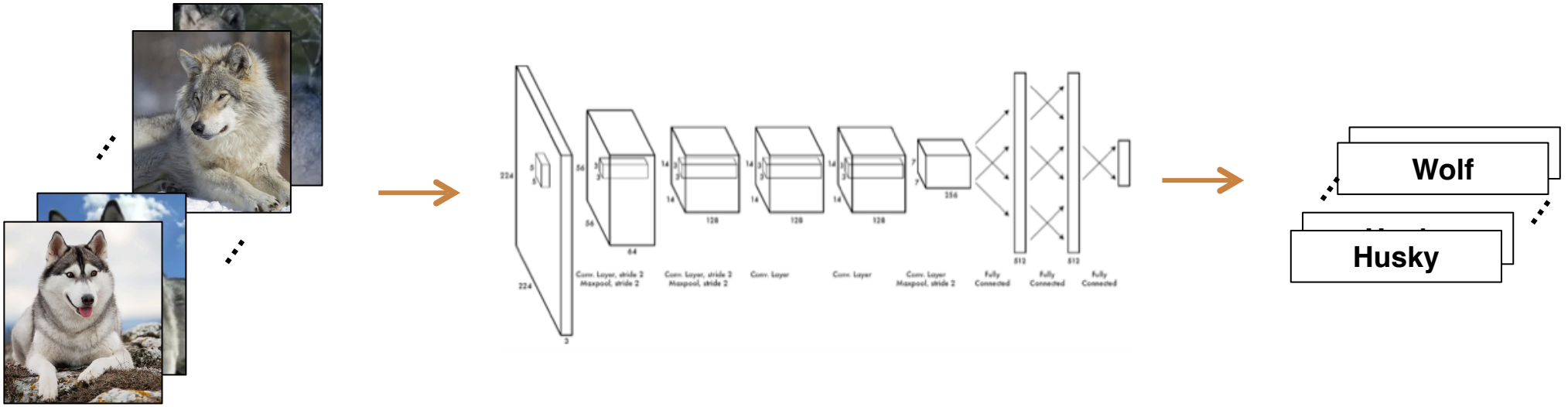
HUSKIES AND WOLVES EXAMPLE (RIBEIRO ET AL, 2016)

Goal: Automatic recognition of a husky or a wolf based on a picture

1.2 Robustness in DNNs and hidden confounding variables

HUSKIES AND WOLVES EXAMPLE (RIBEIRO ET AL, 2016)

Step 1: Train neural network parameters

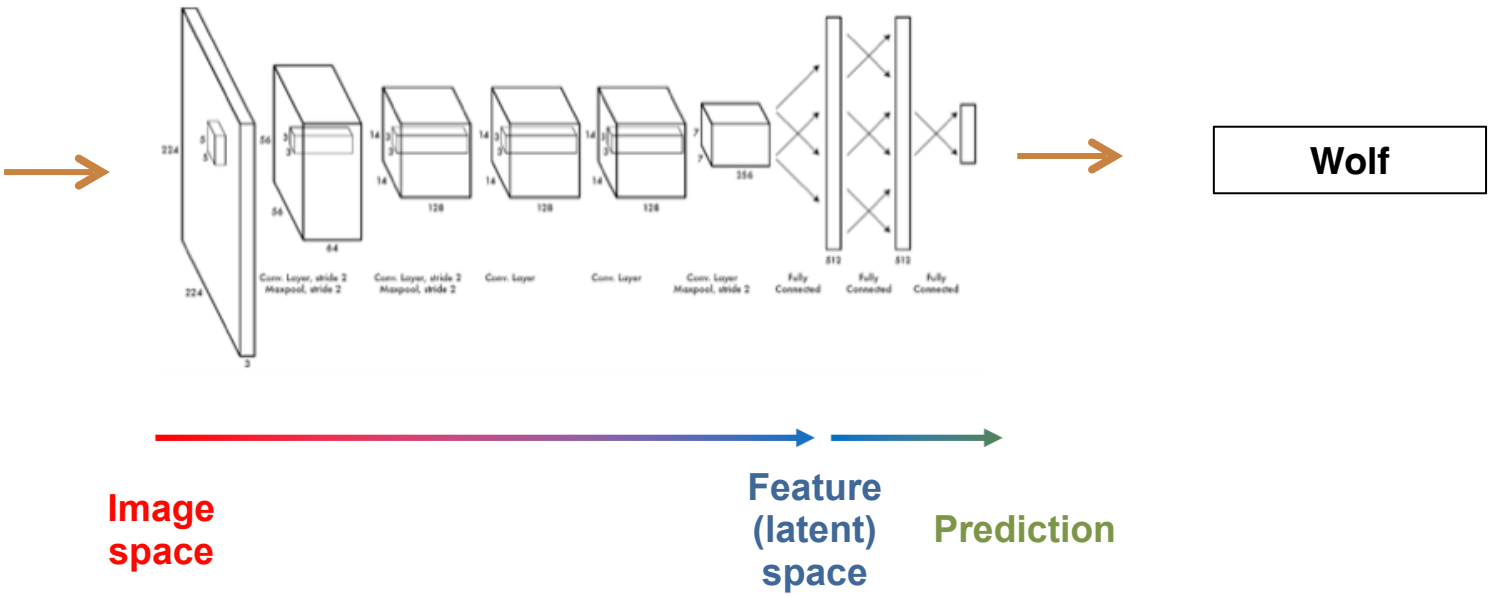


1.2 Robustness in DNNs and hidden confounding variables

HUSKIES AND WOLVES EXAMPLE (RIBEIRO ET AL, 2016)

Step 2: Predictions

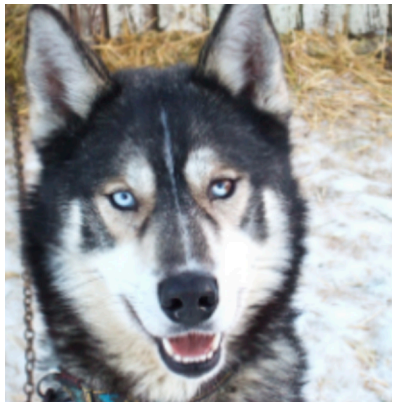
Good predictions on 95% of the images in the test set!

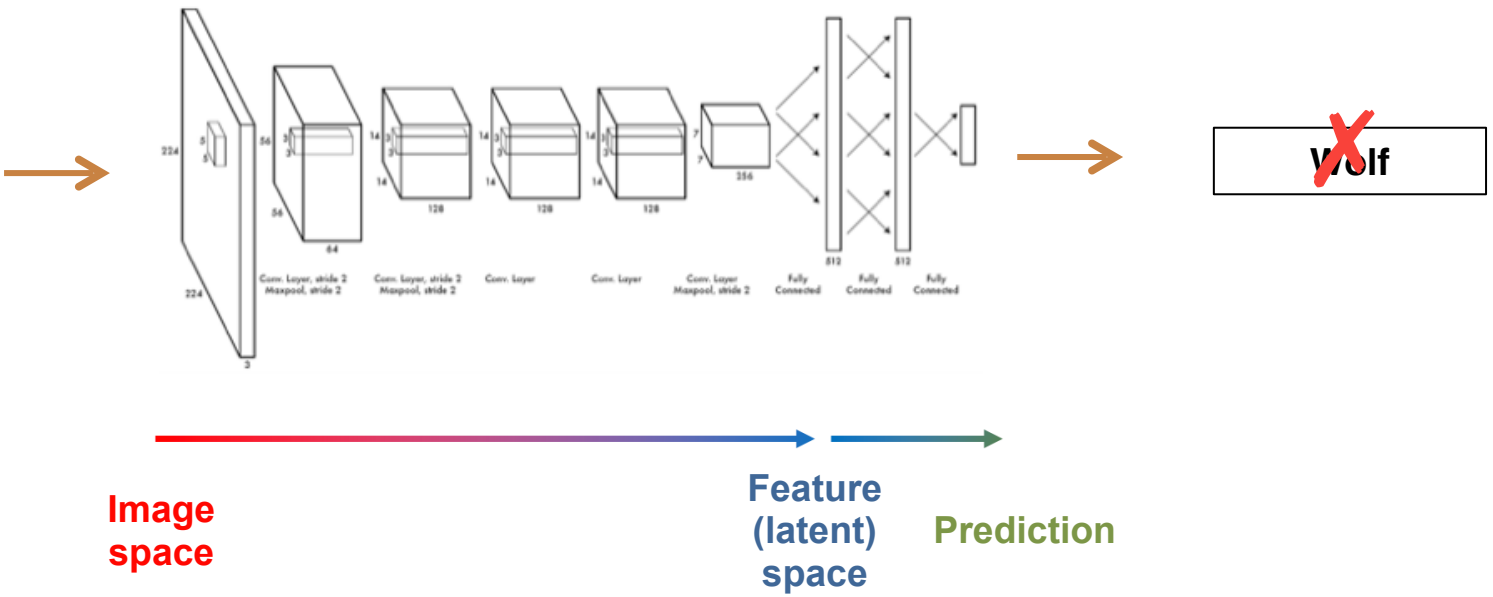


1.2 Robustness in DNNs and hidden confounding variables

HUSKIES AND WOLVES EXAMPLE (RIBEIRO ET AL, 2016)

Step 2: Predictions

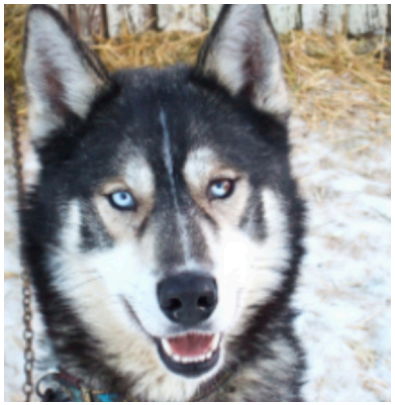


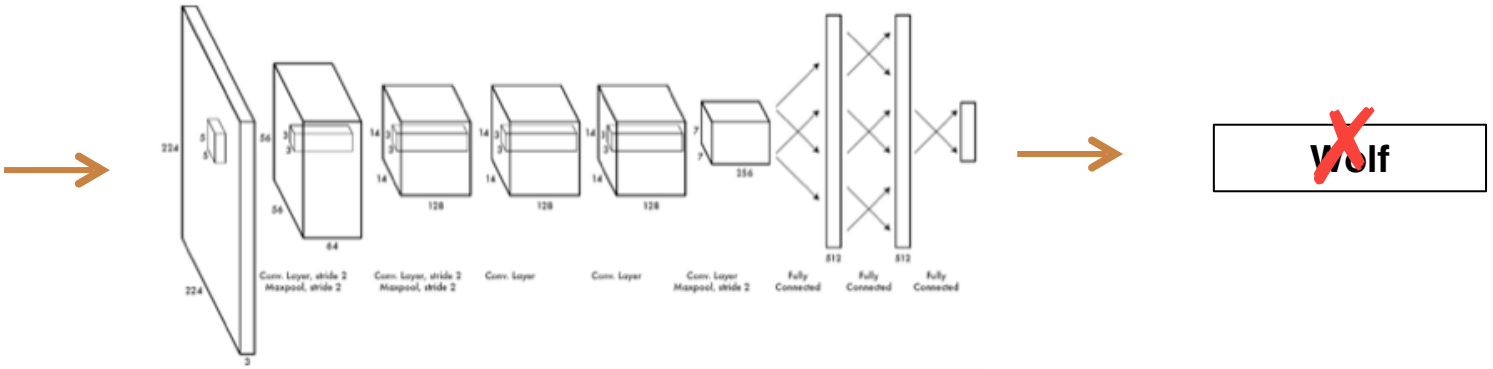


1.2 Robustness in DNNs and hidden confounding variables

HUSKIES AND WOLVES EXAMPLE (RIBEIRO ET AL, 2016)

Step 2: Predictions

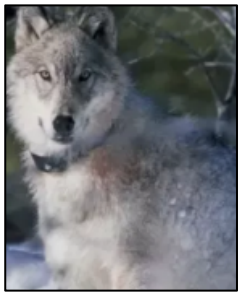




Why?

⇢ In the training set, most pictures representing a wolf also represent a snowy background, which is not the case for huskies.

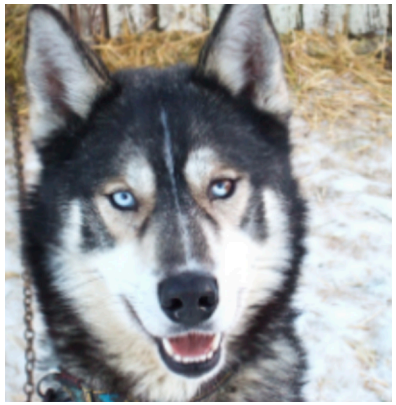
⇢ The neural-network associated a snowy background to wolves

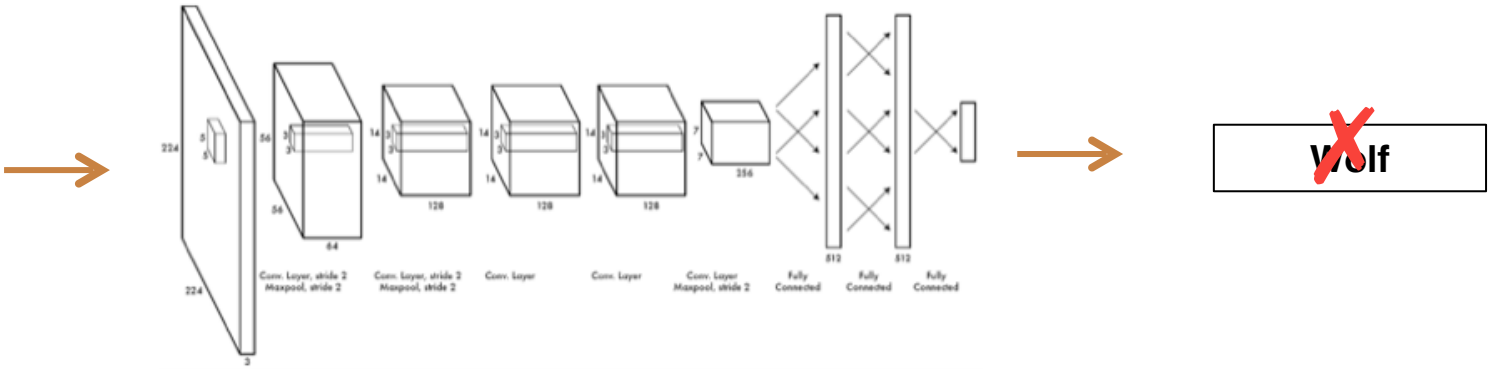


1.2 Robustness in DNNs and hidden confounding variables

HUSKIES AND WOLVES EXAMPLE (RIBEIRO ET AL, 2016)

Step 2: Predictions





Neural networks are trained to make good decisions on average…

but even righteous decisions can be made for the wrong reasons.

1.3 Legal requirements with respect to explainability

EMERGENCE OF A RIGHT TO EXPLANATION

- Fr (Loi Informatique et Libertés — 1978) : « Right to understand the rules of automatic treatments and their main characteristics »
- NYC Bill (Dec. 2017) : Local laws related to automatic decision systems
- E.U. (RGPD, art 22 — 2018) : « Right not to be subject to a decision solely based on automated processing, including profiling » ; « Underlying logic of the algorithm and its consequences must be given »
- E.U. (AI act — 2021) : « Definition of High risk AI systems » ; « Right to understand automatic decisions made by A.I. systems for High Risk applications »

- Evolution of the legal constraints with the evolution of the technologies
- Being able to explain DNN decisions increasingly becomes a legal requirement for many applications

Introduction

Part 1: Why XIA has become an important field of applied research?

- 1.1: How DNNs treat the information?
- 1.2: Robustness and hidden confounding variables
- 1.3: Legal requirements w.r.t. to explainability

Part 2 : Three solutions to explain machine learning decisions

- 2.1: LIME
- 2.2: GradCAM
- 2.3: GEMS-AI

2 Three XIA techniques

Recent papers dealing with Explainable Artificial Intelligence (XAI) :

"Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI", A. Barrieta et al, 2019

"Interpretable Explanations of Black Boxes by Meaningful Perturbation", Ruth C. Fong, Andrea Vedaldi, 2017

"*MAGIX: model agnostic globally interpretable explanations*," N. Puri, P. Gupta, P. Agarwal, S. Verma, and B. Krishnamurthy, CoRR,vol. abs/1706.07160, 2017.

"Why should I trust you? Explaining the predictions of any classifier.", T. Ribeiro, S. Singh, and C. Guestrin, 2016 - International Conference on Knowledge Discovery and Data Mining, ACM2016

"Local Rule-Based Explanations of Black Box Decision Systems" (LORE),Riccardo Guidotti et al 2018,

"Anchors: High-precision model-agnostic explanations," T. Ribeiro, S. Singh, and C. Guestrin, , in AAAI Conference on Artificial Intelligence, 2018.

"Visualizing the feature importance for black box models", G. Casalicchio, C. Molnar, B. Bischl, arXiv:1804.06620.

"*Auditing black-box models for indirect influence"*, P. Adler, C. Falk, S. A. Friedler, T. Nix, G. Rybeck, C. Scheidegger, B. Smith, S. Venkatasubramanian, Knowledge and Information Systems 54 (1) (2018) 95–122.

"*Entropic Variable Projection for Explainability and Intepretability"*, F. Bachoc and F. Gamboa and M. Halford and J.-M. Loubes and L. Risser, 2018, arXiv:1810.07924.

"Grad-cam: Visual explanations from deep networks via gradient-based localization", R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.

"Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning" , N. Papernot, P. McDaniel, (2018). arXiv:1803.04765.

"Interpretable convolutional neural networks", Q. Zhang, Y. Nian Wu, S.-C. Zhu, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8827–8836.

"InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets" , X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, P. Abbeel, (2016). arXiv:1606.03657

"Not just a black box: Learning important features through propagating activation differences", Shrikumar, P. Greenside, A. Shcherbina, A. Kundaje, 2016, arXiv:1605.01713

"Interpretable explanations of black boxes by meaningful perturbation", R. C. Fong, A. Vedaldi, in: IEEE International Conference on Computer Vision, 2017, pp. 3429–3437.

"On the Robustness of Interpretability Methods", Alvarez-Melis et T. S. Jaakkola, *arXiv:1806.08049 [cs, stat]*, juin 2018.

"Interpretable Deep Learning under Fire", X. Zhang, N. Wang, H. Shen, S. Ji, X. Luo, et T. Wang, *arXiv:1812.00891 [cs]*, sept. 2019.

"Improving the Adversarial Robustness and Interpretability of Deep Neural Networks by Regularizing their Input Gradients", S. Ross et F. Doshi-Velez, arXiv:1711.09404 [cs], nov. 2017.

"Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR", Wachter, B. Mittelstadt, et C. Russell, SSRN Journal, 2017.

… and many others …

We focus on:

- Post-hoc explanations
- Neural networks
- Images and tables

2.1 Local Interpretable Model-agnostic Explanations (LIME)

"Why Should I Trust You?"
Explaining the Predictions of Any Classifier

Marco Tulio Ribeiro
University of Washington
Seattle, WA 98105, USA
marcotcr@cs.uw.edu

Sameer Singh
University of Washington
Seattle, WA 98105, USA
sameer@cs.uw.edu

Carlos Guestrin
University of Washington
Seattle, WA 98105, USA
guestrin@cs.uw.edu

https://arxiv.org/pdf/1602.04938.pdf
https://homes.cs.washington.edu/~marcotcr/blog/lime/
https://github.com/marcotcr/lime

LIME explains why a specific (local) prediction is made by using an explainable surrogate model

Input variables X_i | **Trained black-box model** f_θ | **Prediction** $\hat{Y}_i = f_\theta(X_i)$

Training a local surrogate models to explain the prediction of X_i with f_θ:

- Randomly perturb $X_i \rightarrow \{X_i^p\}_{p=1,\ldots,P}$
- Define a distance for the perturbed observations $\pi_{X_i}(X_i^p) = dist(X_i, X_i^p)$.
- Consider an explainable model $g_{\theta'}$ (e.g. a linear model, …)
- Optimise the parameters θ' by minimising: $\sum_{p=1}^{P} \pi_{X_i}(X_i^p)(g_{\theta'}(X_i^p) - f_\theta(X_i^p))^2$
- Explain the prediction thanks to $g(\theta')$

2.1 Local Interpretable Model-agnostic Explanations (LIME)

"Why Should I Trust You?"
Explaining the Predictions of Any Classifier

Marco Tulio Ribeiro
University of Washington
Seattle, WA 98105, USA
marcotcr@cs.uw.edu

Sameer Singh
University of Washington
Seattle, WA 98105, USA
sameer@cs.uw.edu

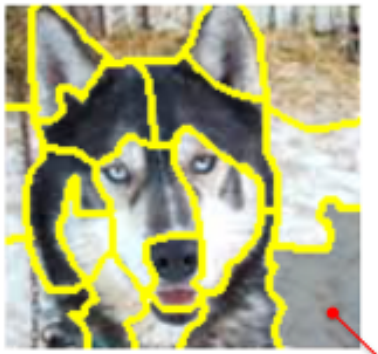
Carlos Guestrin
University of Washington
Seattle, WA 98105, USA
guestrin@cs.uw.edu

https://arxiv.org/pdf/1602.04938.pdf
https://homes.cs.washington.edu/~marcotcr/blog/lime/
https://github.com/marcotcr/lime

Application to the « Wolves and Hukies » example

Husky predicted as a Wolf

(1) Image segmentation



(2) Modify the intensities in each region

(3) Quantify the sensitivity of the predictions to intensity changes

Intensity changes

(4) Explanation : regions for which the intensity changes have the highest impact on the predictions

2.1 Local Interpretable Model-agnostic Explanations (LIME)

"Why Should I Trust You?"
Explaining the Predictions of Any Classifier

Marco Tulio Ribeiro
University of Washington
Seattle, WA 98105, USA
marcotcr@cs.uw.edu

Sameer Singh
University of Washington
Seattle, WA 98105, USA
sameer@cs.uw.edu

Carlos Guestrin
University of Washington
Seattle, WA 98105, USA
guestrin@cs.uw.edu

https://arxiv.org/pdf/1602.04938.pdf
https://homes.cs.washington.edu/~marcotcr/blog/lime/
https://github.com/marcotcr/lime

Application to tabular data (Adult census dataset — https://www.kaggle.com/uciml/adult-census-income):

Age (X^1)	**Education.num** (X^2)	**Marital.status** (X^3)	**Hours.per.week** (X^4)	…	**Loan granted — True** (Y)	**Loan granted — Predicted** ($\hat{Y} = f_\theta(X)$)
54	4	Divorced	40		No	No
41	10	Never-married	60		Yes	Yes
51	13	Married-civ	40		Yes	No
39	14	Married-civ	65		Yes	Yes
49	10	Divorced	50		No	Yes
…	…	…	…		…	…

2.1 Local Interpretable Model-agnostic Explanations (LIME)

"Why Should I Trust You?"
Explaining the Predictions of Any Classifier

Marco Tulio Ribeiro
University of Washington
Seattle, WA 98105, USA
marcotcr@cs.uw.edu

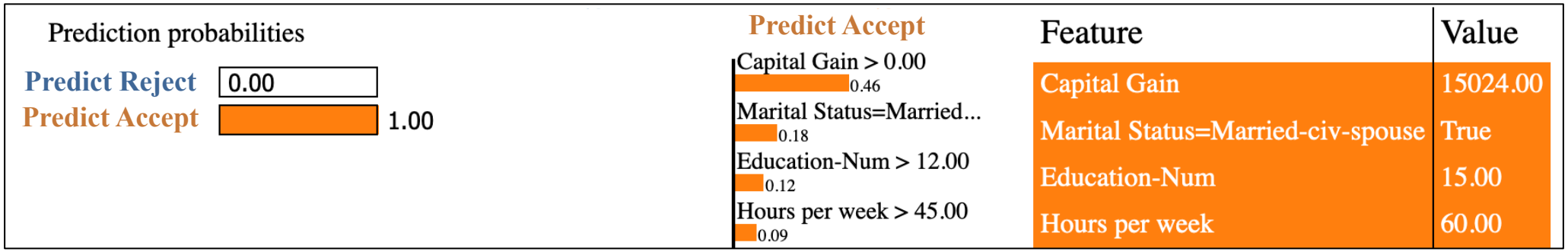
Sameer Singh
University of Washington
Seattle, WA 98105, USA
sameer@cs.uw.edu

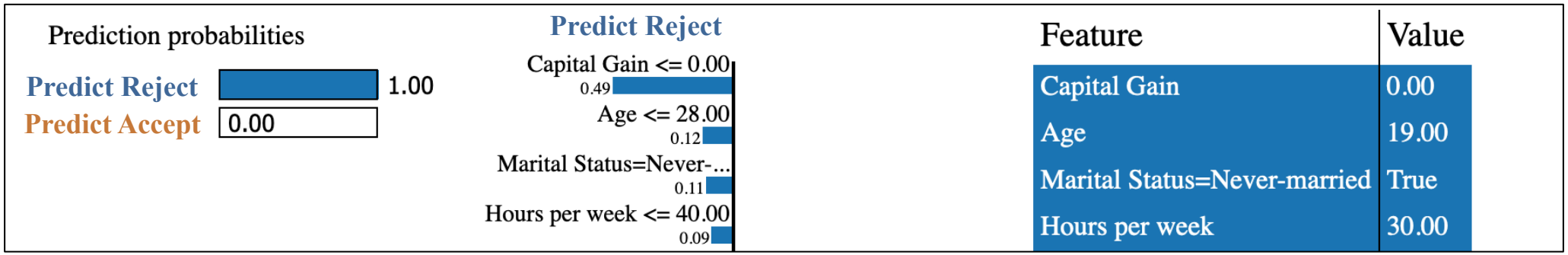
Carlos Guestrin
University of Washington
Seattle, WA 98105, USA
guestrin@cs.uw.edu

https://arxiv.org/pdf/1602.04938.pdf
https://homes.cs.washington.edu/~marcotcr/blog/lime/
https://github.com/marcotcr/lime

Application to tabular data (Adult census dataset — https://www.kaggle.com/uciml/adult-census-income):

```
feature_names = ["Age", "Workclass", "fnlwgt", "Education", "Education-Num", "Marital Status","Occupation", "Relati
onship", "Race", "Sex", "Capital Gain", "Capital Loss","Hours per week", "Country"]
```





2.2 Grad-CAM

Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization

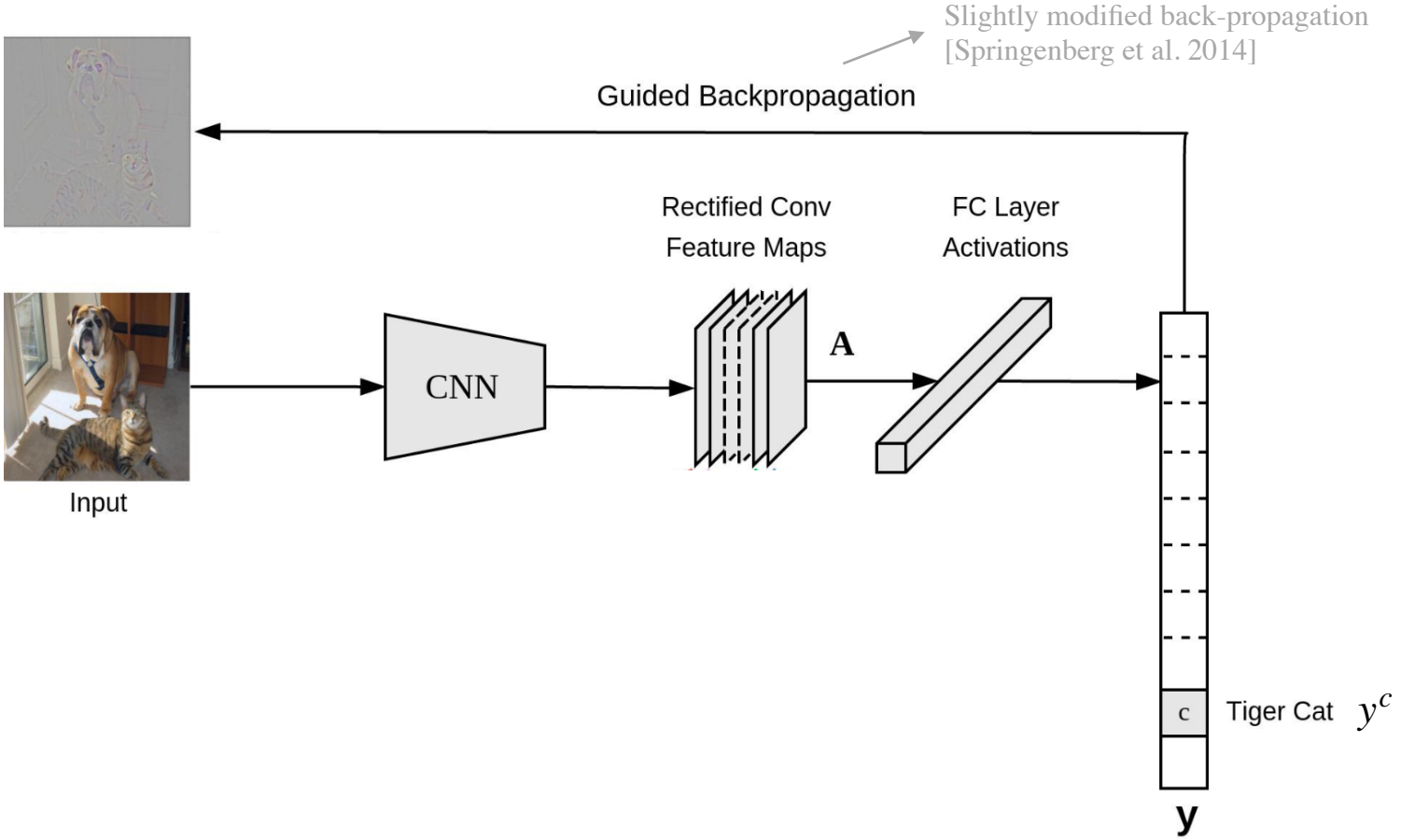
Ramprasaath R. Selvaraju · Michael Cogswell · Abhishek Das · Ramakrishna Vedantam · Devi Parikh · Dhruv Batra
Georgia Institute of Technology, Atlanta, GA, USA
Facebook AI Research, Menlo Park, CA, USA

https://arxiv.org/pdf/1610.02391.pdf
http://gradcam.cloudcv.org/
https://github.com/ramprs/grad-cam/

Instead of back-propagating the gradient of $R_\theta = \sum_i loss_i$ in the hidden layers of the neural-network

→ back-propagate the gradient of an output variable *c*

Compute how y^c is sensitive to the NN inputs.



2.2 Grad-CAM

Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization

Ramprasaath R. Selvaraju · Michael Cogswell · Abhishek Das · Ramakrishna Vedantam · Devi Parikh · Dhruv Batra
Georgia Institute of Technology, Atlanta, GA, USA
Facebook AI Research, Menlo Park, CA, USA

https://arxiv.org/pdf/1610.02391.pdf
http://gradcam.cloudcv.org/
https://github.com/ramprs/grad-cam/

Instead of back-propagating the gradient of $R_\theta = \sum_i loss_i$ in the hidden layers of the neural-network

→ back-propagate the gradient of an output variable *c*

GB for "Cat"

GB for "Dog"

Not that convincing … but a good starting point! → Not class-discriminative but high resolution

Grad-CAM will compute a special mask for this result

2.2 Grad-CAM

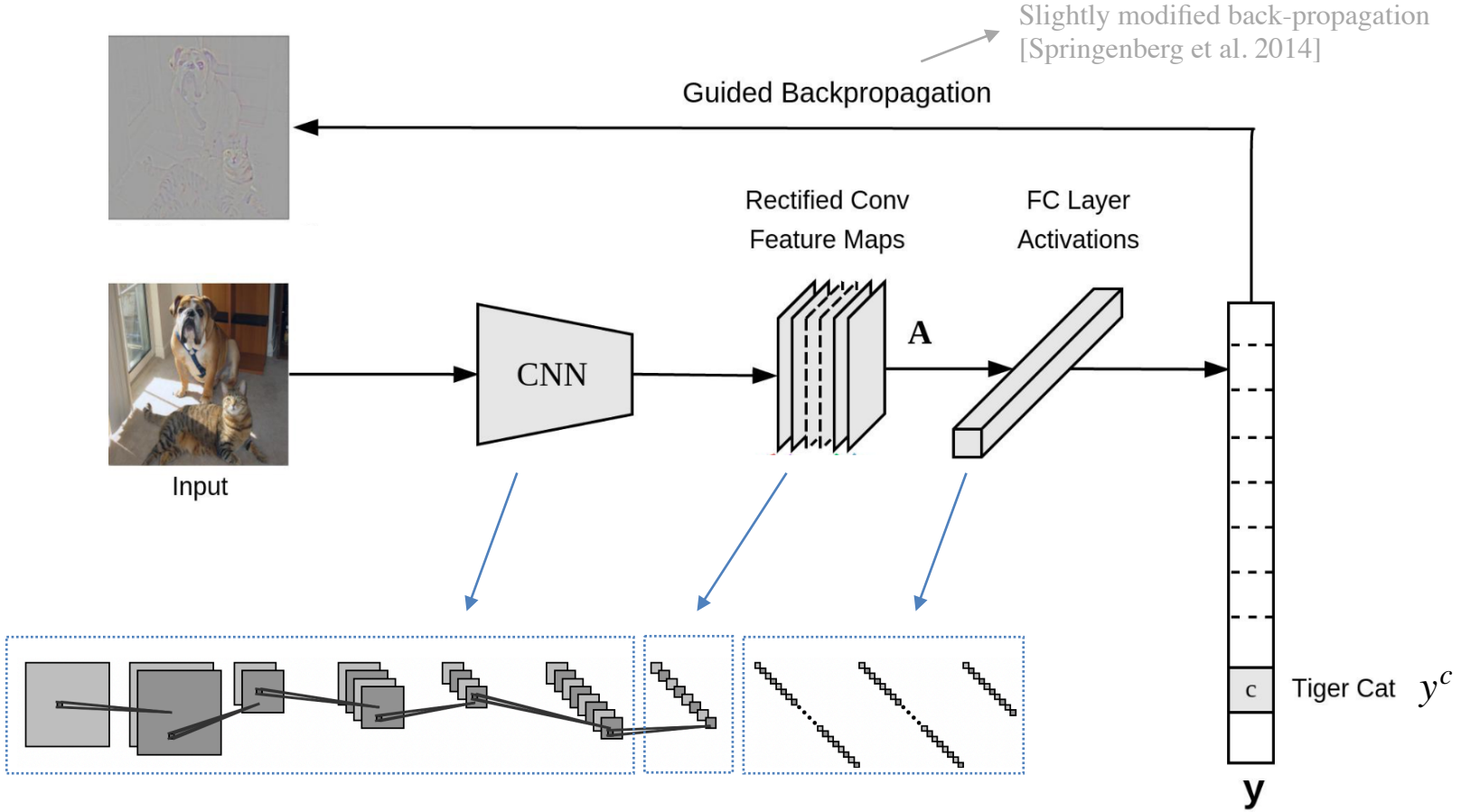
Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization

Ramprasaath R. Selvaraju · Michael Cogswell · Abhishek Das · Ramakrishna Vedantam · Devi Parikh · Dhruv Batra
Georgia Institute of Technology, Atlanta, GA, USA
Facebook AI Research, Menlo Park, CA, USA

https://arxiv.org/pdf/1610.02391.pdf
http://gradcam.cloudcv.org/
https://github.com/ramprs/grad-cam/

Instead of back-propagating the gradient of $R_\theta = \sum_i loss_i$ in the hidden layers of the neural-network

→ back-propagate the gradient of an output variable *c*



Make further hypotheses on the CNN architecture →VGG, ResNet, …

2.2 Grad-CAM

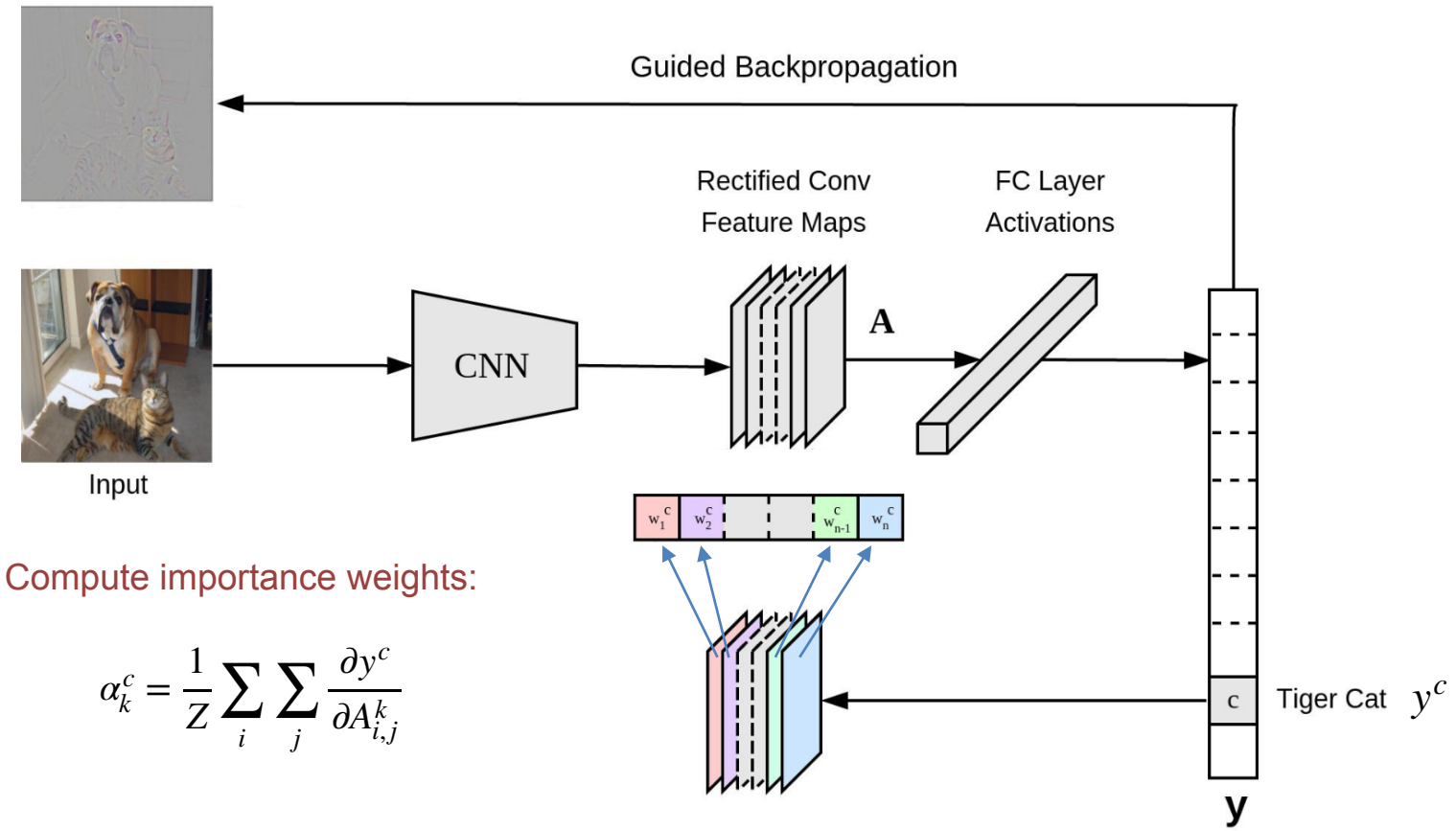
Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization

Ramprasaath R. Selvaraju · Michael Cogswell · Abhishek Das · Ramakrishna Vedantam · Devi Parikh · Dhruv Batra
Georgia Institute of Technology, Atlanta, GA, USA
Facebook AI Research, Menlo Park, CA, USA

https://arxiv.org/pdf/1610.02391.pdf
http://gradcam.cloudcv.org/
https://github.com/ramprs/grad-cam/

To get a more class discriminative Grad-CAM uses the Rectified Convolution Feature Maps $A^k_{i,j}$

(where k is a channel associated to a feature and (i, j) are coordinates in these subsampled images of detected features)



2.2 Grad-CAM

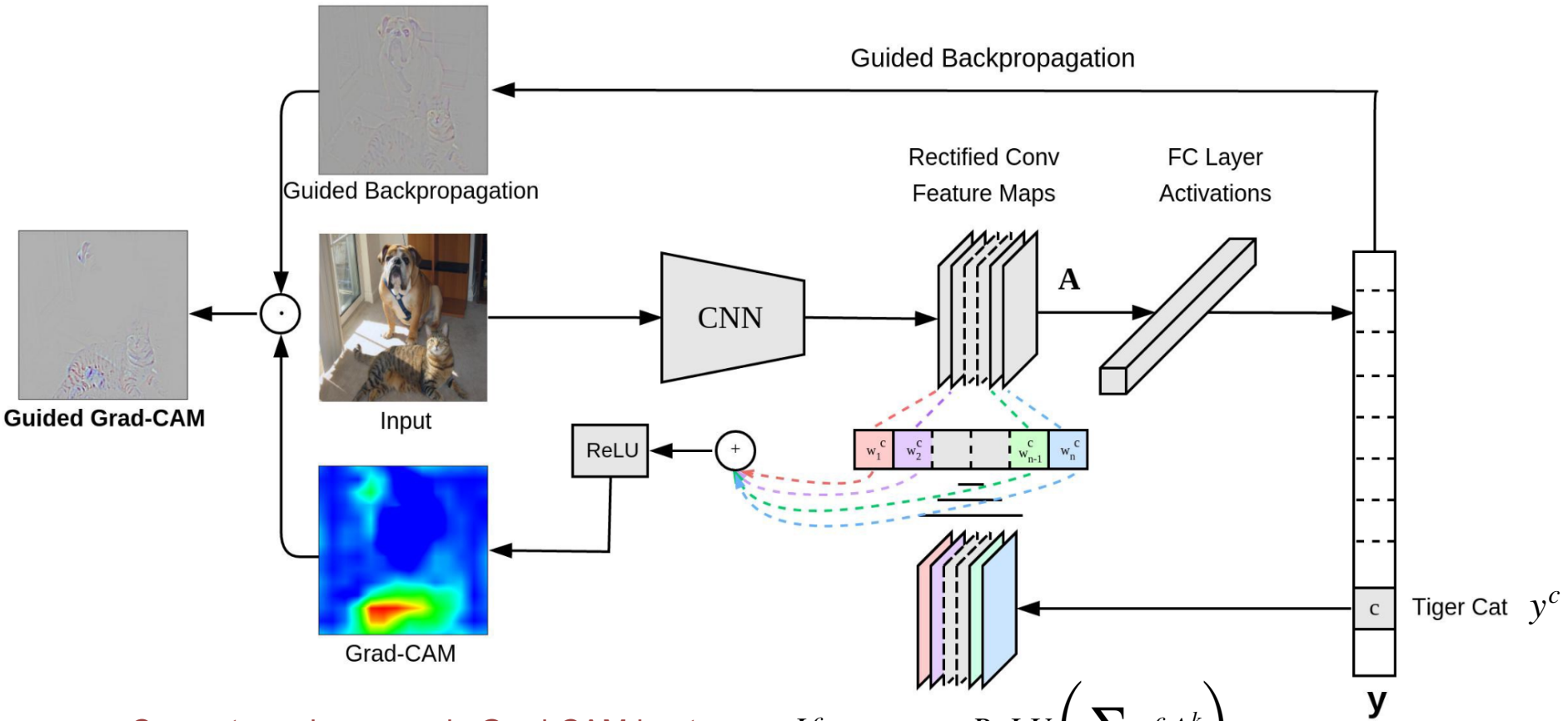
Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization

Ramprasaath R. Selvaraju · Michael Cogswell · Abhishek Das · Ramakrishna Vedantam · Devi Parikh · Dhruv Batra
Georgia Institute of Technology, Atlanta, GA, USA
Facebook AI Research, Menlo Park, CA, USA

https://arxiv.org/pdf/1610.02391.pdf
http://gradcam.cloudcv.org/
https://github.com/ramprs/grad-cam/

To get a more class discriminative Grad-CAM uses the Rectified Convolution Feature Maps $A^k_{i,j}$

(where k is a channel associated to a feature and (i, j) are coordinates in these subsampled images of detected features)



Compute and up-sample Grad-CAM heat map: $L^c_{Grad-CAM} = ReLU\left(\sum_k \alpha^c_k A^k\right)$

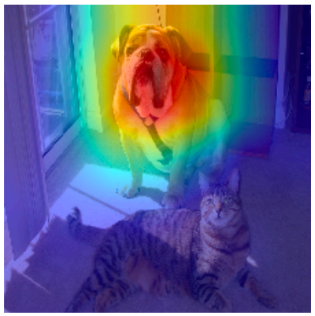
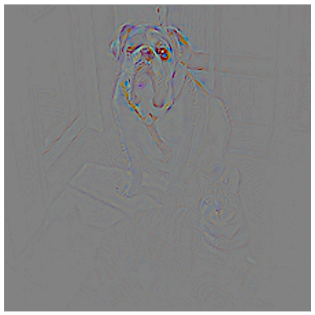
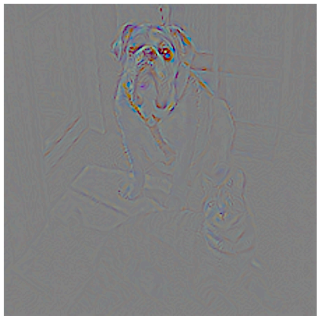
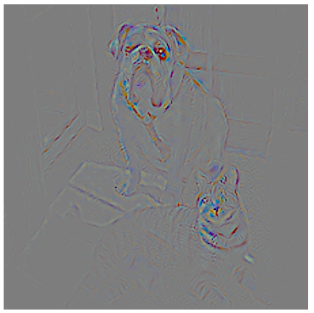
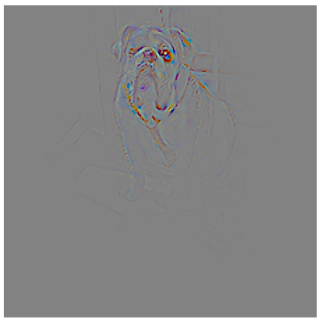
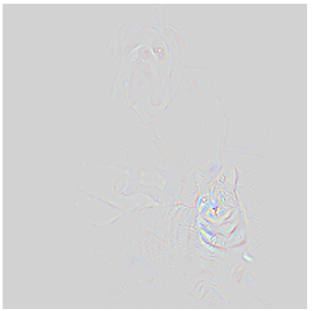
2.2 Grad-CAM

Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization

Ramprasaath R. Selvaraju · Michael Cogswell · Abhishek Das · Ramakrishna Vedantam · Devi Parikh · Dhruv Batra
Georgia Institute of Technology, Atlanta, GA, USA
Facebook AI Research, Menlo Park, CA, USA

https://arxiv.org/pdf/1610.02391.pdf
http://gradcam.cloudcv.org/
https://github.com/ramprs/grad-cam/

Results

Predicted class	#1 boxer	#2 bull mastiff	#3 tiger cat
Grad-CAM [1]			
Guided backpropagation [2]			
Guided Grad-CAM [1]			

2.3 GEMS-AI

Explaining Machine Learning Models using Entropic Variable Projection

François Bachoc[1], Fabrice Gamboa[1,3], Max Halford[2], Jean-Michel Loubes[1,3] and Laurent Risser[1,3]

[1]Institut de Mathématiques de Toulouse
[2] Institut de recherche en informatique de Toulouse
[3] Artificial and Natural Intelligence Toulouse Institute (3IA ANITI)

https://arxiv.org/pdf/1810.07924.pdf

https://gems-ai.aniti.fr/

« What-if machine » for group-explainability

Test set

$\{X_i, Y_i\}_{i=n+1,\dots,n+m}$

with

$X_i = \{X_i^1, \dots, X_i^p\}$

« Black-box » decision rules

X_i^1 →
X_i^2 →
X_i^3 →
… …
X_i^p →

→ $\hat{Y}_i := f_\theta(X_i)$

Intuition :

- Re-weighting the observations $\{X_i, Y_i, \hat{Y}_i\}_{i=n+1,\dots,n+m}$ to emphasise a specific average property of the test set
- Then explain how other properties vary.

2.3 GEMS-AI

Explaining Machine Learning Models using Entropic Variable Projection

François Bachoc[1], Fabrice Gamboa[1,3], Max Halford[2], Jean-Michel Loubes[1,3] and Laurent Risser[1,3]

[1]Institut de Mathématiques de Toulouse
[2] Institut de recherche en informatique de Toulouse
[3] Artificial and Natural Intelligence Toulouse Institute (3IA ANITI)

https://arxiv.org/pdf/1810.07924.pdf

https://gems-ai.aniti.fr/

Example (based on the adult income dataset https://www.kaggle.com/uciml/adult-census-income)

Age (X^1)	**Education.num** (X^2)	**Marital.status** (X^3)	**Hours.per.week** (X^4)	…	**Loan granted — True** (Y)	**Loan granted — Predicted** ($\hat{Y} = f_\theta(X)$)
54	4	Divorced	40		No	No
41	10	Never-married	60		Yes	Yes
51	13	Married-civ	40		Yes	No
39	14	Married-civ	65		Yes	Yes
49	10	Divorced	50		No	Yes
…	…	…	…		…	…

2.3 GEMS-AI

Explaining Machine Learning Models using Entropic Variable Projection

François Bachoc[1], Fabrice Gamboa[1,3], Max Halford[2], Jean-Michel Loubes[1,3] and Laurent Risser[1,3]

[1]Institut de Mathématiques de Toulouse
[2] Institut de recherche en informatique de Toulouse
[3] Artificial and Natural Intelligence Toulouse Institute (3IA ANITI)

https://arxiv.org/pdf/1810.07924.pdf

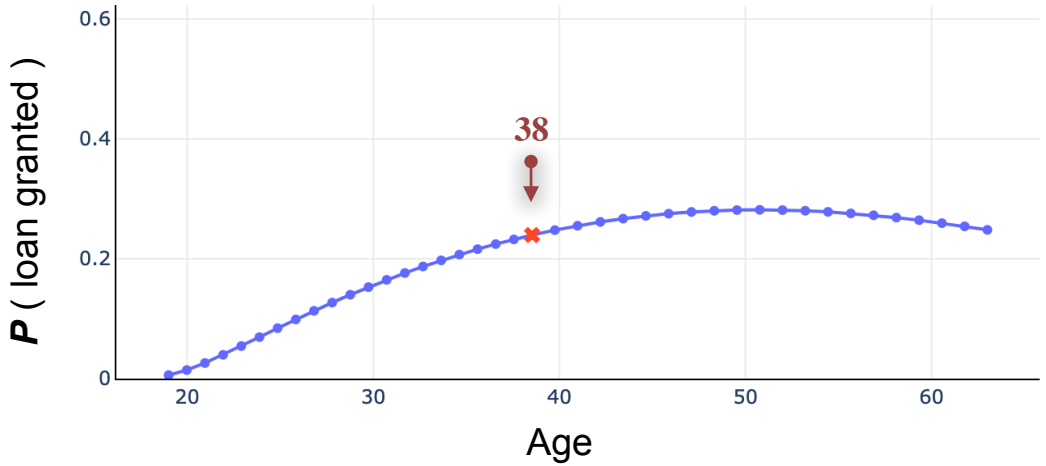
https://gems-ai.aniti.fr/

What-if the average age is 38 instead of 42 in the test set?

Compute optimal weights →

	Age (X^1)	Education.num (X^2)	Marital.status (X^3)	Hours.per.week (X^4)	…	Loan granted — True (Y)	Loan granted — Predicted ($\hat{Y} = f_\theta(X)$)
0.81	54	4	Divorced	40		No	No
1.01	41	10	Never-married	60		Yes	Yes
0.83	51	13	Married-civ	40		Yes	No
1.10	39	14	Married-civ	65		Yes	Yes
0.84	49	10	Divorced	50		No	Yes
…	…	…	…	…		…	…

Compute average properties emphasised by the chosen level of stress



2.3 GEMS-AI

Explaining Machine Learning Models using Entropic Variable Projection

François Bachoc[1], Fabrice Gamboa[1,3], Max Halford[2], Jean-Michel Loubes[1,3] and Laurent Risser[1,3]

[1]Institut de Mathématiques de Toulouse
[2] Institut de recherche en informatique de Toulouse
[3] Artificial and Natural Intelligence Toulouse Institute (3IA ANITI)

https://arxiv.org/pdf/1810.07924.pdf

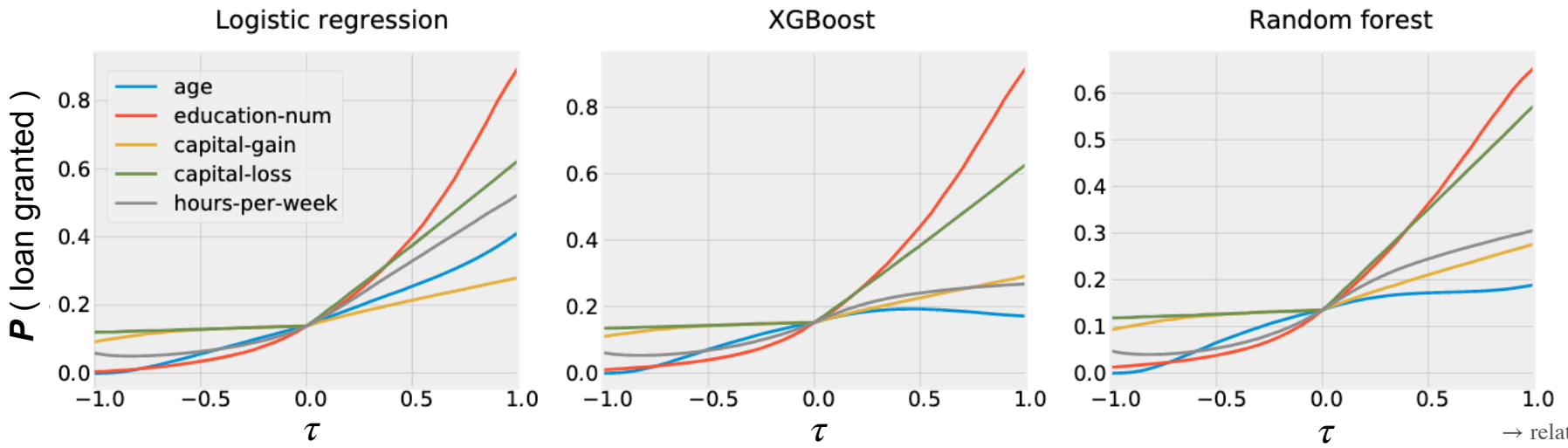
https://gems-ai.aniti.fr/

What-if the average […] is […] instead of [original average value] in the test set? → **Predictions == 1**

Compute optimal weights

… then explain

	Age (X^1)	Education.num (X^2)	Marital.status (X^3)	Hours.per.week (X^4)	…	Loan granted — True (Y)	Loan granted — Predicted ($\hat{Y} = f_\theta(X)$)
…	54	4	Divorced	40		No	No
…	41	10	Never-married	60		Yes	Yes
…	51	13	Married-civ	40		Yes	No
…	39	14	Married-civ	65		Yes	Yes
…	49	10	Divorced	50		No	Yes
…	…	…	…	…		…	…



→ related to quantiles

2.3 GEMS-AI

Explaining Machine Learning Models using Entropic Variable Projection

François Bachoc[1], Fabrice Gamboa[1,3], Max Halford[2], Jean-Michel Loubes[1,3] and Laurent Risser[1,3]

[1]Institut de Mathématiques de Toulouse
[2] Institut de recherche en informatique de Toulouse
[3] Artificial and Natural Intelligence Toulouse Institute (3IA ANITI)

https://arxiv.org/pdf/1810.07924.pdf

https://gems-ai.aniti.fr/

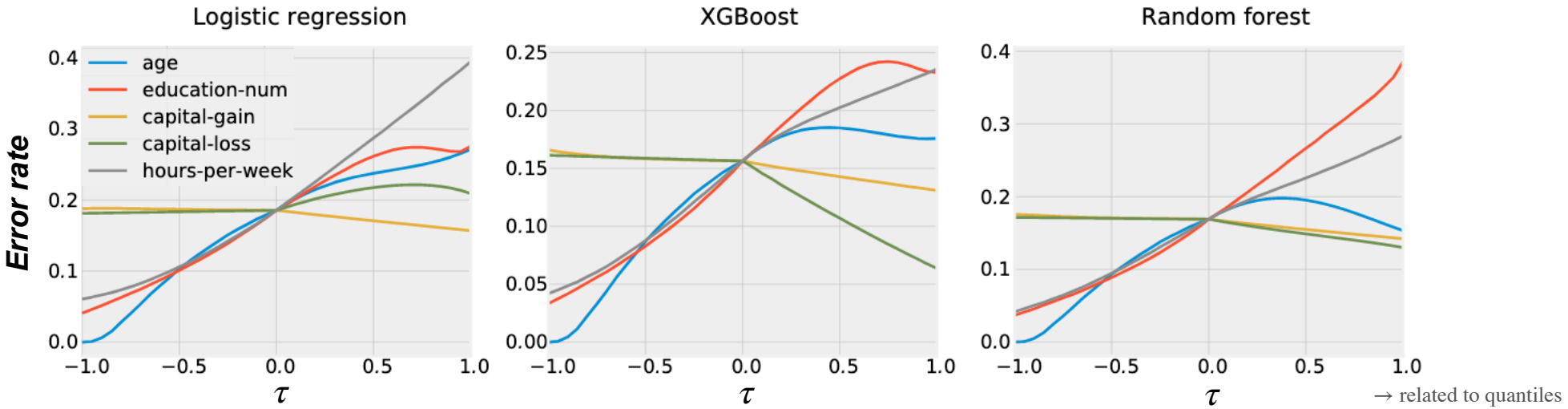
What-if the average […] is […] instead of [original average value] in the test set? → **Error rate**

Compute optimal weights

→

… then explain

	Age (X^1)	Education.num (X^2)	Marital.status (X^3)	Hours.per.week (X^4)	…	Loan granted — True (Y)	Loan granted — Predicted ($\hat{Y} = f_\theta(X)$)
…	54	4	Divorced	40		No	No
…	41	10	Never-married	60		Yes	Yes
…	51	13	Married-civ	40		Yes	No
…	39	14	Married-civ	65		Yes	Yes
…	49	10	Divorced	50		No	Yes
…	…	…	…	…		…	…

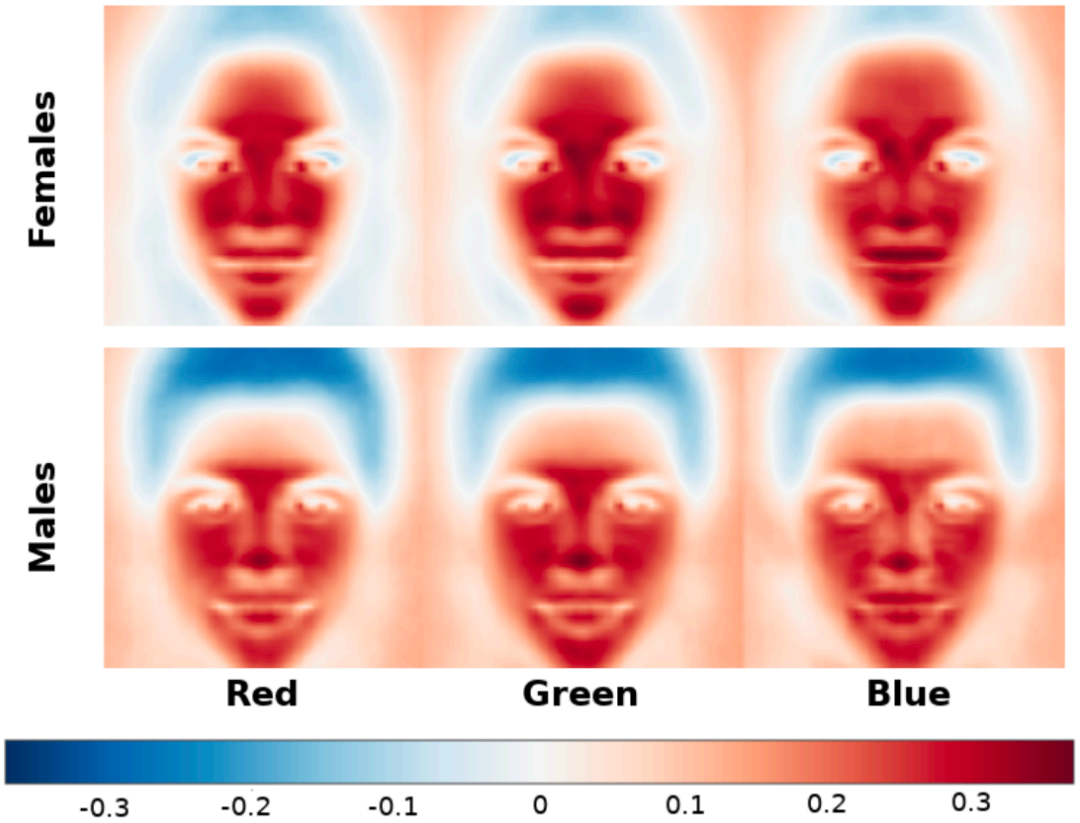


→ related to quantiles

2.3 GEMS-AI

ResNet 18 CNN trained to predict who is attractive → 87% of accurate predictions on the test set

What-if the average pixel intensities are locally higher or lower → Predictions == Attractive



Average pixel influences to predict whether someone is attractive or not by distinguishing males and females

- Explainability has become an important topic in trustworthy machine-learning.
- In some contexts, it is made mandatory because of legal constraints
- In other contexts, it helps validating that the decision are made for good reasons
- Various solutions exist
- *Concept-based explanations* and *hybrid models* are interesting perspectives for explanations based on the hidden data representations