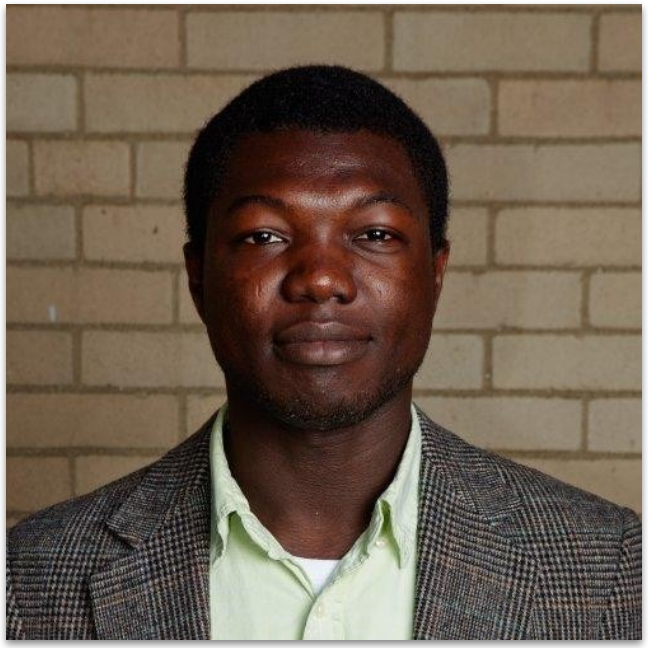
Explaining Machine Learning Predictions: State-of-the-art, Challenges, Opportunities

Hima Lakkaraju Julius Adebayo Sameer Singh

https://explainml-tutorial.github.io

With some comments and additional slides by P. Geurts

Hima Lakkaraju
Harvard University



Julius Adebayo
MIT

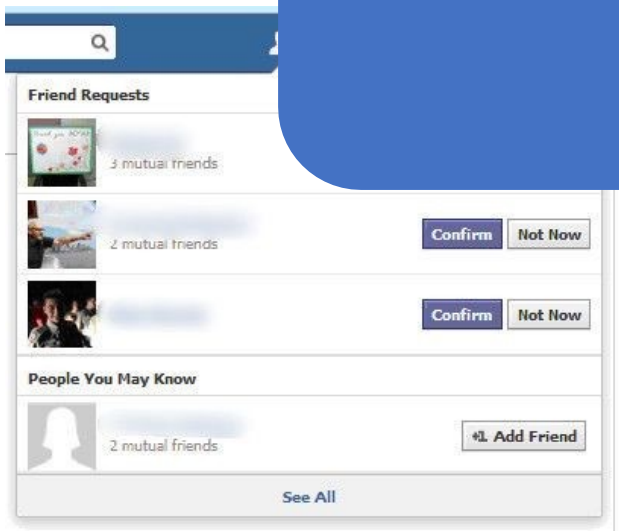
Sameer Singh
UC Irvine

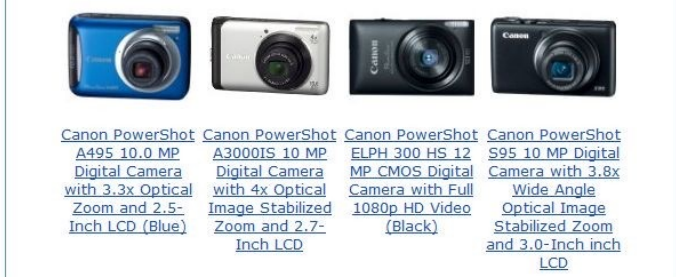
Slides and Video: explainml-tutorial.github.io

[Weller 2017]

Motivation

Machine Learning is EVERYWHERE!!





Motivation

Model understanding is absolutely critical in several domains -- particularly those involving *high stakes decisions*!

Motivation: Why Model Understanding?

Input

This model is relying on incorrect features to make ion!! Let model

Model understanding facilitates debugging

Predictive Model

Prediction = Siberian Husky

[Larson et. al. 2016]

Motivation: Why Model Understanding?

Defendant Details

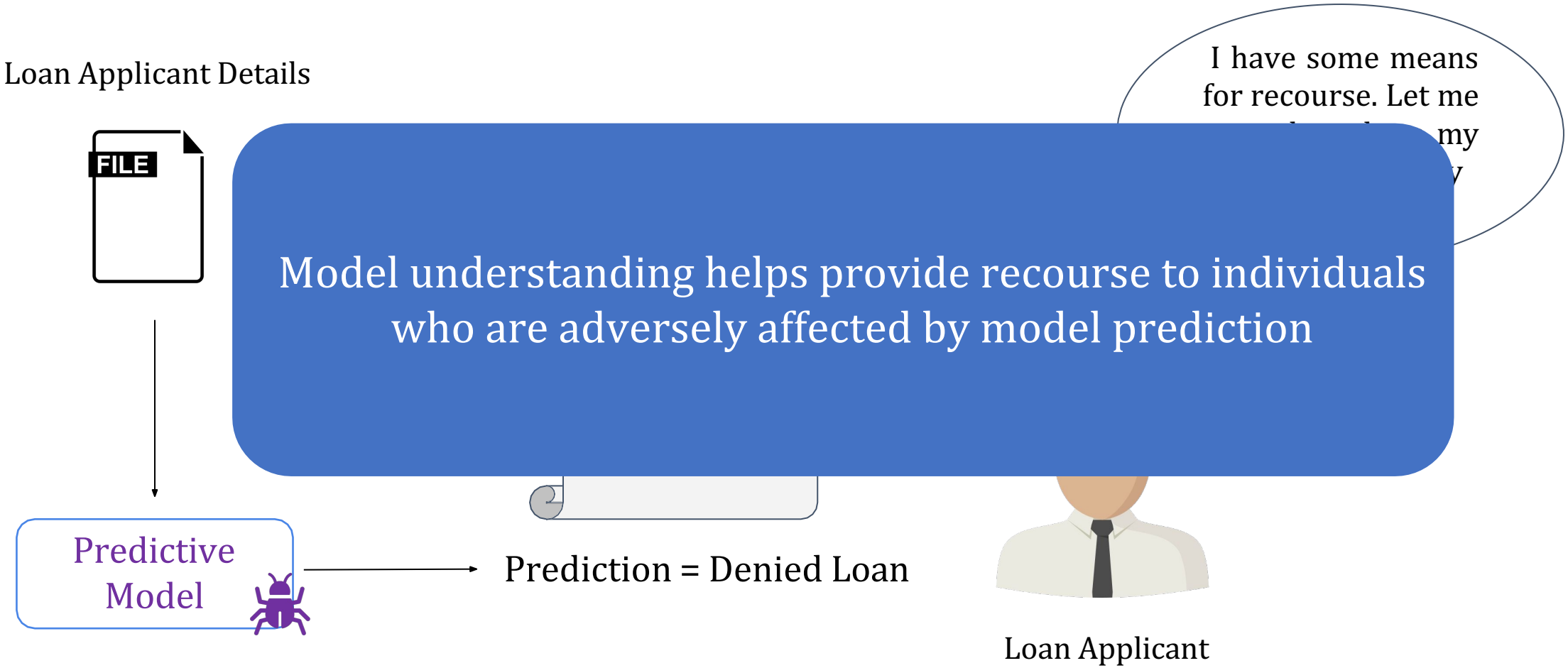
Predictive Model

Prediction = Risky to Release

This prediction is biased. Race and gender are being ake the ion

Model understanding facilitates bias detection

Motivation: Why Model Understanding?



Motivation: Why Model Understanding?

Patient Data

25, Female, Cold
32, Male, No
31, Male, Cou
.
.
.
.

Predictive Model

Model Understanding

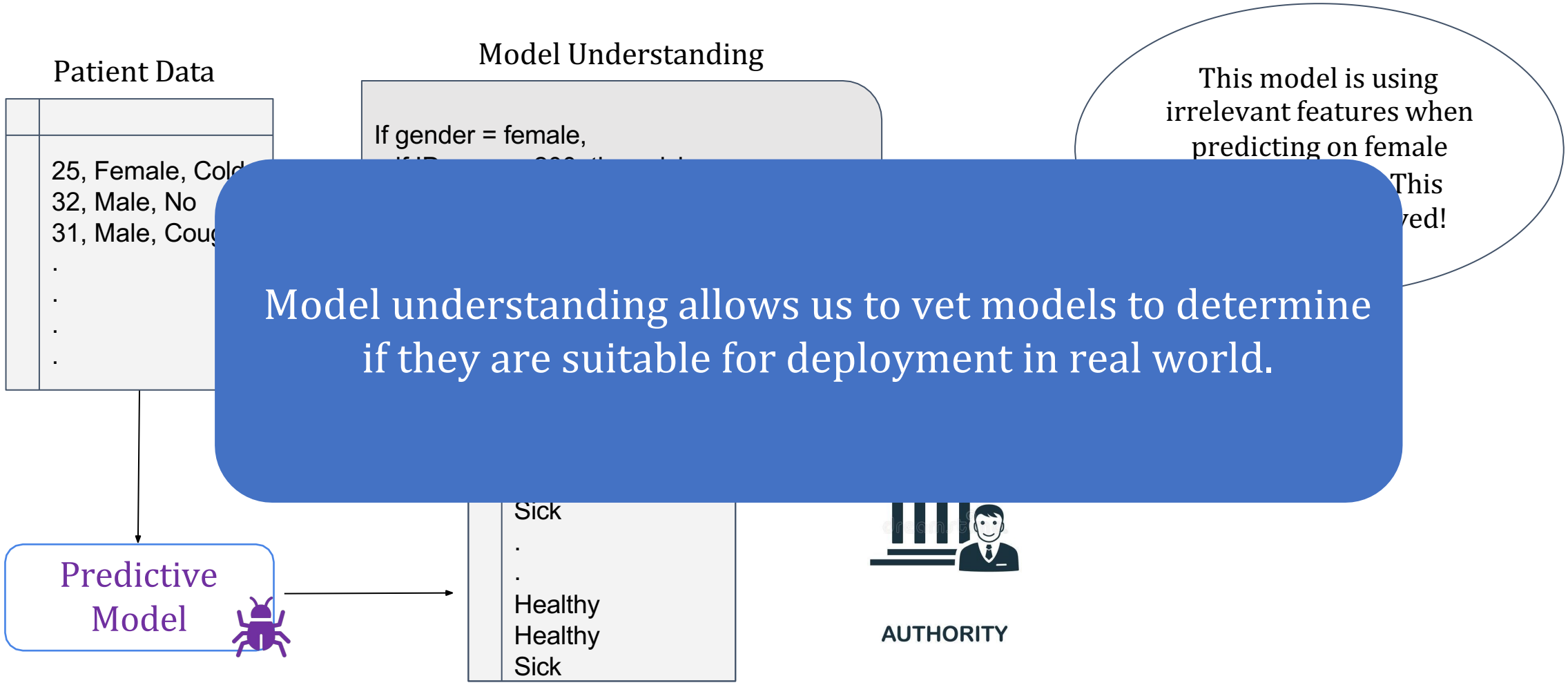
If gender = female,

Sick
.
.
Healthy
Healthy
Sick

This model is using irrelevant features when predicting on female
I should
tions
.

Model understanding helps assess if and when to trust model predictions when making decisions.

Motivation: Why Model Understanding?



Motivation: Why Model Understanding?

Utility

Debugging

Bias Detection

Recourse

If and when to trust model predictions

Vet models to assess suitability for deployment

Stakeholders

End users (e.g., loan applicants)

Decision makers (e.g., doctors, judges)

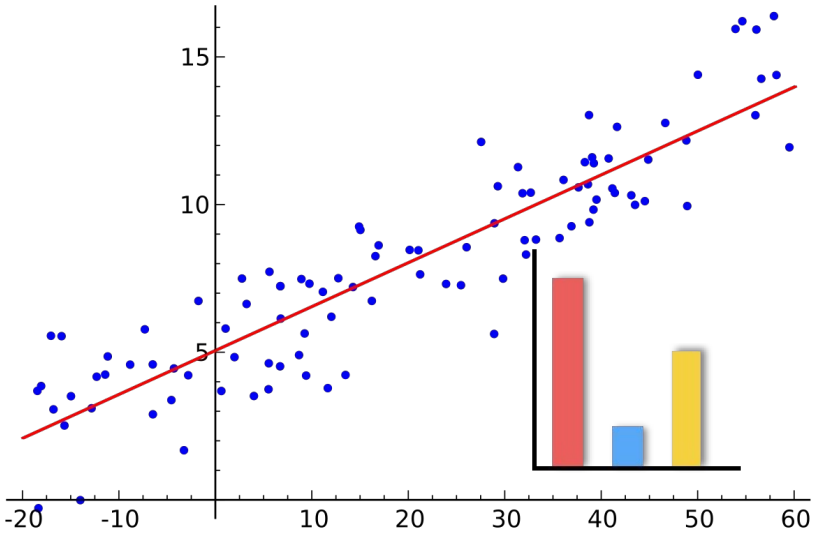
Regulatory agencies (e.g., FDA, European commission)

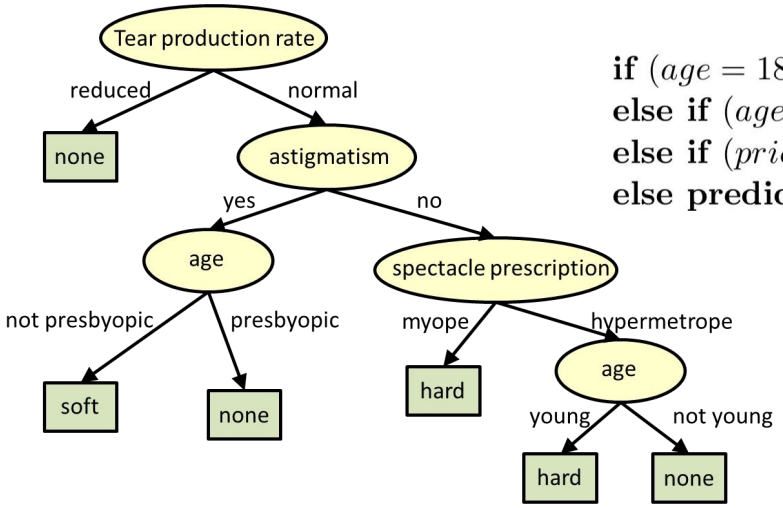
Researchers and engineers

[Letham and Rudin 2015; Lakkaraju et. al. 2016]

Achieving Model Understanding

Take 1: Build *inherently interpretable* predictive models



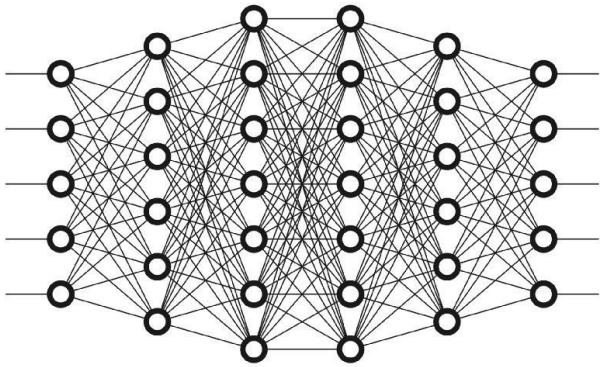


if $(age = 18 - 20)$ **and** $(sex = male)$ **then predict** yes
else if $(age = 21 - 23)$ **and** $(priors = 2 - 3)$ **then predict** yes
else if $(priors > 3)$ **then predict** yes
else predict no

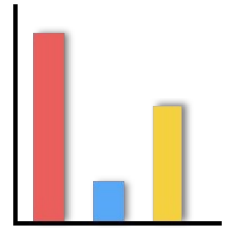
[Ribeiro et. al. 2016, Ribeiro et al. 2018; Lakkaraju et. al. 2019]

Achieving Model Understanding

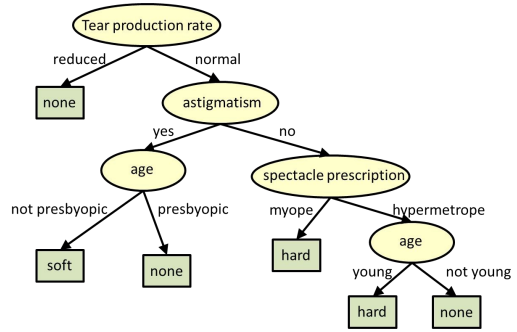
Take 2: *Explain* pre-built models *in a post-hoc manner*



Explainer



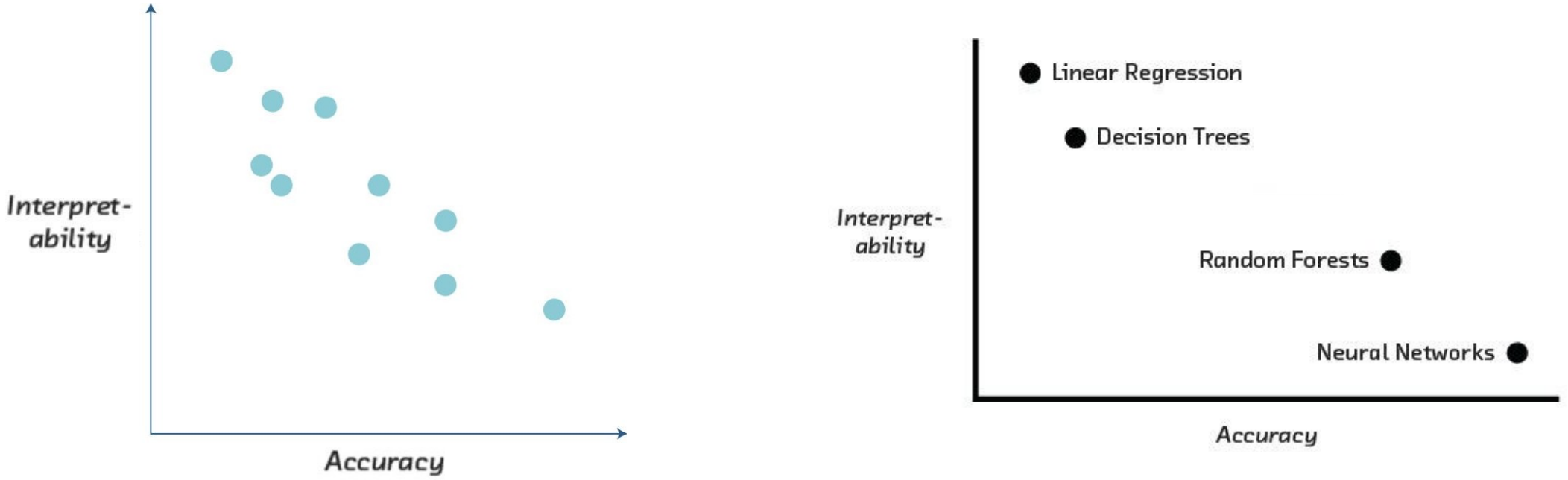
if $(age = 18 - 20)$ **and** $(sex = male)$ **then predict** yes
else if $(age = 21 - 23)$ **and** $(priors = 2 - 3)$ **then predict** yes
else if $(priors > 3)$ **then predict** yes
else predict no



[Cireșan et. al. 2012, Caruana et. al. 2006, Frosst et. al. 2017, Stewart 2020]

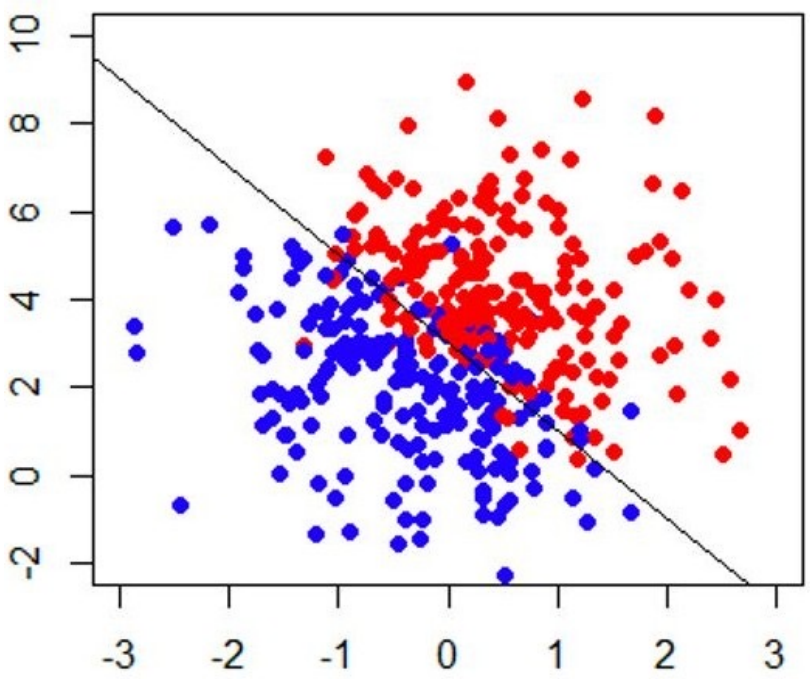
Inherently Interpretable Models vs. Post hoc Explanations

Example

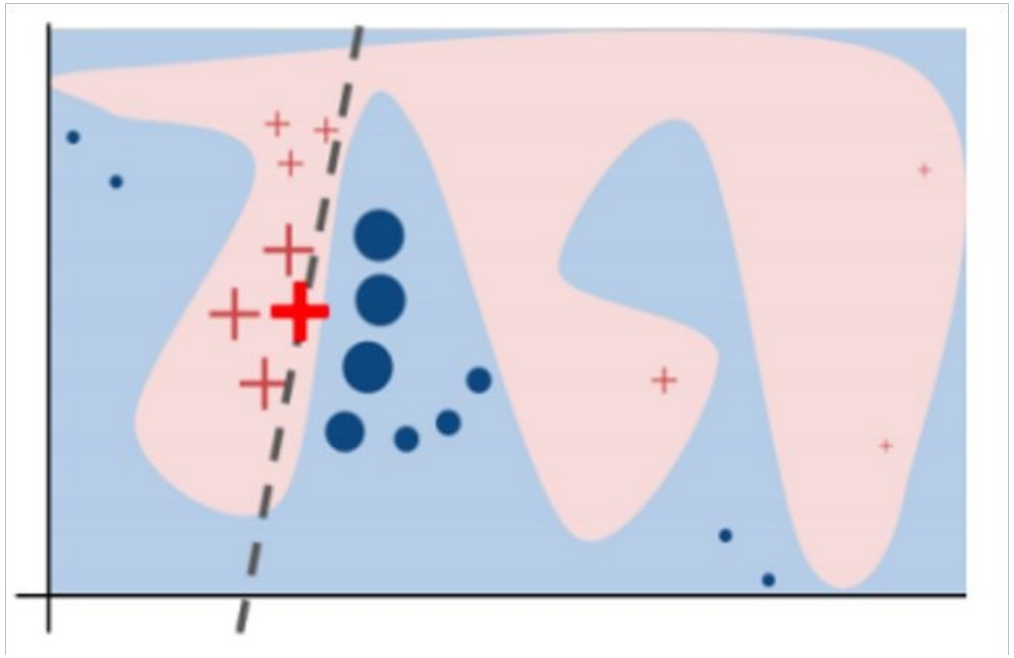


In ***certain*** settings, *accuracy-interpretability trade offs* may exist.

Inherently Interpretable Models vs. Post hoc Explanations



can build interpretable + accurate models



complex models might achieve higher accuracy

[Ribeiro et. al. 2016]

Inherently Interpretable Models vs. Post hoc Explanations

Sometimes, you don't have enough data to build your model from scratch.

And, all you have is a (proprietary) black box!

[Rudin 2019]

Inherently Interpretable Models vs. Post hoc Explanations

If you *can build* an interpretable model which is also adequately accurate for your setting, DO IT!

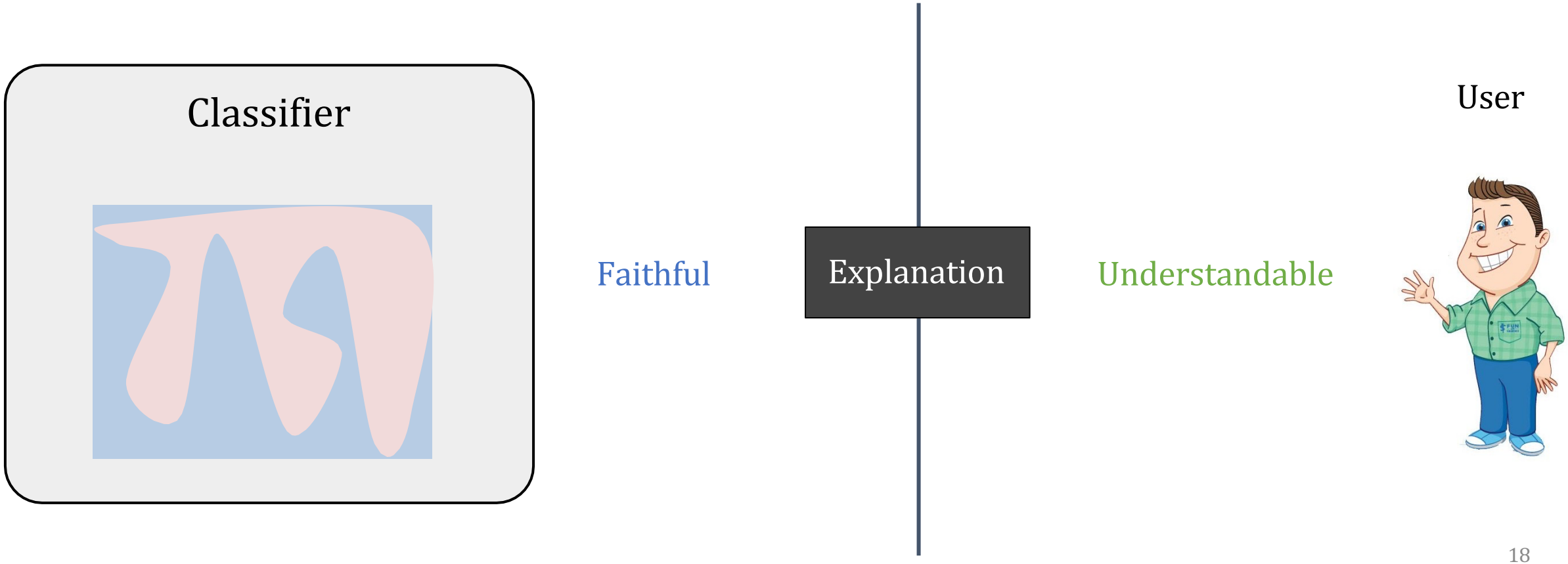
Otherwise, ***post hoc explanations*** come to the rescue!

This tutorial will focus on post hoc explanations!

What is an Explanation?

What is an Explanation?

Definition: Interpretable description of the model behavior



[Lipton 2016]

What is an Explanation?

Definition: Interpretable description of the model behavior

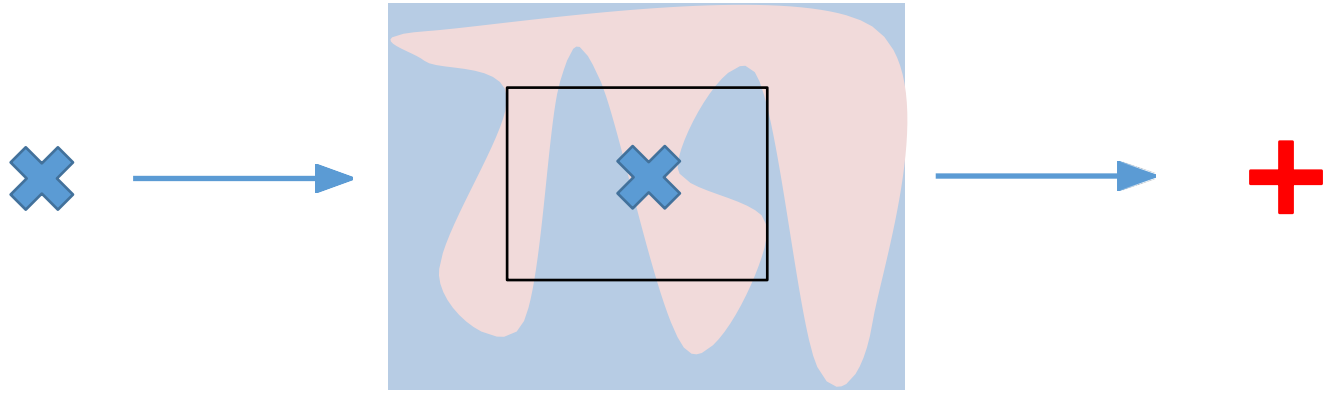
Classifier

- Send all the model parameters θ?
- Send many example predictions?
- Summarize with a program/rule/tree
- Select most important features/points
- Describe how to *flip* the model prediction
- …

User

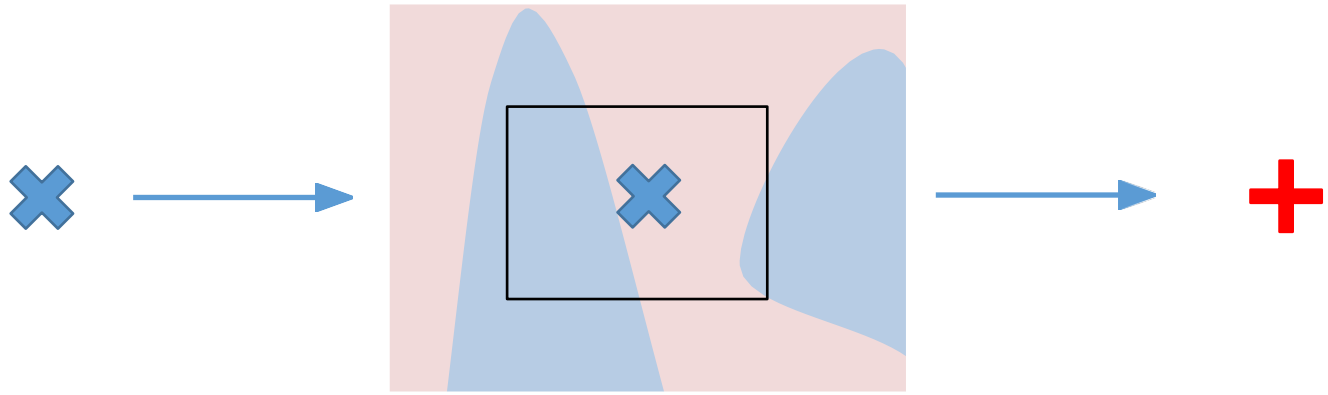
Local versus Global Explanations

Global explanation may be too complicated



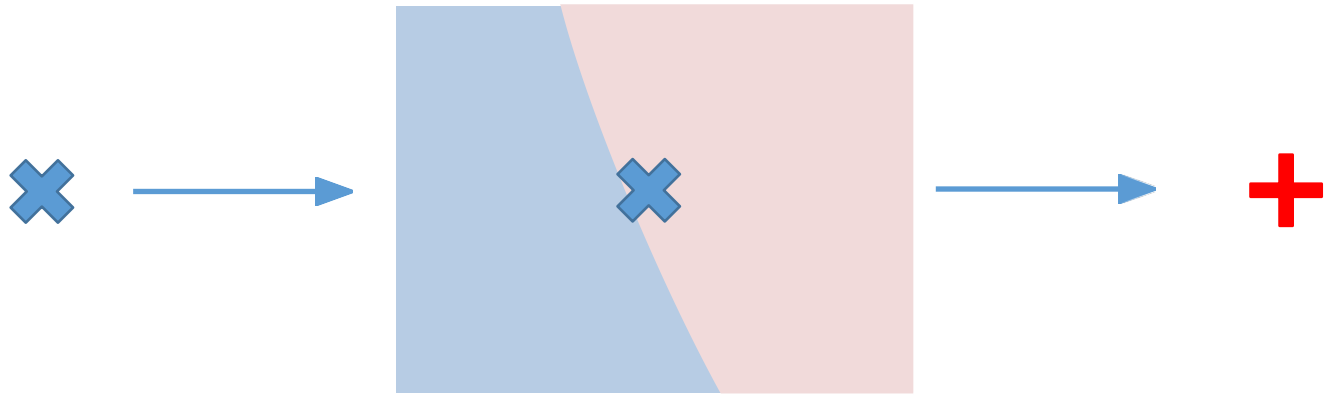
Local versus Global Explanations

Global explanation may be too complicated



Local versus Global Explanations

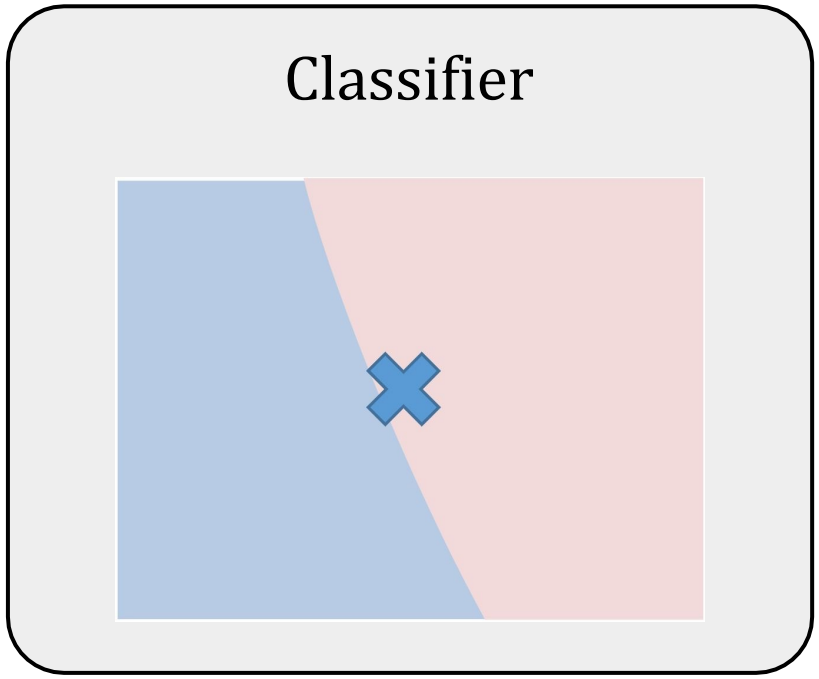
Global explanation may be too complicated



Definition: Interpretable description of the model behavior *in a target neighborhood*.

Local Explanations

Definition: Interpretable description of the model behavior *in a target neighborhood*.



Send many example predictions?

Summarize with a program/rule/tree

Select most important features/points

Describe how to *flip* the model prediction

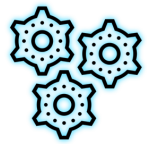
...

User

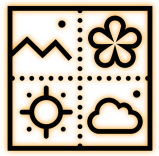
Local Explanations vs. Global Explanations

Local Explanations	Global Explanations
Explain individual predictions	Explain complete behavior of the model
Help unearth biases in the *local neighborhood* of a given instance	Help shed light on *big picture biases* affecting larger subgroups
Help vet if individual predictions are being made for the right reasons	Help vet if the model, at a high level, is suitable for deployment

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



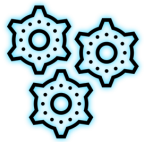
Explanations in Different Modalities

Evaluation of Explanations

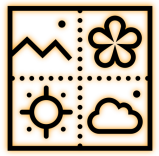
Limits of Post hoc Explainability

Future of Post hoc Explainability

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



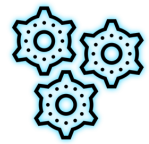
~~Explanations in Different Modalities~~

Evaluation of Explanations

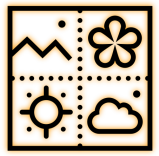
Limits of Post hoc Explainability

Future of Post hoc Explainability

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



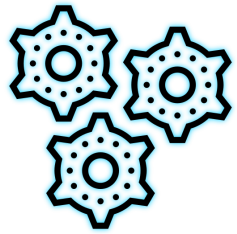
~~Explanations in Different Modalities~~

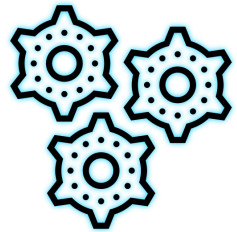
Evaluation of Explanations

Limits of Post hoc Explainability

Future of Post hoc Explainability

Approaches for Post hoc Explainability





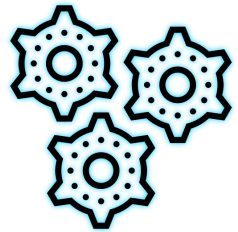
Approaches for Post hoc Explainability

Local Explanations

- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals



Approaches for Post hoc Explainability

Local Explanations

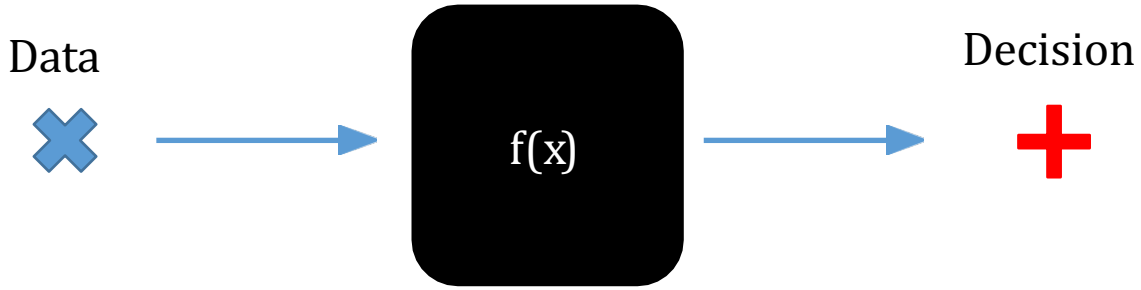
- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

Being Model-Agnostic…

No access to the internal structure…



Not restricted to specific models

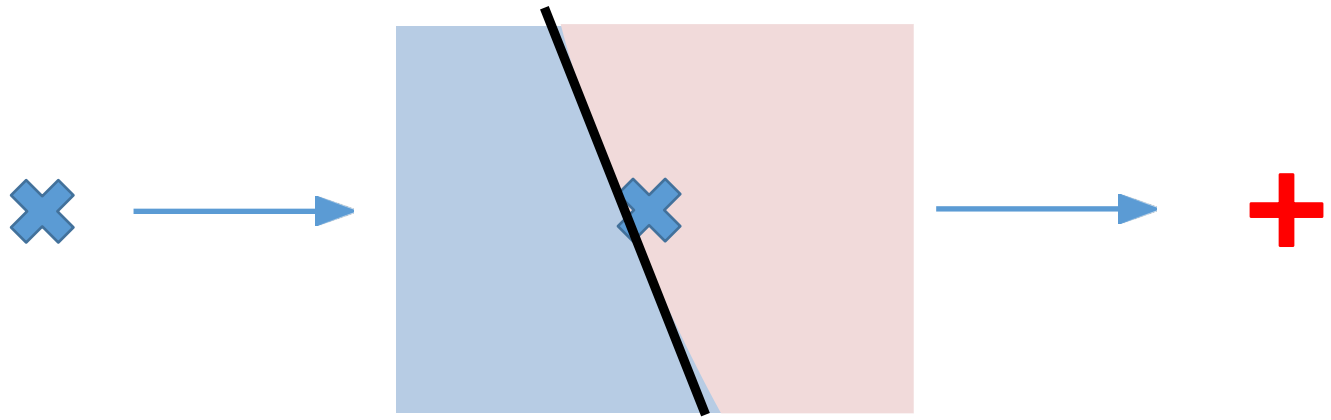
Practically easy: not tied to PyTorch, Tflow, etc.

Study models that you don’t have access to!

[Ribeiro et al. 2016]

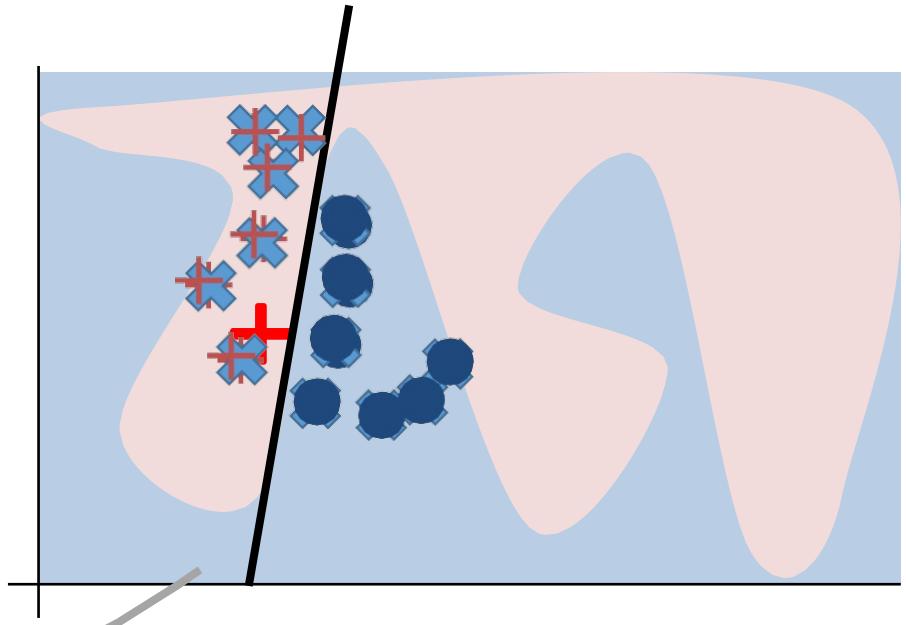
LIME: Sparse, Linear Explanations

Identify the important dimensions,
and present their relative importance



LIME: Sparse Linear Explanations

1. Sample points around x_i
2. Use model to predict labels for each sample
3. Weigh samples according to distance to x_i
4. Learn simple model on weighted samples
5. Use simple model to explain

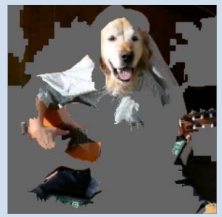


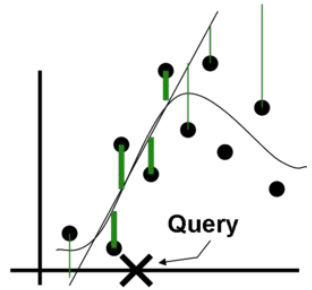
[Ribeiro et al. 2016]

LIME Example - Images

Original Image

P(labrador) = 0.21

Perturbed Instances	P(Labrador)
	0.92
	0.001
	0.34



Locally weighted regression

Explanation

Maybe to a fault?

Yes!

LIME is quite customizable:

- How to perturb?
- Distance/similarity?
- How *local* you want it to be?
- How to express explanation

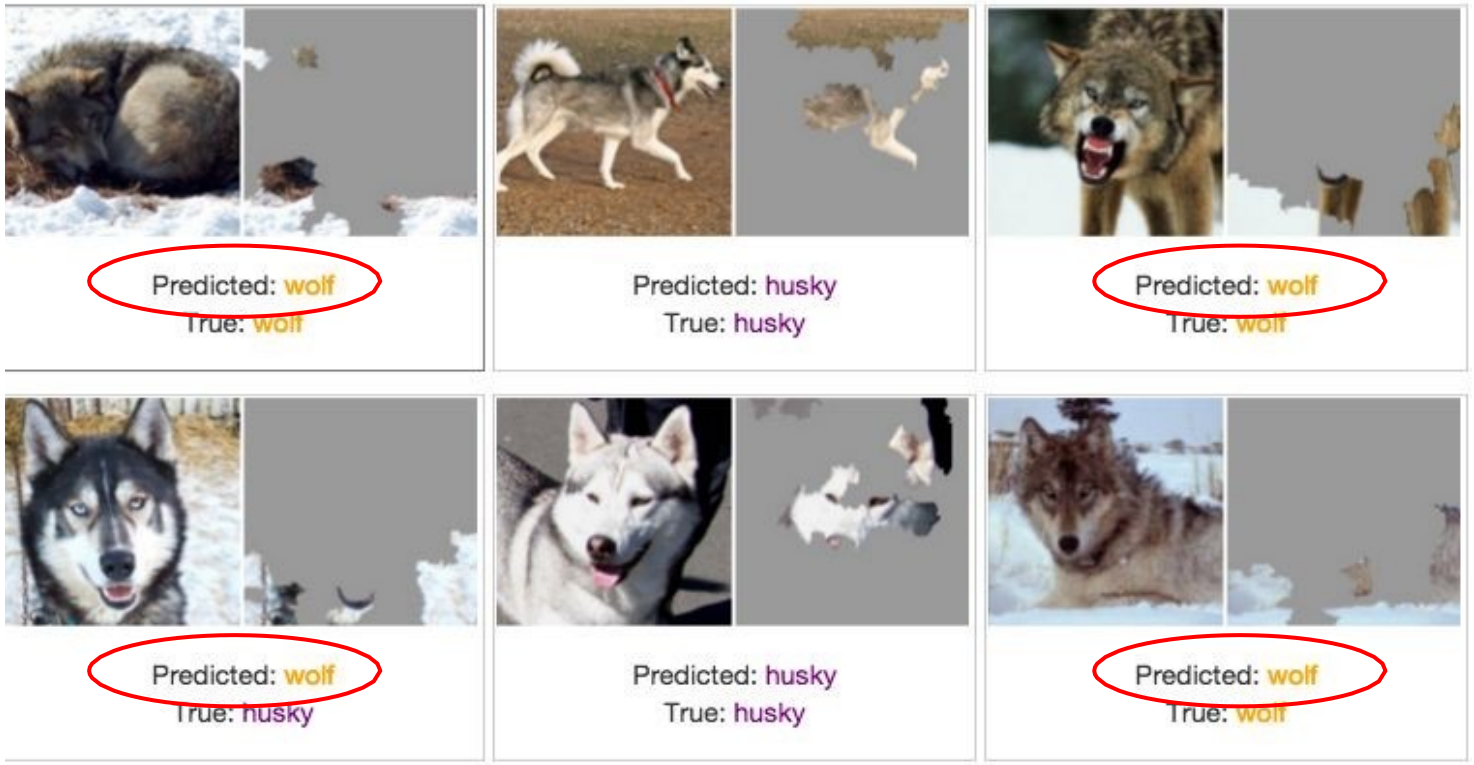
[Ribeiro et al. 2016]

Predict Wolf vs Husky

Only 1 mistake!

[Ribeiro et al. 2016]

Predict Wolf vs Husky

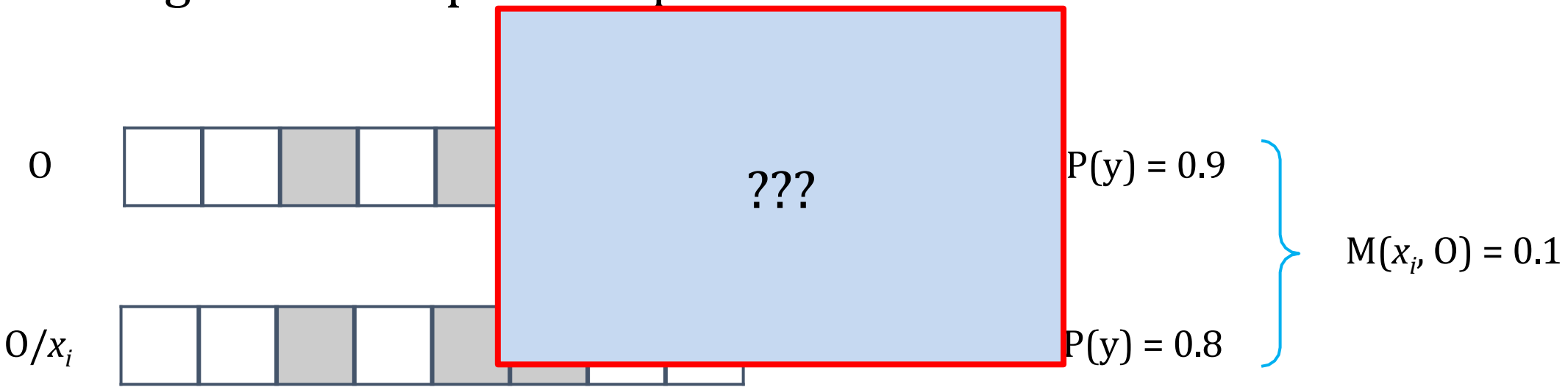


We've built a great snow detector…

[Lundberg & Lee 2017]

SHAP: Shapley Values as Importance

Marginal contribution of each feature towards the prediction, averaged over all possible permutations.



Fairly attributes the prediction to all the features.

Shapley value (Shapley, 1953, Young, 1985, Lundberg and Lee, 2017)

Shapley value is a way to attribute in a fair way the gain of a **game** to several **cooperating** players

Assuming p players $V = \{X_1, \ldots, X_p\}$ and a **characteristic function** $v : 2^V \to \mathbb{R}$, with $v(\emptyset) = 0$, assessing each possible subset of players, the **shapley value** distribution is defined as:

$$\Phi_v(X_m) = \sum_{S \subseteq V^{-m}} \frac{|S|!(p - |S| - 1)!}{p!}(v(S \cup \{X_m\}) - v(S)).$$

It is the **unique** distribution such that:

- $\sum_{X_m} \Phi_v(X_m) = v(V)$ — **Efficiency**
- $\Phi_v(X_m) = 0$ if $v(S \cup \{X_m\}) = v(S)$ for all $S \subseteq V^{-m}$ — **Null player**
- $\Phi_v(X_i) = \Phi_v(X_j)$ if $v(S \cup \{X_i\}) = v(S \cup \{X_j\})$ for all $S \subseteq V^{-i,j}$ — **Symmetry**
- $\Phi_v(X_m) \geq \Phi_w(X_m)$ if $(v(S \cup \{X_m\}) - v(S)) \geq (w(S \cup \{X_m\}) - w(S))$ $\forall S \subseteq V^{-m}$ — **Strong monotonicity**

Shapley value (Shapley, 1953, Young, 1985, Lundberg and Lee, 2017)

An example with three players:

$$v(\{X_1\}) = 50, v(\{X_2\}) = 60, v(\{X_3\}) = 80$$
$$v(\{X_1, X_2\}) = 70, v(\{X_1, X_3\}) = 100, v(\{X_2, X_3\}) = 90$$
$$v(\{X_1, X_2, X_3\}) = 120$$

How to distribute the gain? Which player should receive the most?

$$\begin{aligned}\phi_v(X_1) =& \frac{1}{3}(v(\{X_1\}) - v(\emptyset)) \\ &+ \frac{1}{6}(v(\{X_1, X_2\}) - v(\{X_2\})) + (v(\{X_1, X_3\}) - v(\{X_3\})) \\ &+ \frac{1}{3}(v(\{X_1, X_2, X_3\}) - v(\{X_2, X_3\}))\end{aligned}$$

$$\Rightarrow \phi_v(X_1) = 31,6667, \phi_v(X_2) = 31,6667, \phi_v(X_3) = 56,6667$$

SHAP - Shapley Additive exPlanations

Use Shapley value to attribute model prediction $\hat{f}(x)$ at x *fairly* among the features

$$v(S) = \hat{f}_S(x_S) - \hat{f}_\emptyset(x_\emptyset), \text{ with } \hat{f}_S(x) = E\{\hat{f}(X)|X_S = x_S\}$$

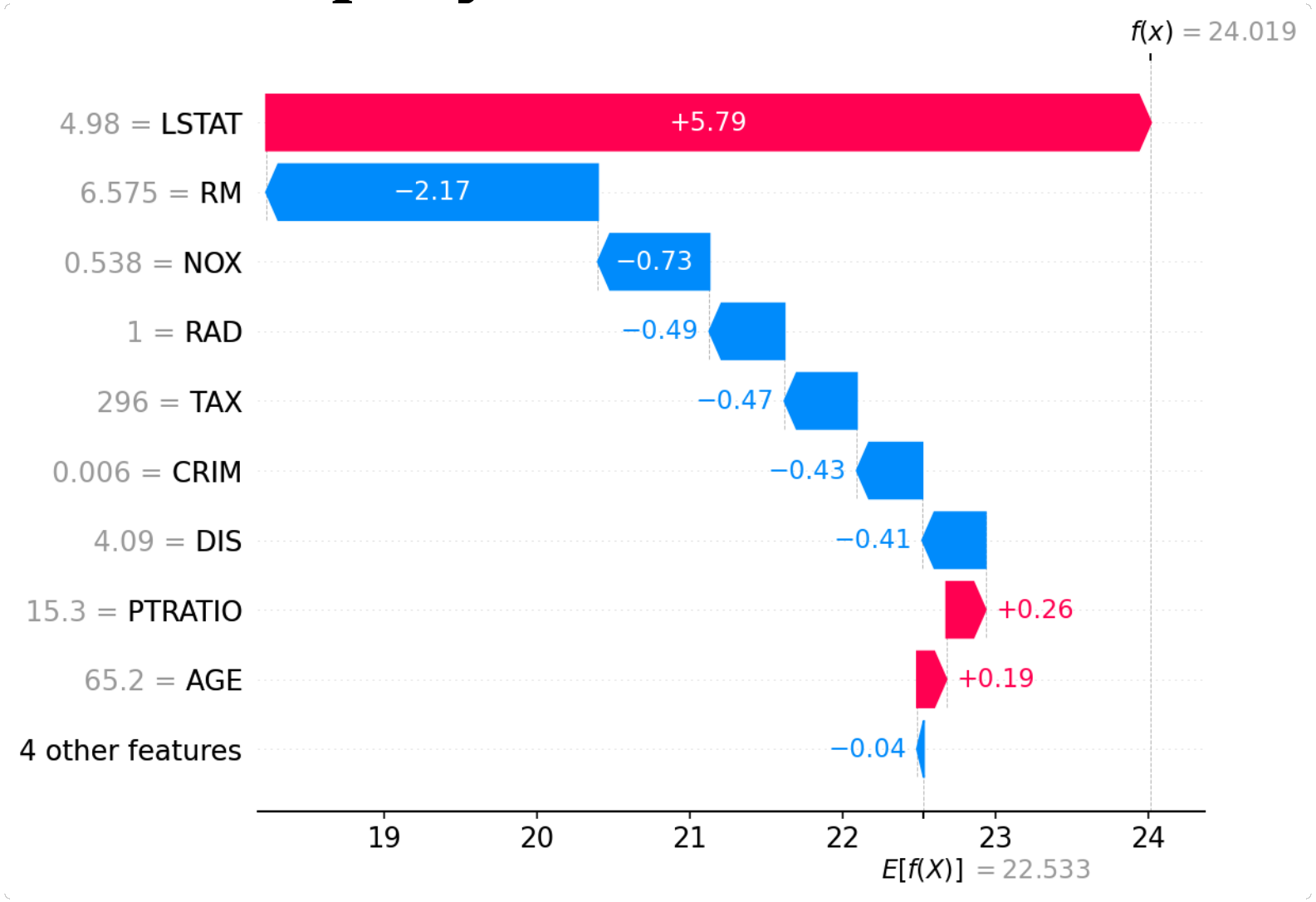
$$\Rightarrow \hat{f}(x) = E\{\hat{f}(X)\} + \phi_v(X_1) + \phi_v(X_2) + \ldots + \phi_v(X_p)$$

Two challenges:

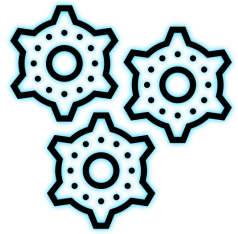
- Estimate $\hat{f}_S(X)$ (often approximated assuming $P(X_{V\setminus S}|X_S) = P(X_{V\setminus S})$)
- Compute the sum over all subsets *S* (MC sampling)

⇒ Several techniques exist to efficiently compute Shapley values for different models (neural networks, trees, etc.)

SHAP - Shapley Additive exPlanations



https://github.com/slundberg/shap



Approaches for Post hoc Explainability

Local Explanations

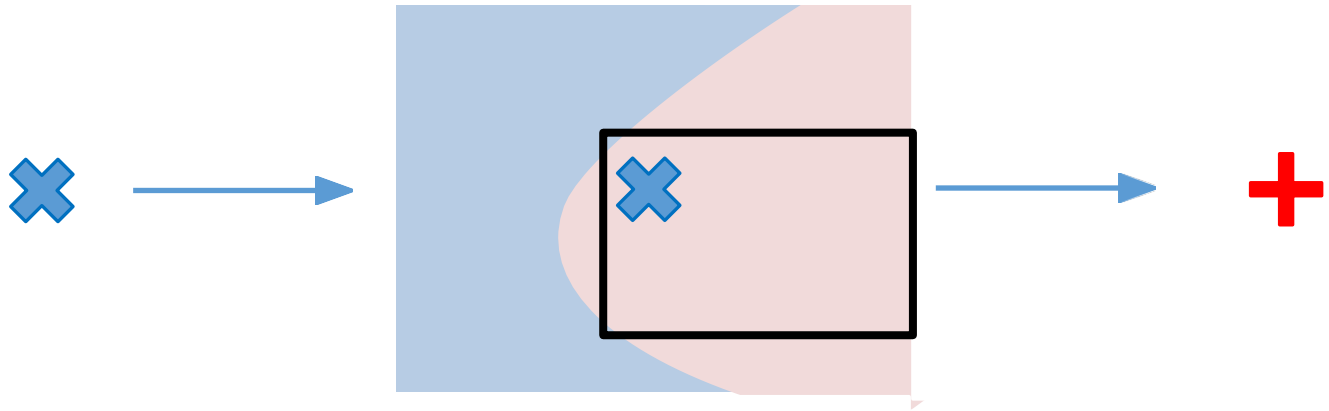
- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

[Ribeiro et al. 2018]

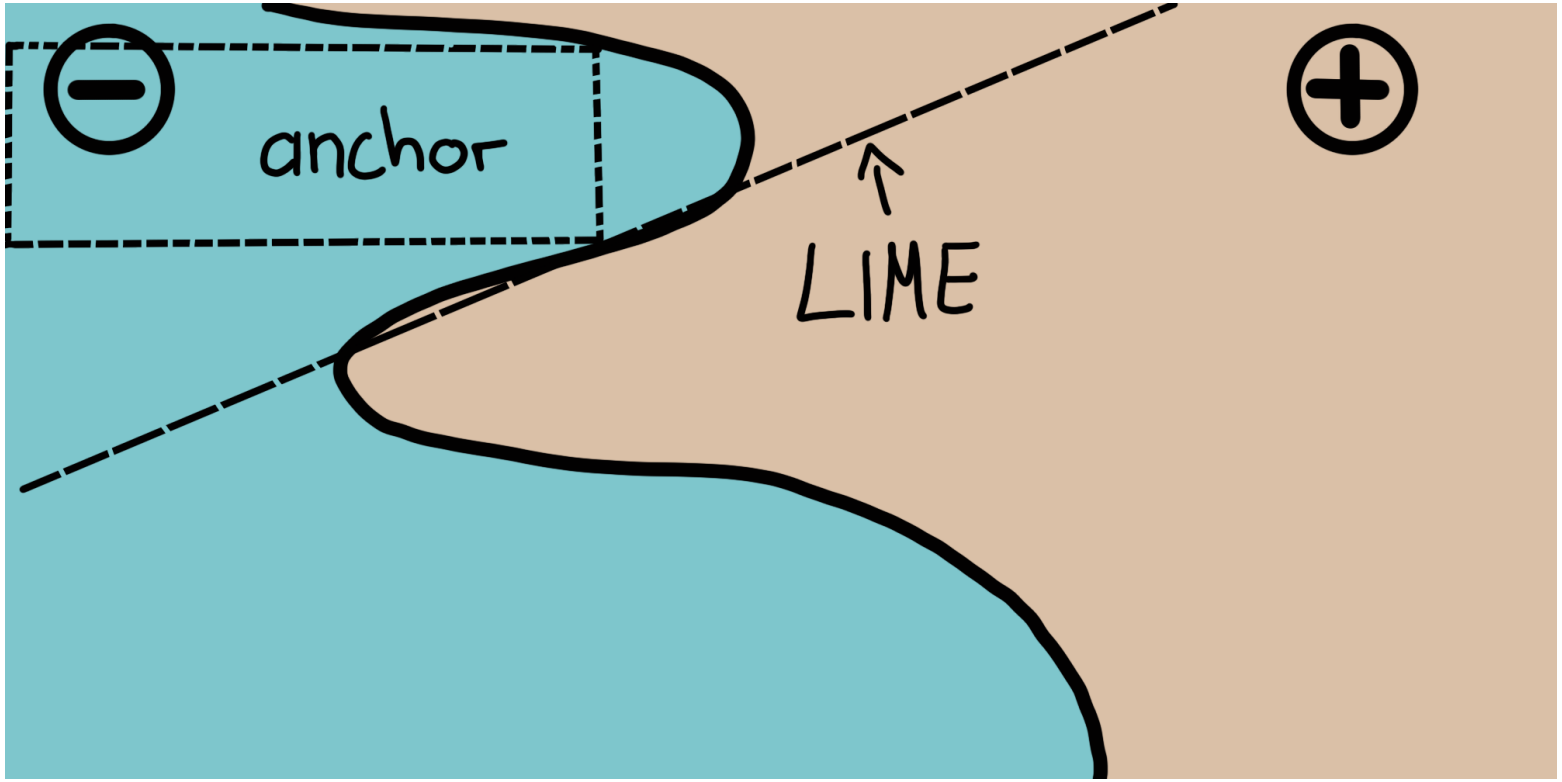
Anchors: Sufficient Conditions



Identify the conditions under which the classifier has the same prediction

[Ribeiro et al. 2018]

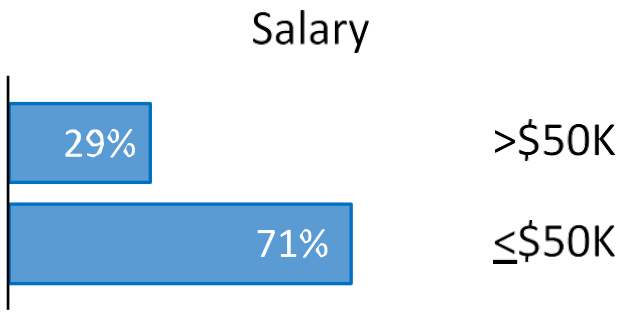
Anchors vs LIME



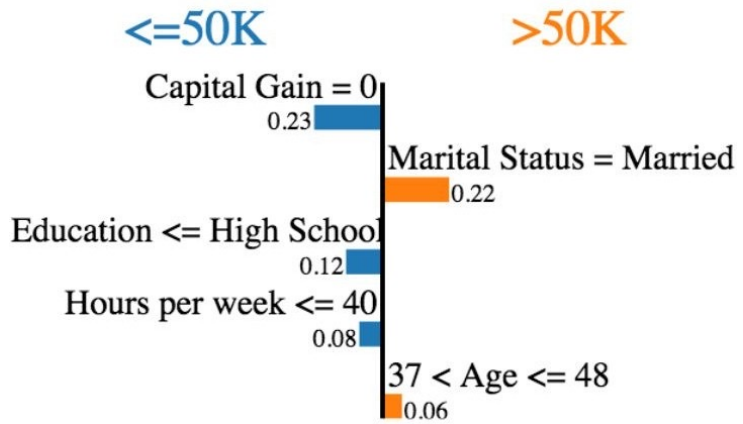
[Ribeiro et al. 2018]

Salary Prediction

Feature	Value
Age	$37 < \text{Age} \leq 48$
Workclass	Private
Education	$\leq$ High School
Marital Status	Married
Occupation	Craft-repair
Relationship	Husband
Race	Black
Sex	Male
Capital Gain	0
Capital Loss	0
Hours per week	≤ 40
Country	United States



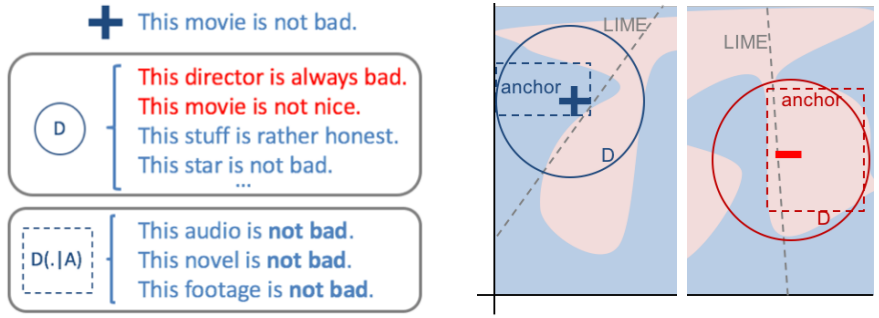
LIME



Anchors

IF Education ≤ High School
Then Predict Salary ≤ 50K

Anchors: more formally



(a) $\mathcal{D}$ and $\mathcal{D}(.|A)$ (b) Two toy visualizations

Let $A : \mathcal{X} \rightarrow \{0, 1\}$ be a rule (a set of predicates based on the inputs) such that $A(x) = 1$ and let $\mathcal{D}_x(.|A)$ be a distribution of neighbors of x matching A. A is an anchor if:

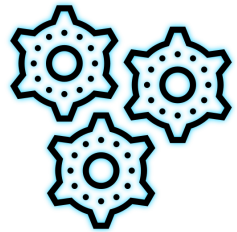
$$E_{\mathcal{D}_x(z|A)}\{1(\hat{f}(x) = \hat{f}(z))\} \geq \tau, A(x) = 1$$

where $0 \leq \tau \leq 1$ is a user-specified precision threshold.

A sophisticated search algorithm is used to find *A* (based on multi-armed bandit algorithms).

Easier to interpret than LIME but requires also the configuration of many hyperparameters.

Approaches for Post hoc Explainability



Local Explanations

- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

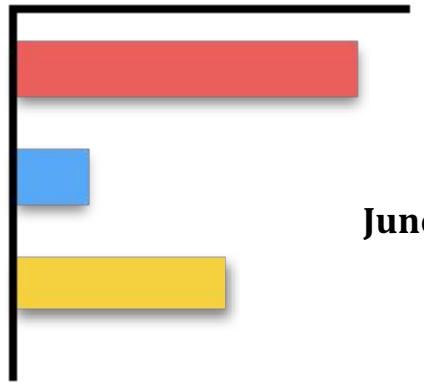
Saliency Map Overview

Input

Model

F_θ

Predictions



Junco Bird

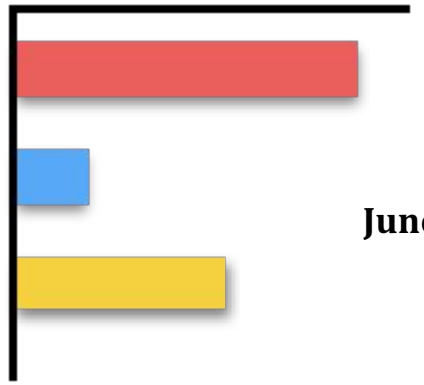
Saliency Map Overview

Input

Model

F_θ

Predictions



Junco Bird

What parts of the input are most relevant for the model's prediction: 'Junco Bird'?

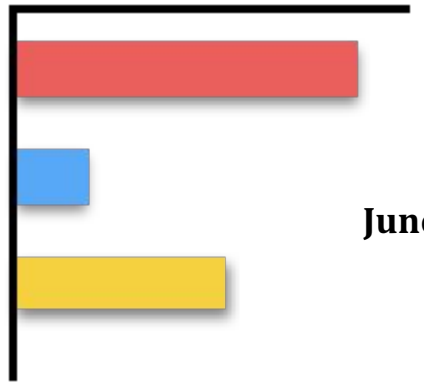
Saliency Map Overview

Input

Model

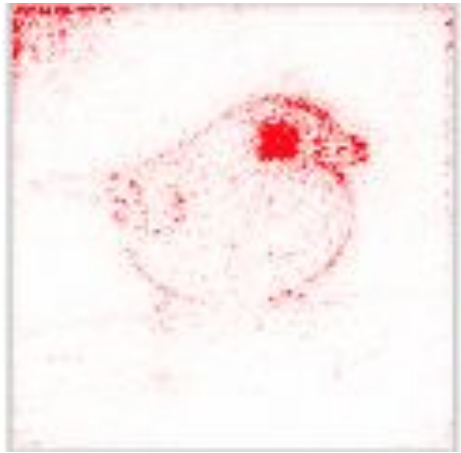
F_θ

Predictions

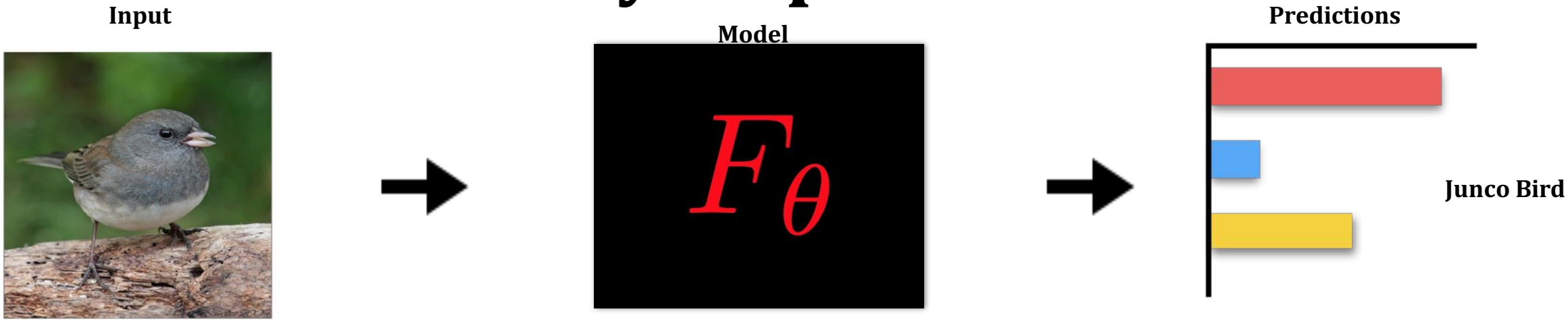


Junco Bird

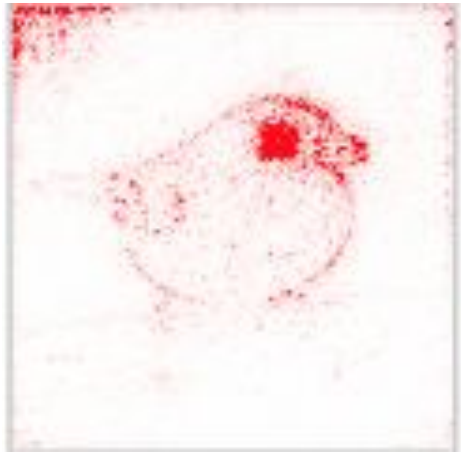
What parts of the input are most relevant for the model's prediction: 'Junco Bird'?



Saliency Map Overview



What parts of the input are most relevant for the model's prediction: 'Junco Bird'?



- **Feature Attribution**
- **'Saliency Map'**
- **Heatmap**

A Linear Model Detour

$$y = w^\top x \qquad x \in \mathbb{R}^d$$

$$y = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d$$

A Linear Model Detour: Sensitivity

$$y = w^\top x \qquad x \in \mathbb{R}^d$$

$$y = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d$$

How much does a unit change in an input dimension induce in the output?

A Linear Model Detour: Sensitivity

$$y = w^\top x \qquad x \in \mathbb{R}^d$$

$$y = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d$$

How much does a unit change in an input dimension induce in the output?

$$\nabla_x y = w$$

$$\downarrow$$

$$\text{Sensitivity} \equiv (w_1, w_2, \ldots, w_d)$$

A Linear Model Detour: Attribution

$$y = w^\top x \qquad x \in \mathbb{R}^d$$

$$y = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d$$

how can we apportion the output across all the input dimensions?

Another notion of relevance

$$y = w^\top x \qquad x \in \mathbb{R}^d$$

$$y = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d$$

how can we apportion the output across all the input dimensions?

$$(w_1 x_1, w_2 x_2, \ldots, w_d x_d)$$

Note: SHAP would output exactly that in the case of a linear model

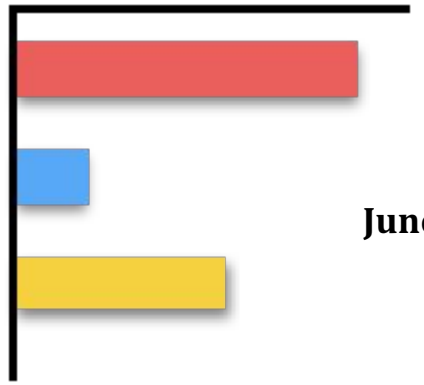
Modern DNN Setting

Input

Model

F_θ

Predictions



Junco Bird

$$F : \mathbb{R}^d \rightarrow \mathbb{R}^c$$ **Model**

$$F_i : \mathbb{R}^d \rightarrow \mathbb{R}$$ **class specific logit**

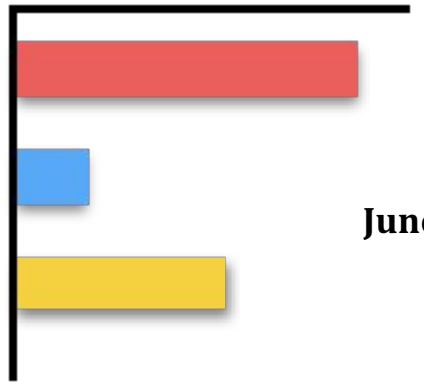
Input-Gradient

Input

Model

F_θ

Predictions



Junco Bird

Input-Gradient

$$\nabla_x F_i(x) \rightarrow \in \mathbb{R}^d$$

Logit

Input

Same dimension as the input.

Baehrens et. al. 2010; Simonyan et. al. 2014 .

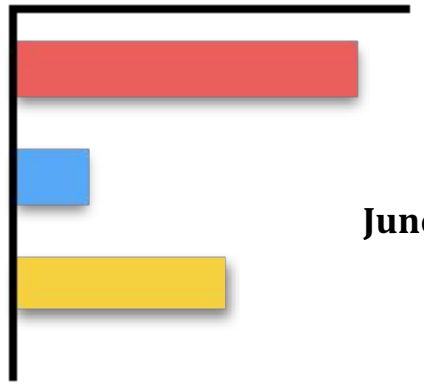
Input-Gradient

Input

Model

F_θ

Predictions



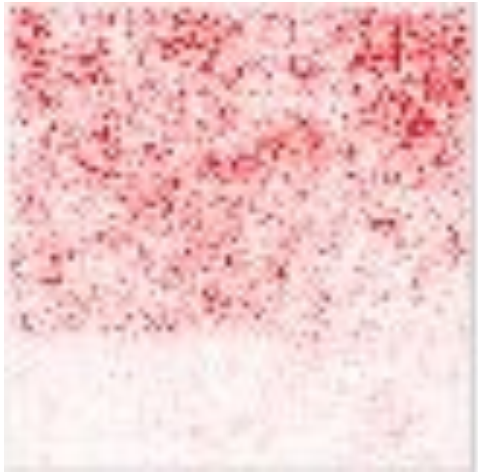
Junco Bird

Input-Gradient

$$\nabla_x F_i(x)$$

Input

Logit



Visualize as a heatmap

[Baehrens et. al. 2010](); [Simonyan et. al. 2014]().

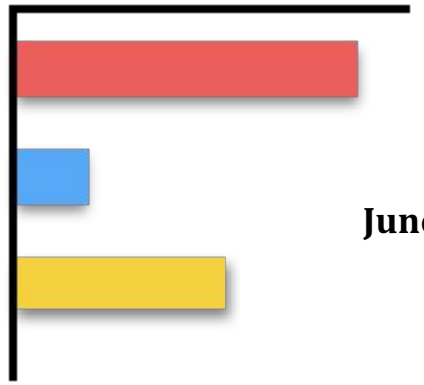
Input-Gradient

Input

Model

F_θ

Predictions



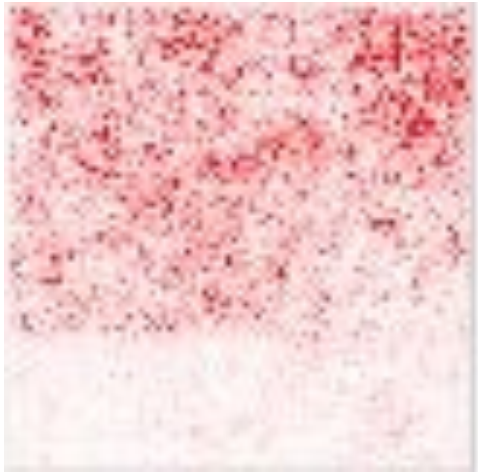
Junco Bird

Input-Gradient

$$\nabla_x F_i(x)$$

Logit

Input



Challenges

- Visually noisy & difficult to interpret.
- 'Gradient saturation.'

Shrikumar et. al. 2017.

Baehrens et. al. 2010; Simonyan et. al. 2014 .

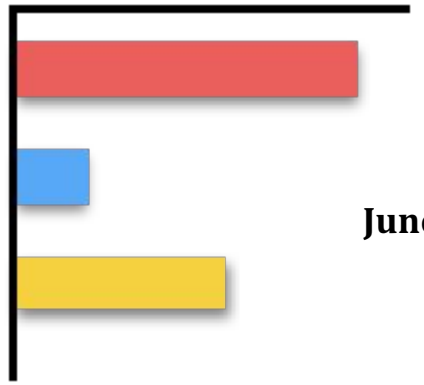
SmoothGrad

Input

Model

F_θ

Predictions



Junco Bird

SmoothGrad

$$\frac{1}{N}\sum_{i}^{N}\nabla_{(x+\epsilon)}F_i(x+\epsilon)$$

Gaussian noise

Average Input-gradient of ‘noisy’ inputs.

[Smilkov et. al. 2017]

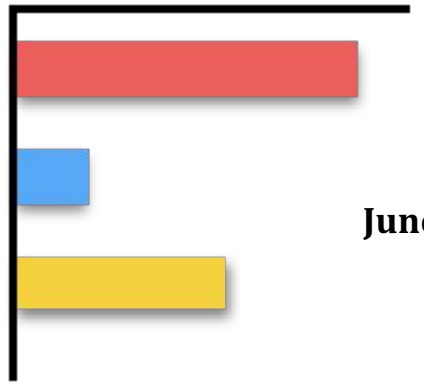
SmoothGrad

Input

Model

F_θ

Predictions



Junco Bird

SmoothGrad

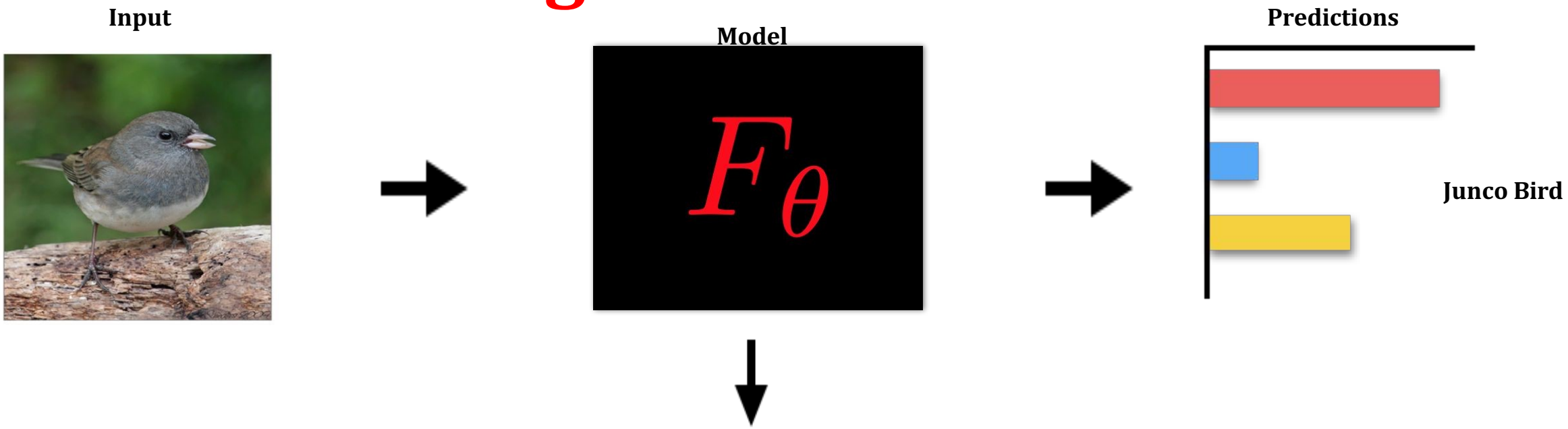
$$\frac{1}{N}\sum_{i}^{N}\nabla_{(x+\epsilon)}F_i(x+\epsilon)$$

Gaussian noise

Average Input-gradient of 'noisy' inputs.

Smilkov et. al. 2017

Integrated Gradients



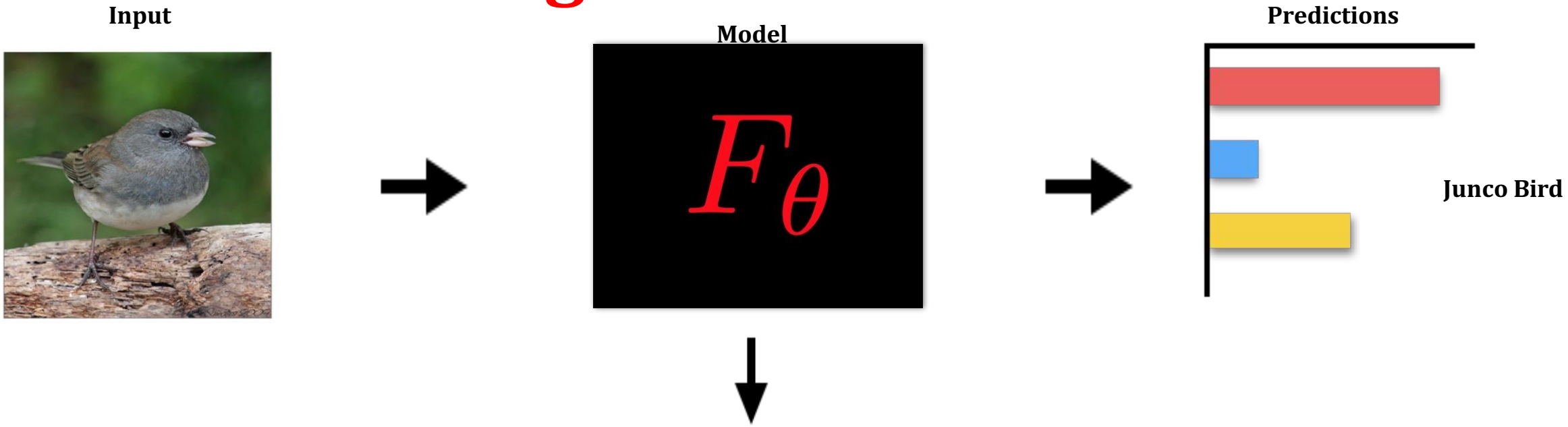
$$(x - \tilde{x}) \times \int_{\alpha=0}^{1} \frac{\partial F(\tilde{x} + \alpha \times (x - \tilde{x}))}{\partial x}$$

Baseline input

Path integral: 'sum' of interpolated gradients

[Sundararajan et. al. 2017]

Integrated Gradients



$$(x - \tilde{x}) \times \int_{\alpha=0}^{1} \frac{\partial F(\tilde{x} + \alpha \times (x - \tilde{x}))}{\partial x}$$

Baseline input

Note: Like SHAP, Integrated gradients are motivated from several desirable axioms that they are the only ones to satisfy
(Sundararajan et al., 2017)

Sundararajan et. al. 2017

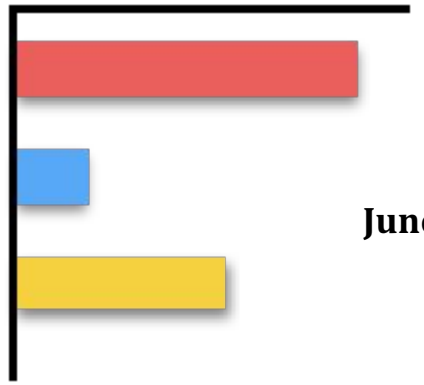
Gradient-Input

Input

Model

F_θ

Predictions



Junco Bird

Gradient-Input

$$\nabla_x F(x) \odot x$$

Input gradient

Input

Element-wise product of input-gradient and input.

Shrikumar et. al. 2017, Ancona et. al. 2018.

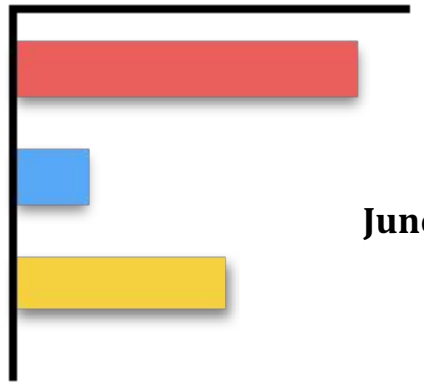
Gradient-Input

Input

Model

F_θ

Predictions



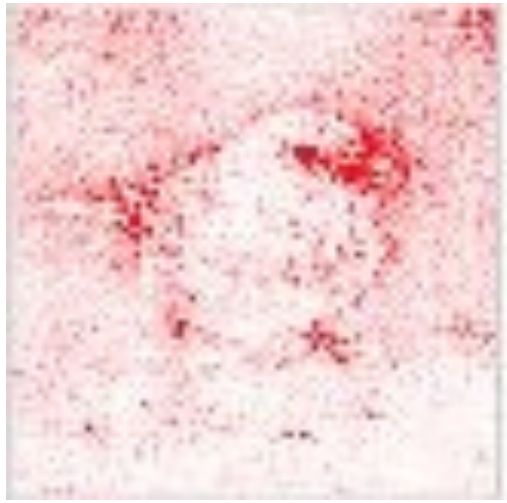
Junco Bird

Gradient-Input

$$\nabla_x F(x) \odot x$$

logit gradient

Input



Element-wise product of input-gradient and input.

Shrikumar et. al. 2017, Ancona et. al. 2018.

‘Modified Backprop’ Approaches

Compute feature relevance by modifying the backpropagation.

Note:

- These feature importance measures are not model-agnostic anymore.
- Previous methods are kind of model-agnostic but they require that gradients can be computed.

‘Modified Backprop’ Approaches

Compute feature relevance by modifying the backpropagation.

activation: $$f_i^{l+1} = relu(f_i^l) = \max(f_i^l, 0)$$

backpropagation: $$R_i^l = (f_i^l > 0) \cdot R_i^{l+1}, \text{ where } R_i^{l+1} = \frac{\partial f^{out}}{\partial f_i^{l+1}}$$

Source: [Springenberg & Dosovitskiy et. al. 2015]

‘Modified Backprop’ Approaches

Compute feature relevance by modifying the backpropagation.

activation: $$f_i^{l+1} = relu(f_i^l) = \max(f_i^l, 0)$$

backpropagation: $$R_i^l = (f_i^l > 0) \cdot R_i^{l+1}, \text{ where } R_i^{l+1} = \frac{\partial f^{out}}{\partial f_i^{l+1}}$$

guided backpropagation: $$R_i^l = (f_i^l > 0) \cdot (R_i^{l+1} > 0) \cdot R_i^{l+1}$$

Source: [Springenberg & Dosovitskiy et. al. 2015]

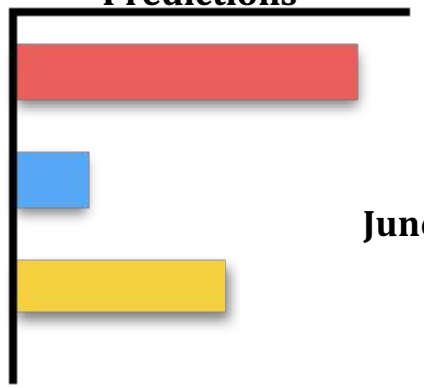
Attribution: Guided BackProp

Input

Model

F_θ

Predictions



Junco Bird

Guided BackProp

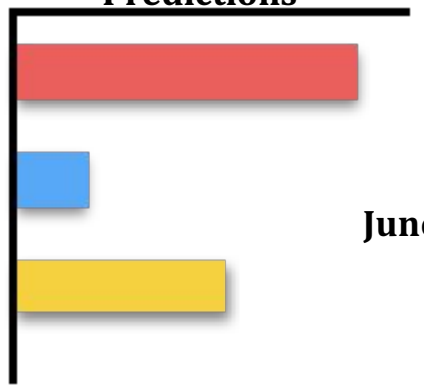
Attribution: Guided BackProp

Input

Model

F_θ

Predictions



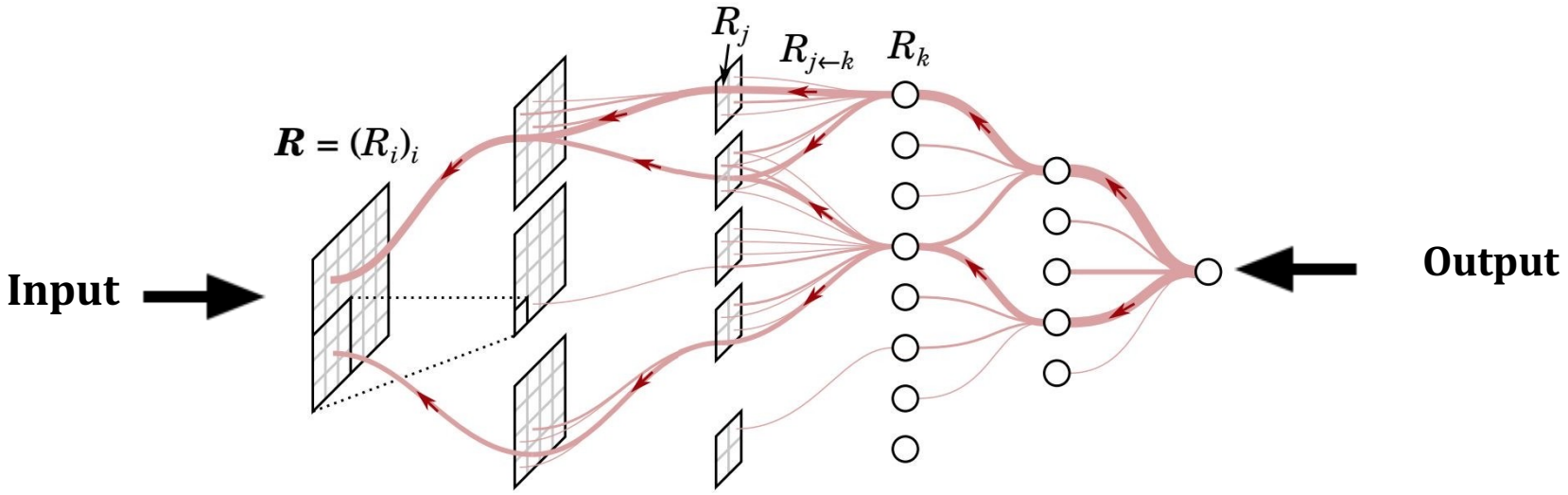
Junco Bird

Guided BackProp

Input Gradient

Layer Relevance Propagation (LRP)

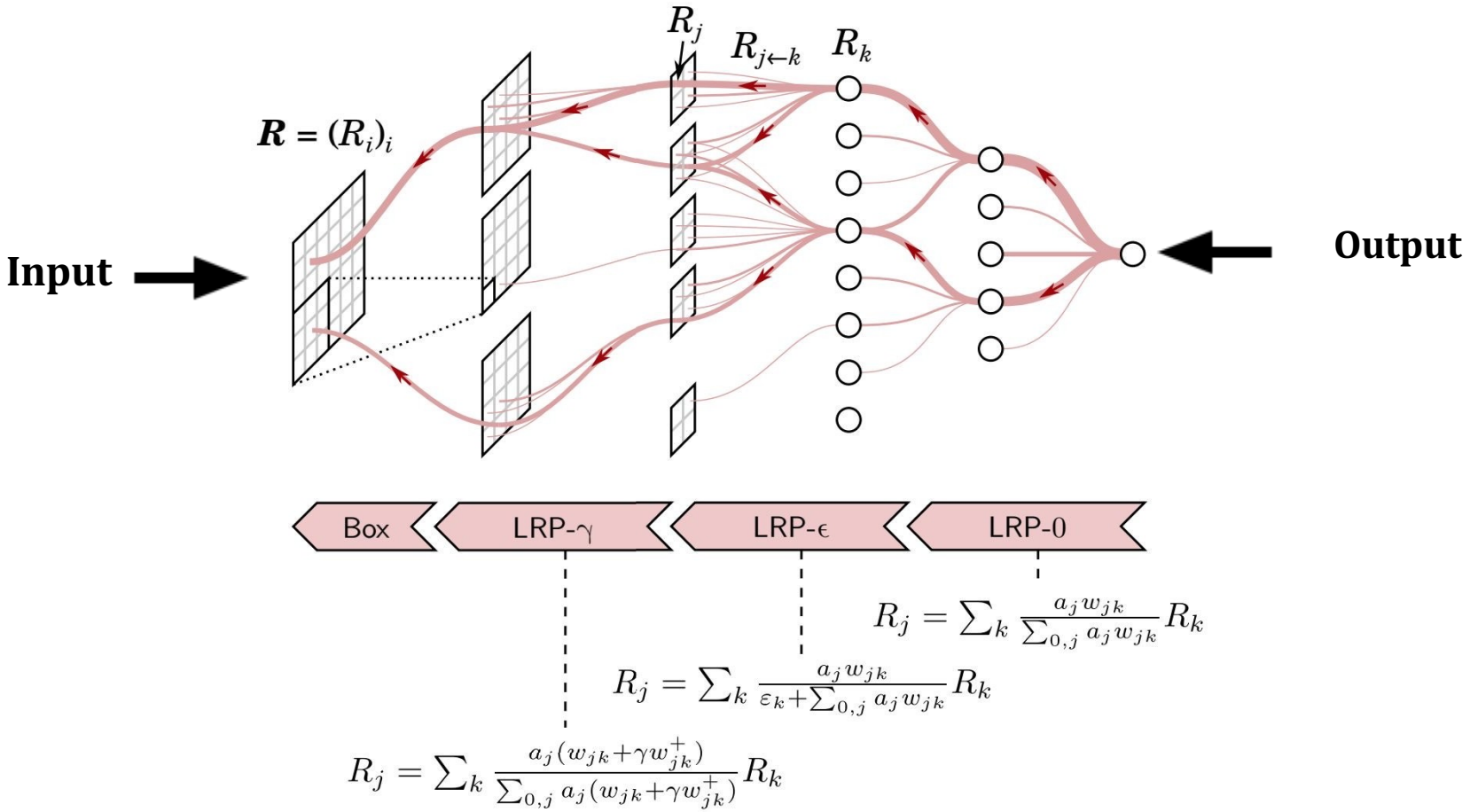
Compute feature relevance iteratively and propagate. Different **propagation rules** can be specified.



Source: heatmapping.org; ECML-PKDD 2020 Tutorial.

Layer Relevance Propagation (LRP)

Compute feature relevance iteratively and propagate. Different **propagation rules** can be specified.



Source: heatmapping.org; ECML-PKDD 2020 Tutorial.

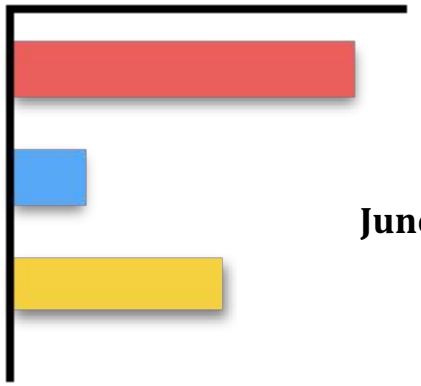
Layer Relevance Propagation (LRP)

Input

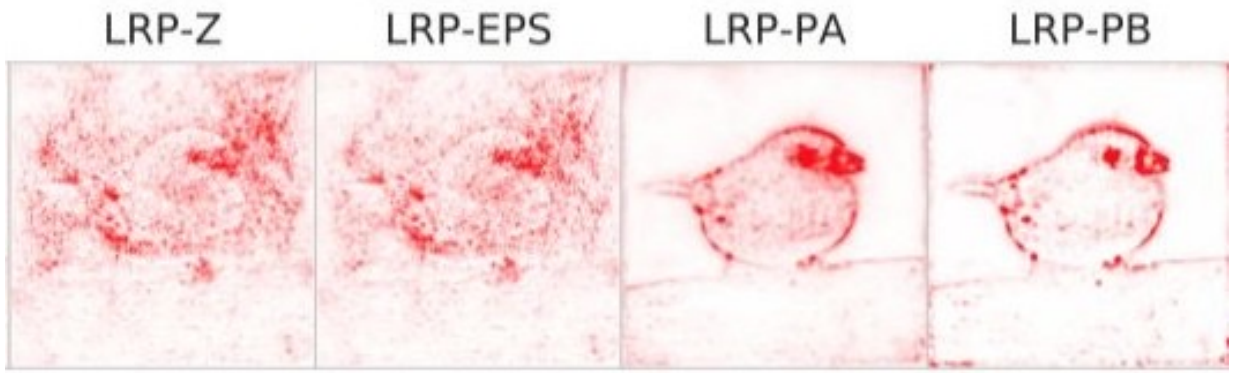
Model

F_θ

Predictions



Junco Bird



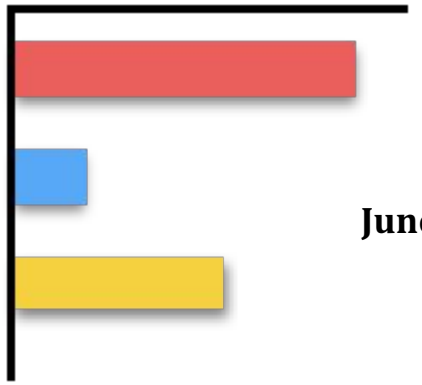
Recap

Input

Model

F_θ

Predictions



Junco Bird

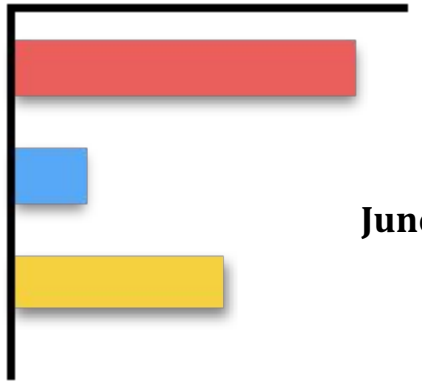
Recap

Input

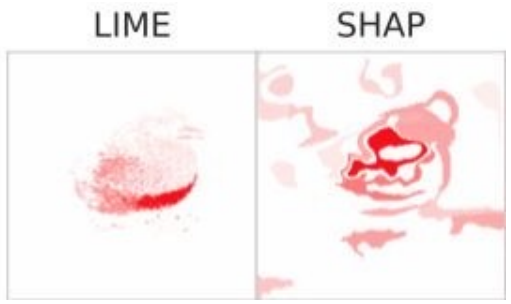
Model

F_θ

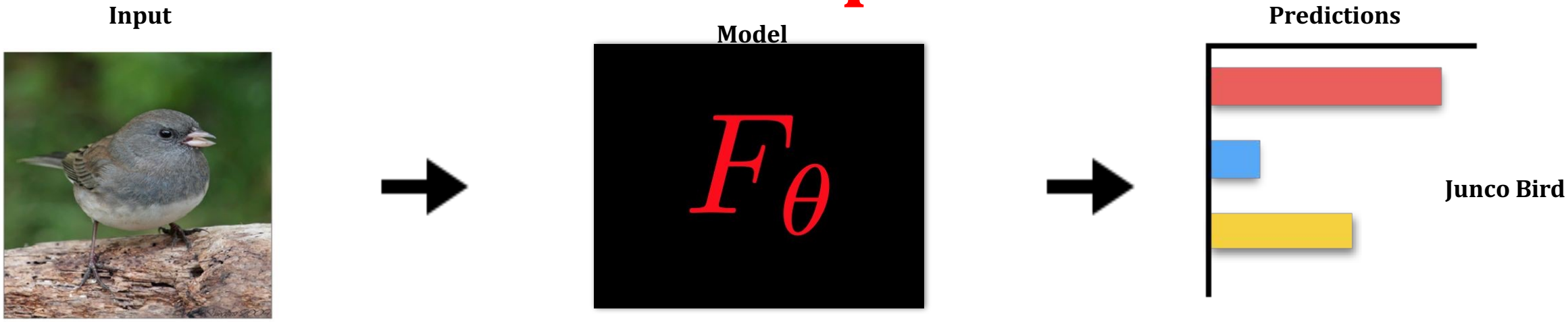
Predictions

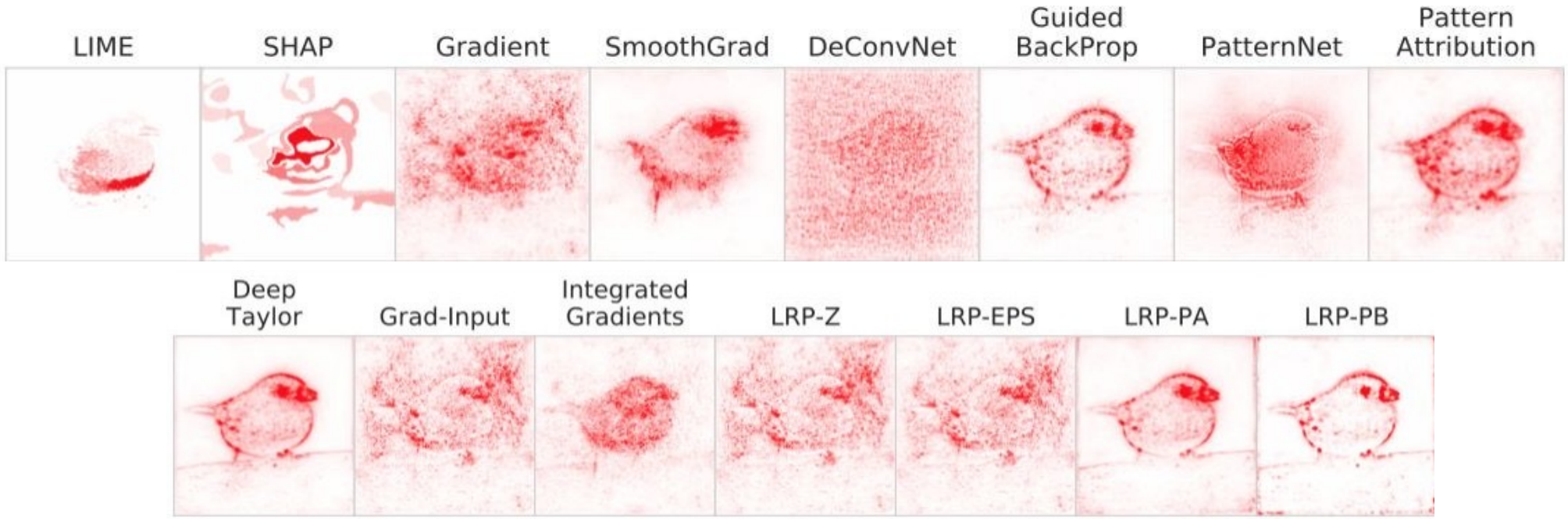


Junco Bird



Recap


Input
Model
F_θ
Predictions
Junco Bird


LIME
SHAP
Gradient
SmoothGrad
DeConvNet
Guided BackProp
PatternNet
Pattern Attribution
Deep Taylor
Grad-Input
Integrated Gradients
LRP-Z
LRP-EPS
LRP-PA
LRP-PB

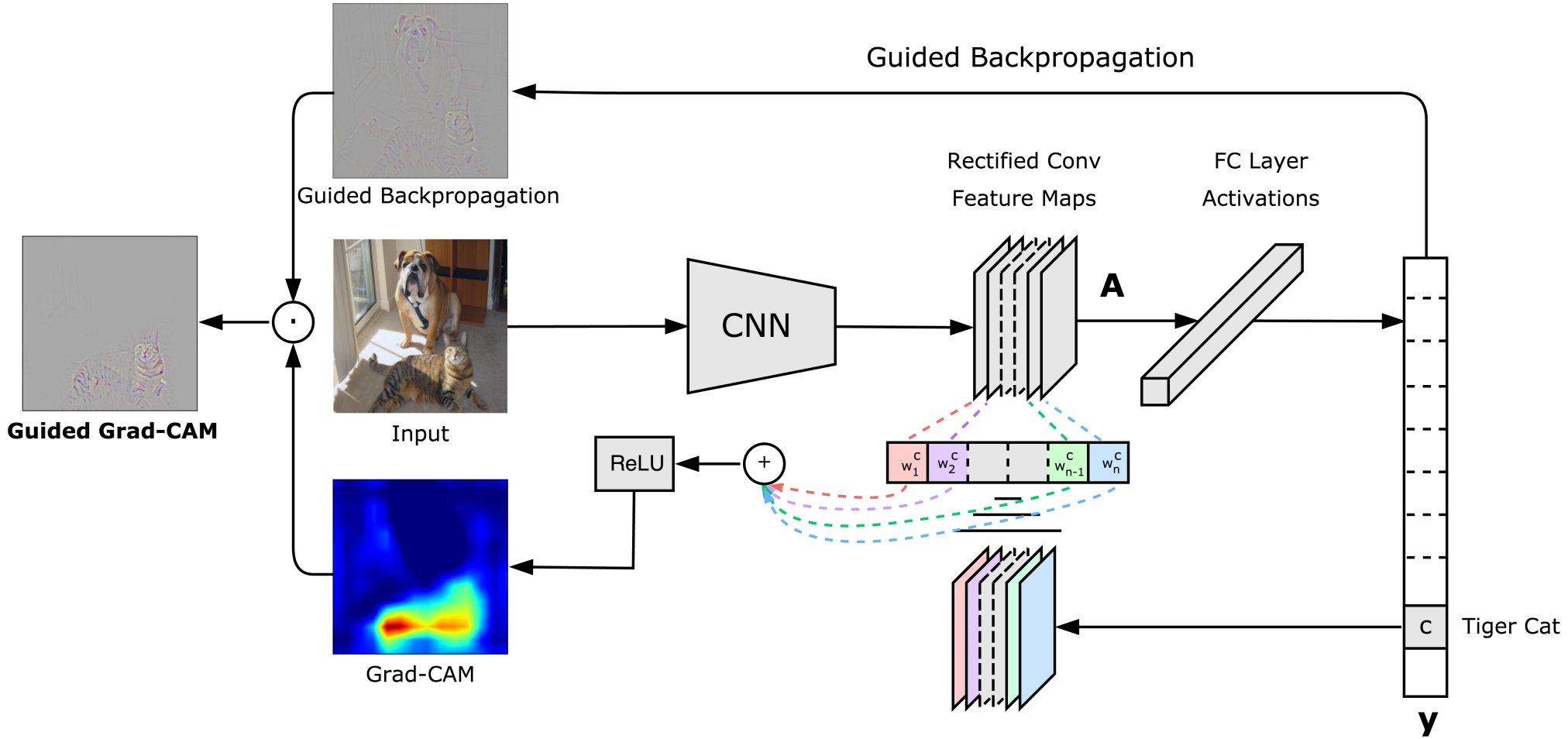
Additional Methods

- **Class Activation Mapping** (Zhou et. al. 2016).
- **Meaningful Perturbation** (Fong et. al. 2017).
- **RISE** (Petsuik et. al. 2018).
- **Extremal Perturbations** (Fong & Patrick 2019).
- **DeepLift** (Shrikumar et. al. 2018).
- **Expected Gradients** (Erion et. al. 2019)
- **Excitation Backprop** (Zhang et. al. 2016)
- **GradCAM** (Selvaraju et. al. 2016)
- **Guided GradCAM** (Selvaraju et. al. 2016)
- **Occlusion** (Zeiler et. al. 2014).
- **Prediction Difference Analysis** (Gu. et. al. 2019).
- **Internal Influence** (Leino et. al. 2018).

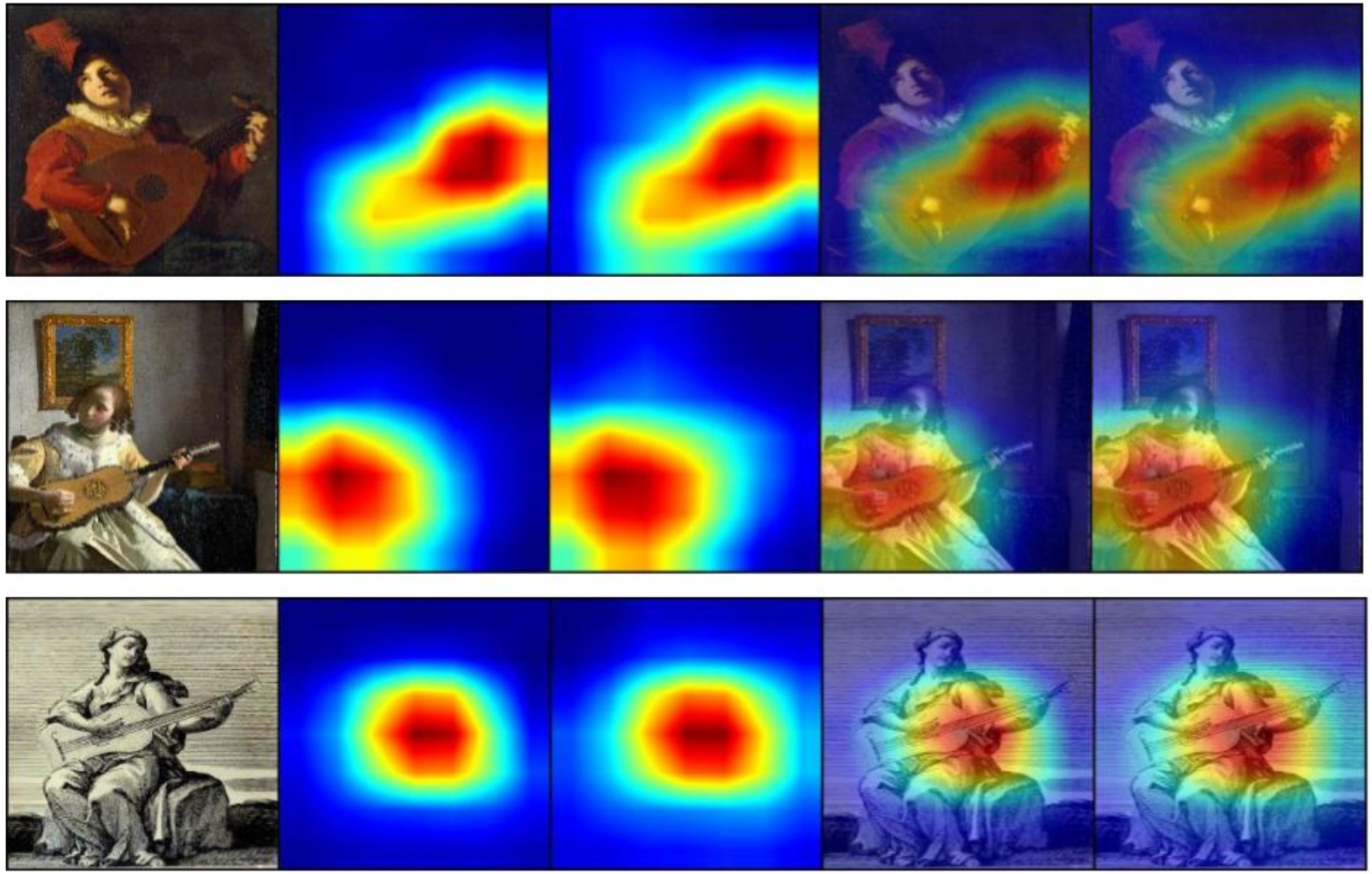
Note: not clear which one is the best to use.

See for additional methods: Samek & Montavon et. al. 2020

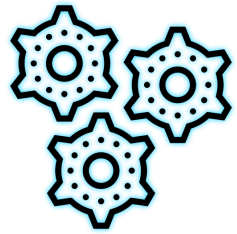
Grad-CAM (Selvaraju et. al. 2016)



Grad-CAM (Selvaraju et. al. 2016)



(Sabatelli et al., 2021)



Approaches for Post hoc Explainability

Local Explanations

- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

Prototype Approaches

Explain a model with synthetic or natural input **‘examples’**.

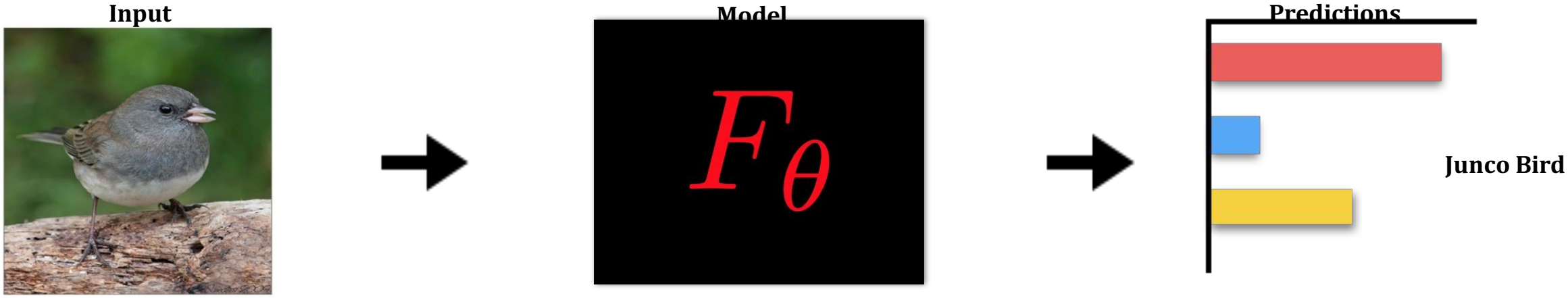
Prototype Approaches

Explain a model with synthetic or natural input **'examples'**.

Insights

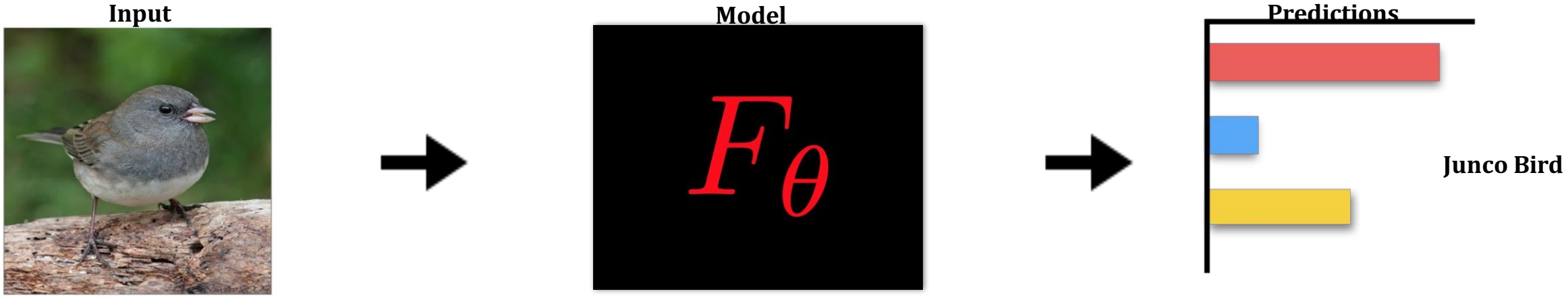
- What kind of input is the model **most likely to misclassify**?
- Which training samples are **mislabelled**?
- Which input **maximally activates** an intermediate neuron?

Training Point Ranking via Influence Functions



Which training data points have the most '*influence*' on the test loss?

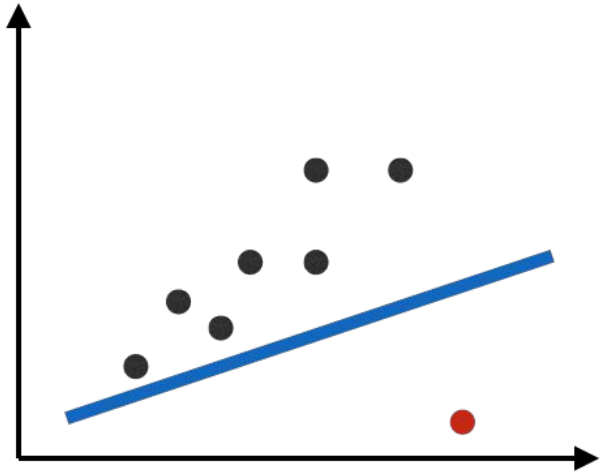
Training Point Ranking via Influence Functions

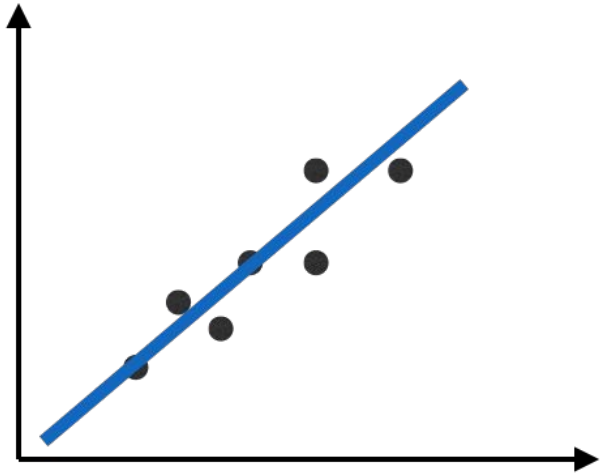


Which training data points have the most '*influence*' on the test loss?

Training Point Ranking via Influence Functions

Influence Function: classic tool used in robust statistics for assessing the effect of a sample on regression parameters (Cook & Weisberg, 1980).





Instead of refitting model for every data point, **Cook's distance** provides analytical alternative.

Training Point Ranking via Influence Functions

Koh & Liang (2017) extend the 'Cook's distance' insight to modern machine learning setting.

$z_i = (x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ $z_j = (x_j, y_j)$ ← **Training sample point** z_{test}

Koh & Liang 2017

Training Point Ranking via Influence Functions

Koh & Liang (2017) extend the 'Cook's distance' insight to modern machine learning setting.

$z_i = (x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ $z_j = (x_j, y_j)$ ← Training sample point z_{test}

ERM Solution

$$\hat{\theta} := \arg\min_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \ell(z_i; \theta)$$

UpWeighted ERM Solution

$$\hat{\theta}_{\epsilon, z_j} := \arg\min_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \ell(z_i; \theta) + \epsilon \ell(z_j; \theta) \quad \epsilon = -\frac{1}{n}$$

Koh & Liang 2017

Training Point Ranking via Influence Functions

Koh & Liang (2017) extend the 'Cook's distance' insight to modern machine learning setting.

$z_i = (x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ $\quad z_j = (x_j, y_j)$ ← Training sample point $\quad z_{\text{test}}$

ERM Solution

$$\hat{\theta} := \arg\min_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \ell(z_i; \theta)$$

UpWeighted ERM Solution

$$\hat{\theta}_{\epsilon, z_j} := \arg\min_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \ell(z_i; \theta) + \epsilon \ell(z_j; \theta) \quad \epsilon = -\frac{1}{n}$$

Influence of Training Point on Parameters

$$\mathcal{I}_{z_j} = \left. \frac{d\hat{\theta}_{\epsilon, z_j}}{d\epsilon} \right|_{\epsilon=0} = -H_{\hat{\theta}}^{-1} \nabla_\theta \ell(z_j, \hat{\theta})$$

Influence of Training Point on Test-Input's loss

$$\mathcal{I}_{z_j, z_{\text{test}}, \text{loss}} = -\nabla_\theta \ell(z_{\text{test}}, \hat{\theta})^\top H_\theta^{-1} \nabla_\theta \ell(z_j, \hat{\theta})$$

Koh & Liang 2017

Training Point Ranking via Influence Functions

Applications:

- compute self-influence to identify mislabelled examples;
- diagnose possible domain mismatch;
- craft training-time poisoning examples.

Alternatives:

- **Representer Points** (Yeh et. al. 2018).
- **TracIn** (Pruthi et. al. appearing at NeuRIPs 2020).

[Koh & Liang 2017]

‘Activation Maximization’

These approaches identify examples, synthetic or natural, that **strongly activate a function (neuron) of interest**.

‘Activation Maximization’

These approaches identify examples, synthetic or natural, that **strongly activate a function (neuron) of interest**.

Implementation Flavors:

- Search for **natural examples within a specified set** (training or validation corpus) that strongly activate a neuron of interest;

- **Synthesize examples**, typically via gradient descent, that strongly activate a neuron of interest.

Feature Visualization

Dataset Examples show us what neurons respond to in practice

Optimization isolates the causes of behavior from mere correlations. A neuron may not be detecting what you initially thought.

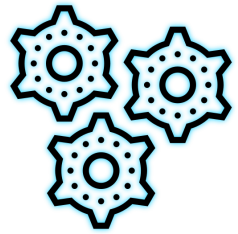
Baseball—or stripes?
mixed4a, Unit 6

Animal faces—or snouts?
mixed4a, Unit 240

Clouds—or fluffiness?
mixed4a, Unit 453

Buildings—or sky?
mixed4a, Unit 492

[Olah et. al. 2017]



Approaches for Post hoc Explainability

Local Explanations

- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

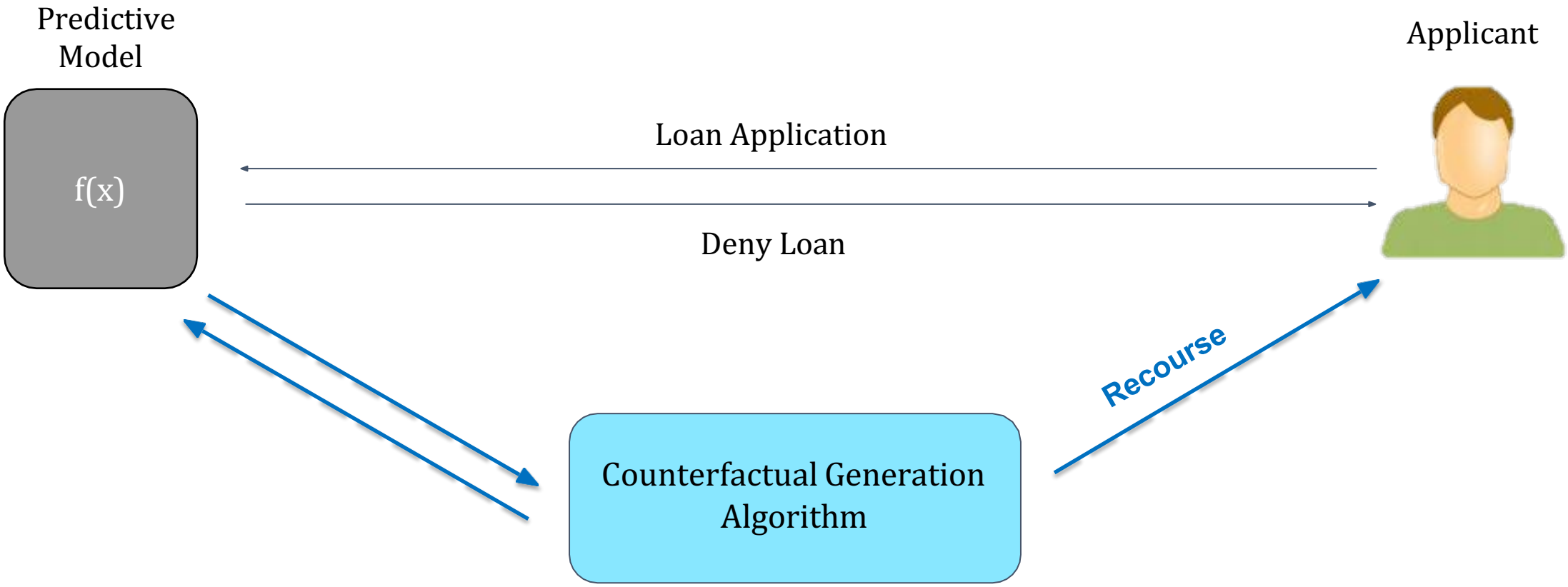
Counterfactual Explanations

As ML models increasingly deployed to make high-stakes decisions (e.g., loan applications), it becomes important to provide recourse to affected individuals.

Counterfactual Explanations

What features need to be changed and by how much to flip a model's prediction ?
(i.e., to reverse an unfavorable outcome).

Counterfactual Explanations



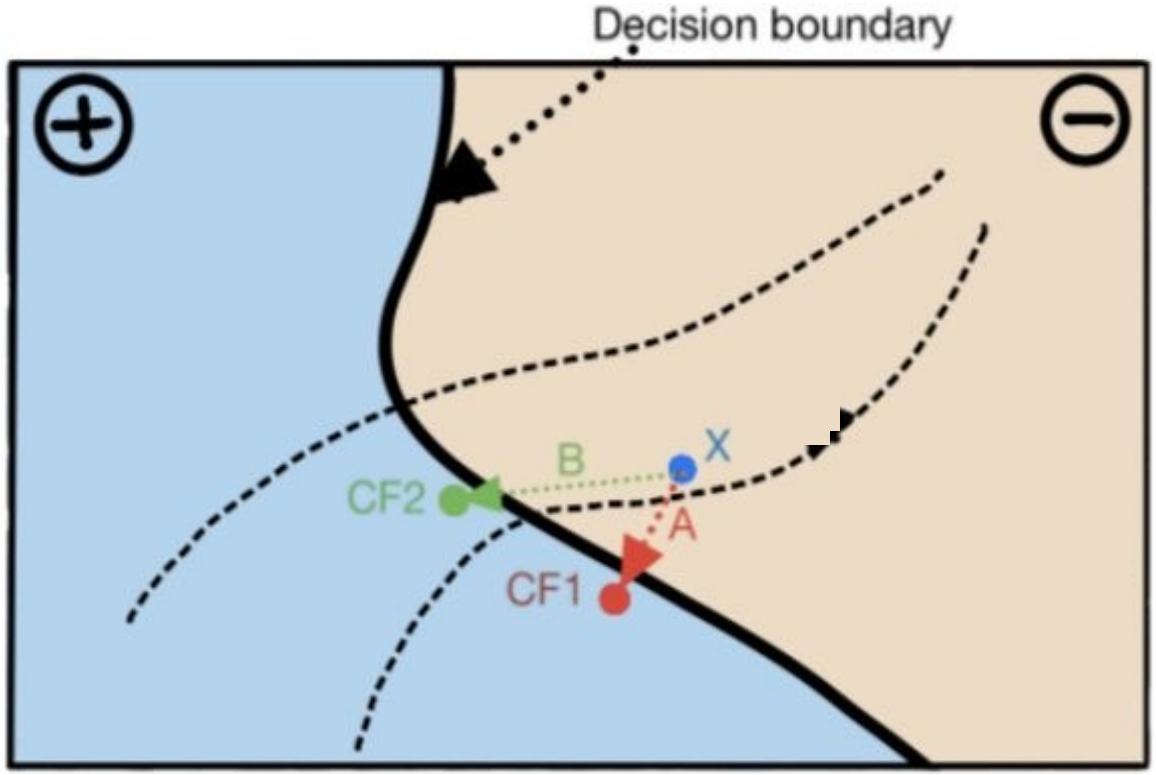
Recourse: Increase your salary by 50K & pay your credit card bills on time for next 3 months

Counterfactual Explanations

- Important to provide "recourse" to affected individuals (GDPR)

- Counterfactual Explanations:
  - *What features need to be changed and by how much to flip a model's prediction (i.e., to reverse an unfavorable outcome).*

Generating Counterfactual Explanations: Intuition



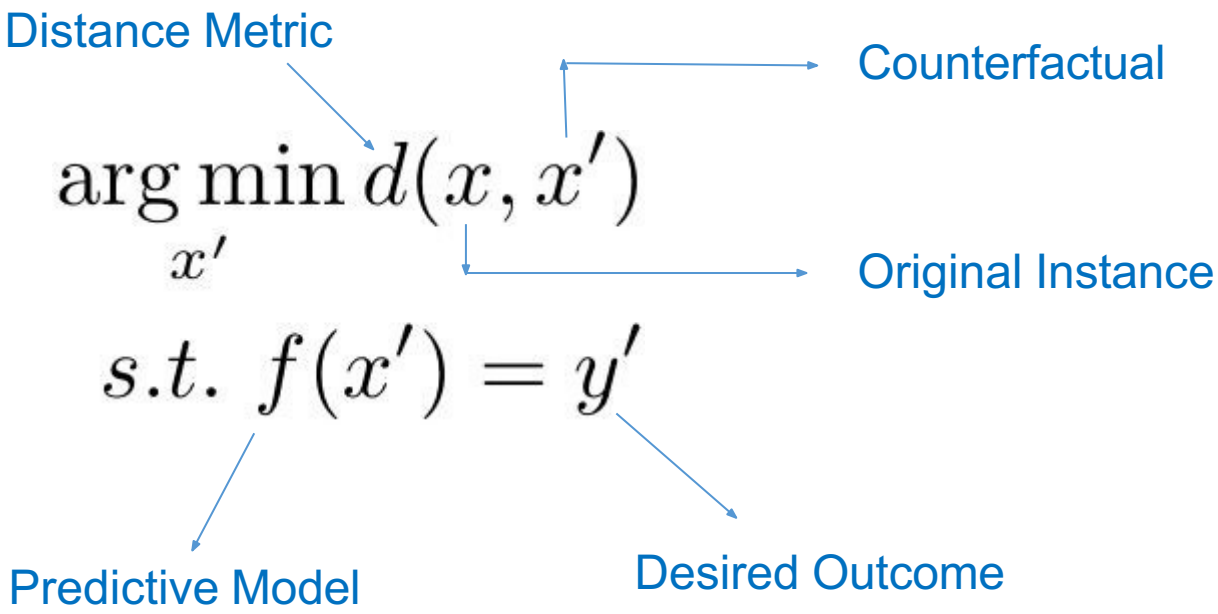
Proposed solutions differ on:

1. How to choose among candidate counterfactuals?

2. How much access is needed to the underlying predictive model?

[Wachter et. al., 2018]

Take 1: Minimum Distance Counterfactuals



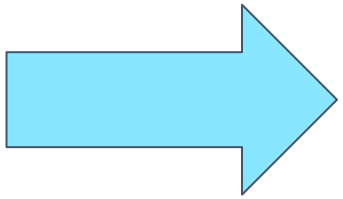
Choice of distance metric dictates what kinds of counterfactuals are chosen.

Wachter et. al. use normalized Manhattan distance.

[Wachter et. al., 2018]

Take 1: Minimum Distance Counterfactuals

$$\arg\min_{x'} d(x, x') \quad s.t.\ f(x') = y'$$



$$\arg\min_{x'} \lambda\ (f(x') - y')^2 + d(x, x')$$

Wachter et. al. solve a differentiable, unconstrained version of the objective using ADAM optimization algorithm with random restarts.

This method ***requires access to gradients*** of the underlying predictive model.

Take 1: Minimum Distance Counterfactuals

Person 1: If your LSAT was 34.0, you would have an average predicted score (0).

Person 2: If your LSAT was 32.4, you would have an average predicted score (0).

Person 3: If your LSAT was 33.5, and you were 'white', you would have an average predicted score (0).

Person 4: If your LSAT was 35.8, and you were 'white', you would have an average predicted score (0).

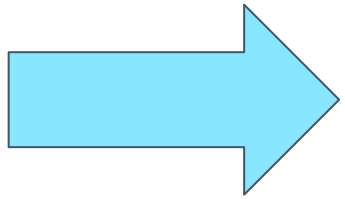
Person 5: If your LSAT was 34.9, you would have an average predicted score (0).

Not feasible to act upon these features!

[Ustun et. al., 2019]

Take 2: Feasible and Least Cost Counterfactuals

$$\arg\min_{x'} d(x, x') \quad s.t.\ f(x') = y'$$



$$\arg\min_{x' \in \mathcal{A}} cost(x, x') \quad s.t.\ f(x') = y'$$

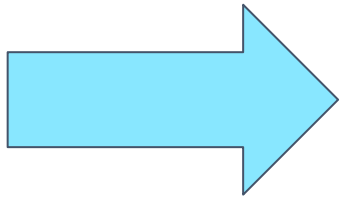
- $\mathcal{A}$ is the set of feasible counterfactuals (input by end user)
  - E.g., changes to race, gender are not feasible

- *Cost* is modeled as total log-percentile shift
  - Changes become harder when starting off from a higher percentile value

[Ustun et. al., 2019]

Take 2: Feasible and Least Cost Counterfactuals

$$\arg\min_{x'} d(x, x') \quad s.t.\ f(x') = y'$$



$$\arg\min_{x' \in \mathcal{A}} cost(x, x') \quad s.t.\ f(x') = y'$$

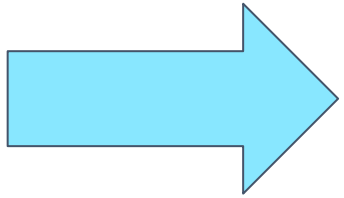
- Ustun et. al. only consider the case where the model is a linear classifier
  - Objective formulated as an IP and optimized using CPLEX

- Requires ***complete access*** to the linear classifier i.e., weight vector

[Ustun et. al., 2019]

Take 2: Feasible and Least Cost Counterfactuals

$$\arg\min_{x'} d(x, x') \quad s.t.\ f(x') = y'$$



$$\arg\min_{x' \in \mathcal{A}} cost(x, x') \quad s.t.\ f(x') = y'$$

Question: What if we have a black box or a non-linear classifier?

Answer: generate a local linear model appr
and then apply Ustun et. al.'s framework

Note: there exist solutions for additive tree models (Cui et al., 2015, SIGKDD)

[Ustun et. al., 2019]

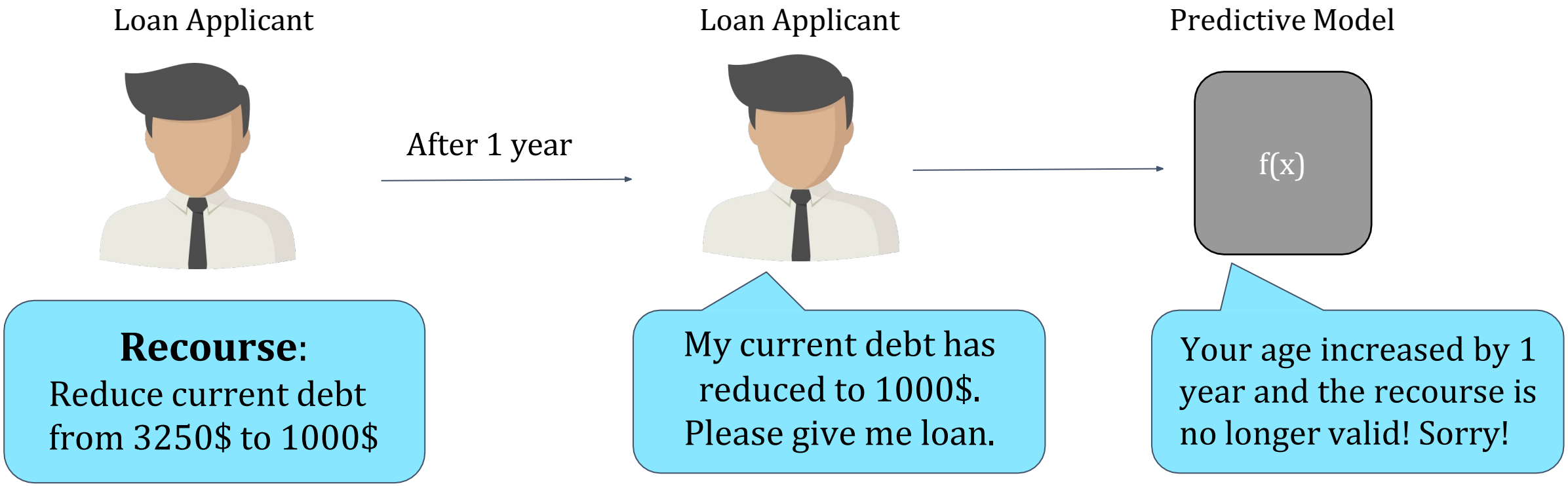
Take 2: Feasible and Least Cost Counterfactuals

Features to Change	Current Values		Required Values
n_credit_cards	5	⟶	3
current_debt	$3,250	⟶	$1,000
has_savings_account	FALSE	⟶	TRUE
has_retirement_account	FALSE	⟶	TRUE

Changing one feature without affecting another might not be possible!

[Mahajan et. al., 2019, Karimi et. al. 2020]

Take 3: Causally Feasible Counterfactuals



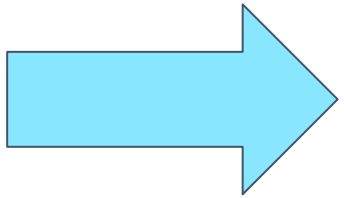
Important to account for *feature interactions* when generating counterfactuals!

But how?!

[Mahajan et. al., 2019]

Take 3: Causally Feasible Counterfactuals

$$\arg\min_{x'} d(x, x') \quad s.t.\ f(x') = y'$$



$$\arg\min_{x'} d\_causal(x, x') \quad s.t.\ f(x') = y'$$

Leverage Structural Causal Model (SCM) to define this new distance metric

[Mahajan et. al., 2019]

Take 3: Causally Feasible Counterfactuals

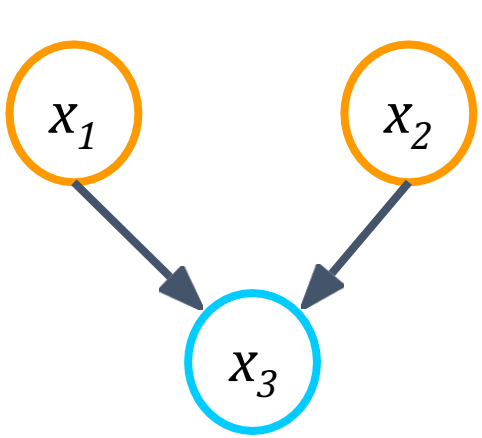
$$d\_causal(x, x') =$$

$$\sum_{u \in U} d(x_u, x'_u) +$$

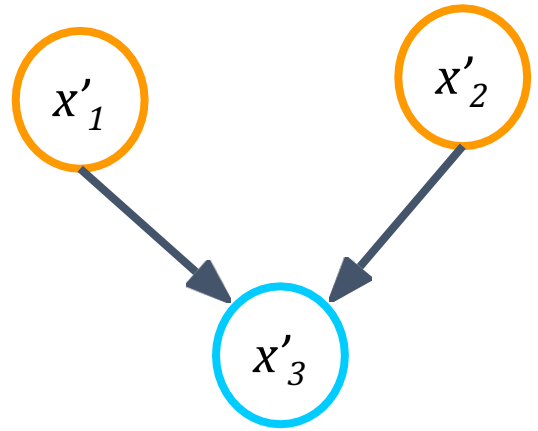
Standard L1/L2 distance for each variable *u* with no parents

$$\sum_{v \in V} d(x_v, \mathbb{E}[x'_v | x'_{v_{p1}}, x'_{v_{p2}}, \cdots x'_{v_{pM}}])$$

For variables *v* with parents, compute L1/L2 distance between value of *v* for original instance and *expected value of v* given its parents for counterfactual

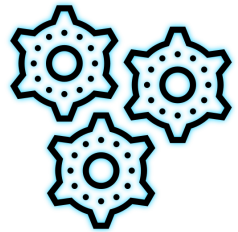


Original Instance



Counterfactual

U is set of nodes without parents in the graph;
V is set of nodes with parents in the graph



Approaches for Post hoc Explainability

Local Explanations

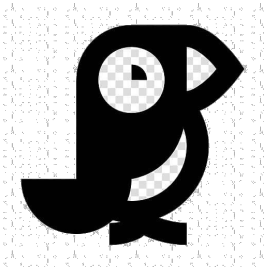
- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

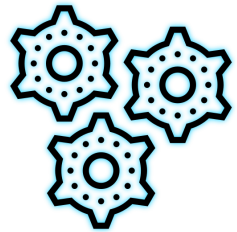
Global Explanations

- Explain the complete behavior of a given (black box) model
  - Provide a *bird's eye view* of model behavior



- Help detect *big picture* model biases persistent across larger subgroups of the population
  - Impractical to manually inspect local explanations of several instances to ascertain big picture biases!

- Global explanations are complementary to local explanations



Approaches for Post hoc Explainability

Local Explanations

- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

Global Explanation as a Collection of Local Explanations

How to generate a global explanation of a (black box) model?

- Generate a local explanation for every instance in the data using one of the approaches discussed earlier
- Pick a subset of k local explanations to constitute the global explanation

What local explanation technique to use?
How to choose the subset of k local explanations?

[Ribeiro et al. 2016]

Global Explanations from Local Feature Importances: SP-LIME

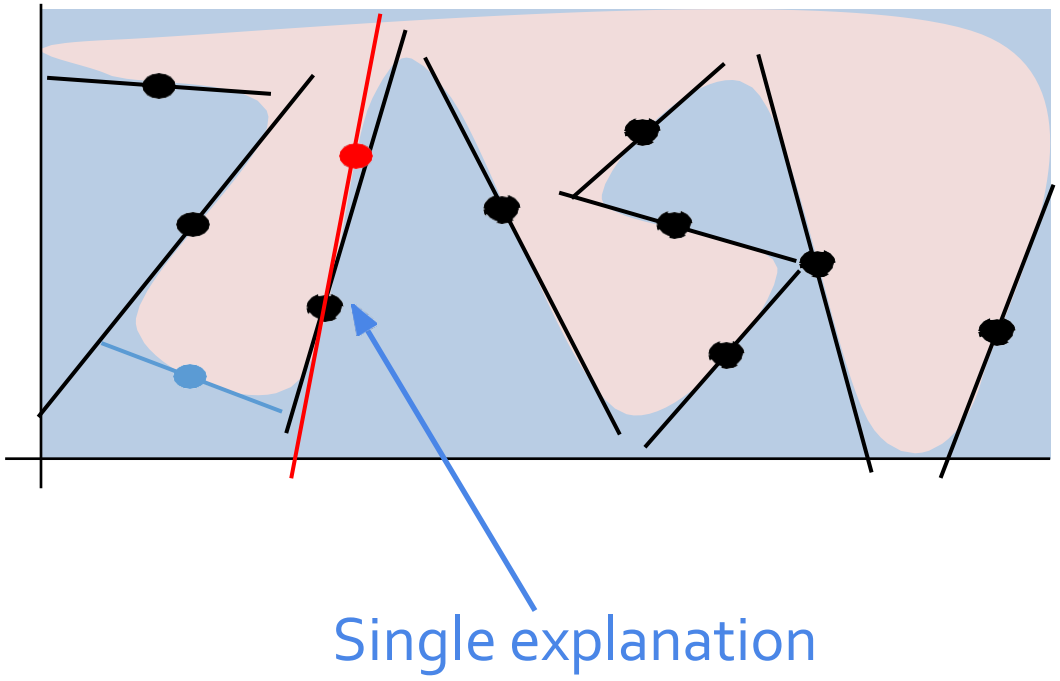
LIME explains a single prediction
local behavior for a single instance

Can't examine all explanations
Instead pick *k* explanations to show to the user

Representative
Should summarize the model's global behavior

Diverse
Should not be redundant in their descriptions

SP-LIME uses submodular optimization and *greedily* picks k explanations



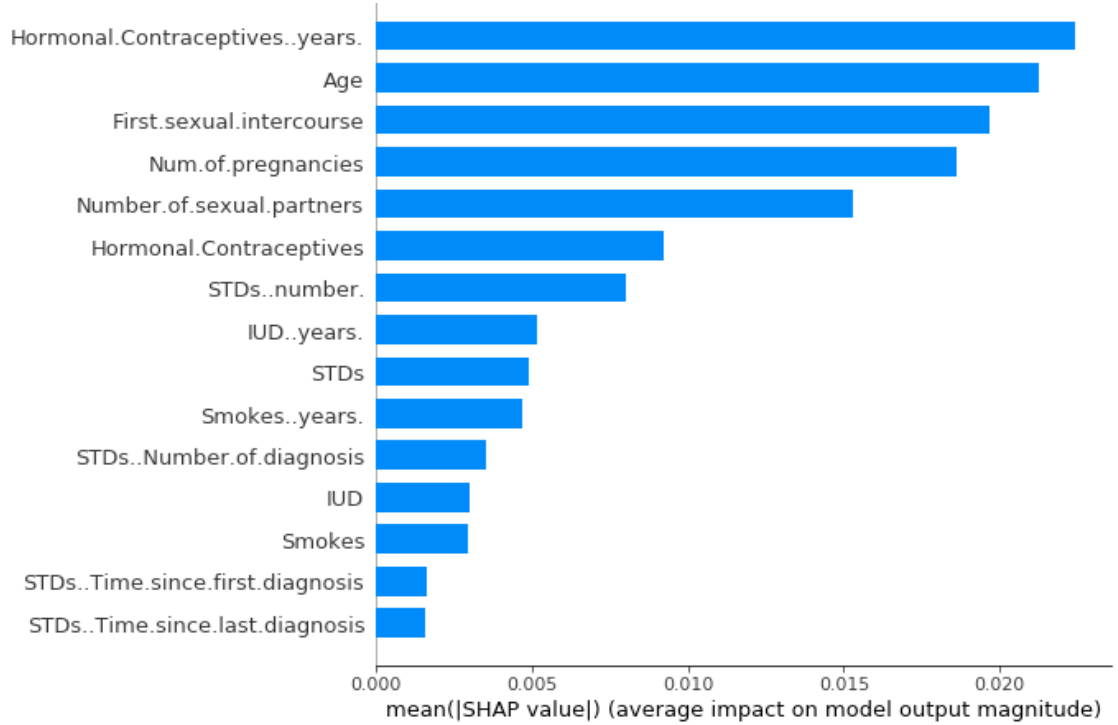
Model Agnostic

SHAP global feature importance

Two ways:

1. average absolute values of local feature importances over a sample:

$$I_j = \frac{1}{n} \sum_{i=1}^{n} |\phi_j^{(i)}|$$



https://christophm.github.io/interpretable-ml-book/

2. Use the loss over the training set as the characteristic function to decompose (SAGE, Covert et al., ICML 2020)

Permutation feature importance (Fisher et al., 2018)

A very simple model-agnostic global feature importance measure:

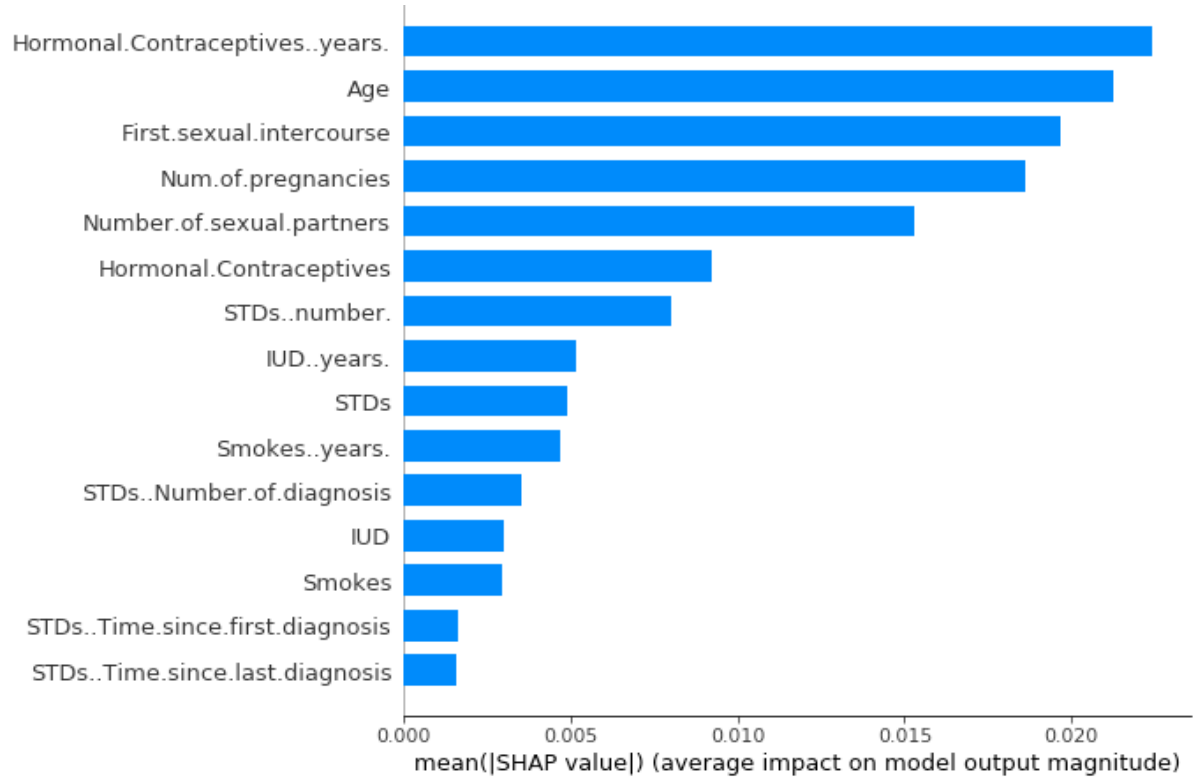
Given (test) data (X, y), a model $\hat{f}$ and a loss L:

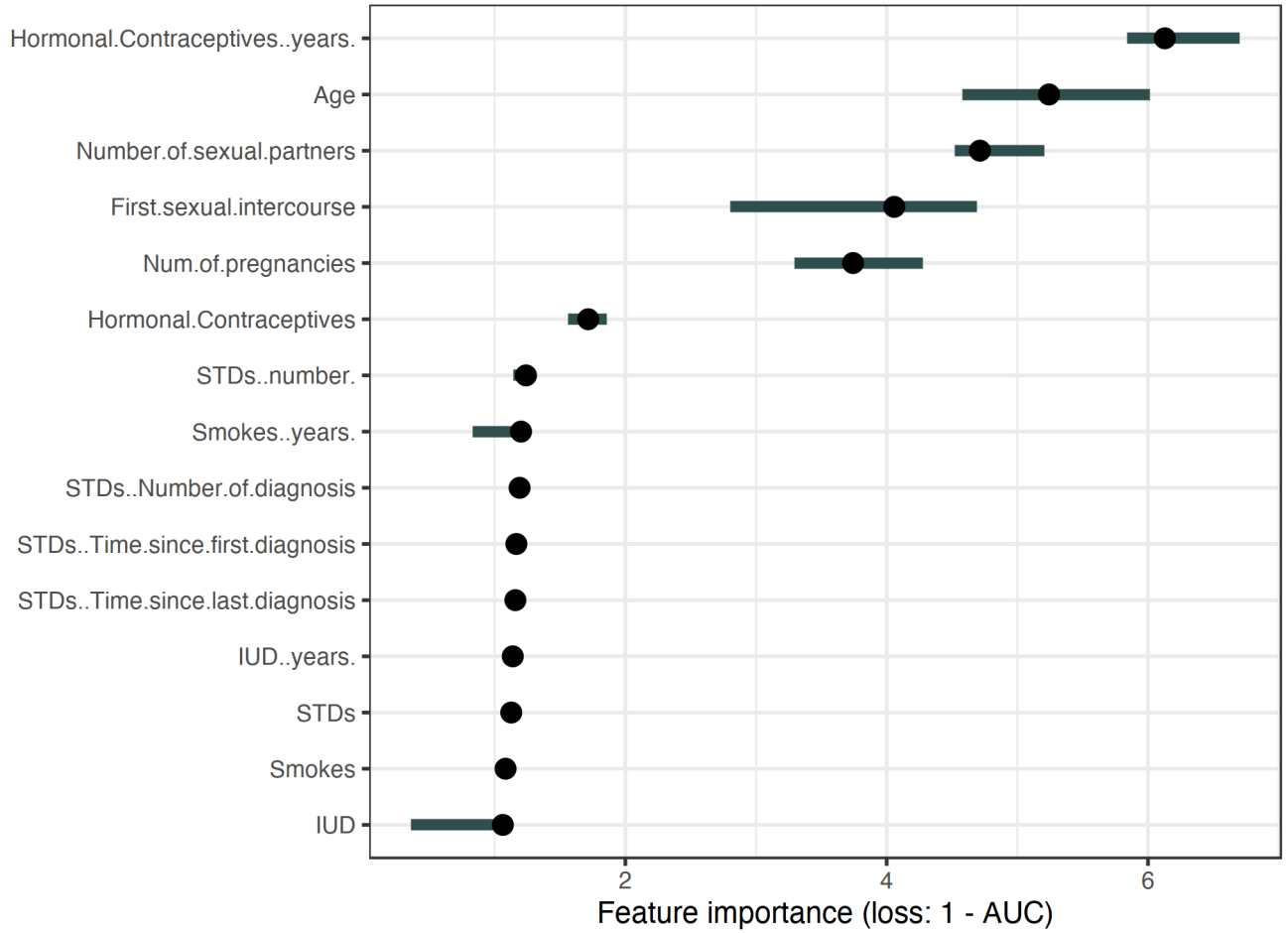
1. Compute original error $e_{orig} = L(y, \hat{f}(X))$.
2. For each feature j:
    1. Generate X_{perm} by permuting feature j in X
    2. Compute $e_{perm} = L(y, \hat{f}(X_{perm}))$
    3. $FI_j = e_{perm}/e_{orig}$ or $FI_j = e_{perm} - e_{orig}$

This is a standard FI measure for Random forests, using out-of-bag samples to estimate error change (Breiman, 2001)

Permutation feature importance

Examples for the prediction of risk of cervical cancer



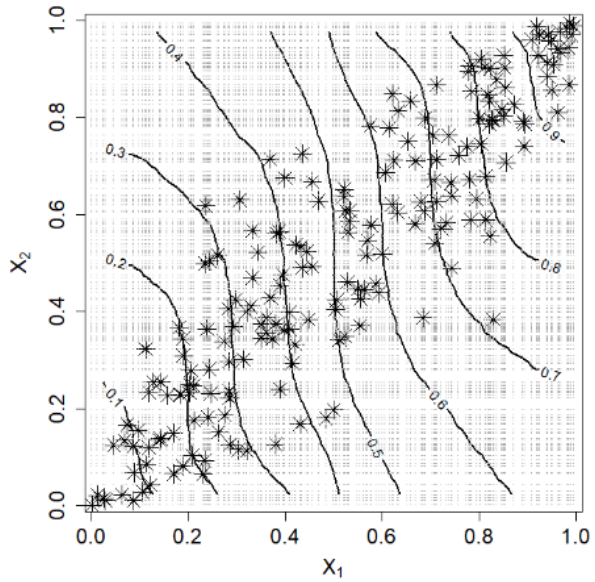


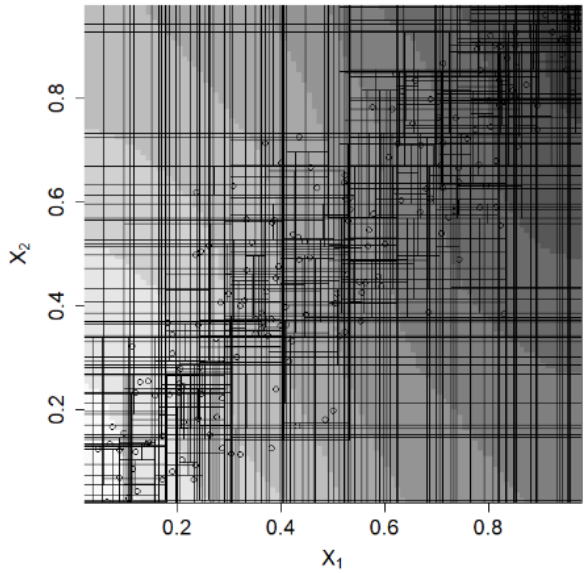
https://christophm.github.io/interpretable-ml-book/

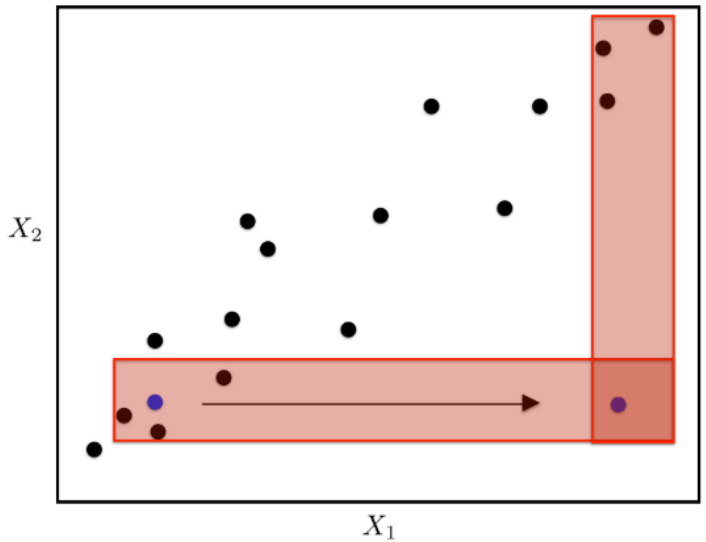
Permutation feature importance

Advantages: simple, very efficient, easy to interpret

Disadvantage: permuted instances can be out of distribution with respect to the model (e.g., when features are correlated)



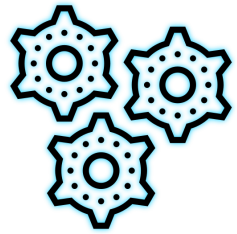




(Hooker and Mentch, 2019)

Corrections exist (e.g., permute-and-retrain) but they come with an additional computational cost

Approaches for Post hoc Explainability



Local Explanations

- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

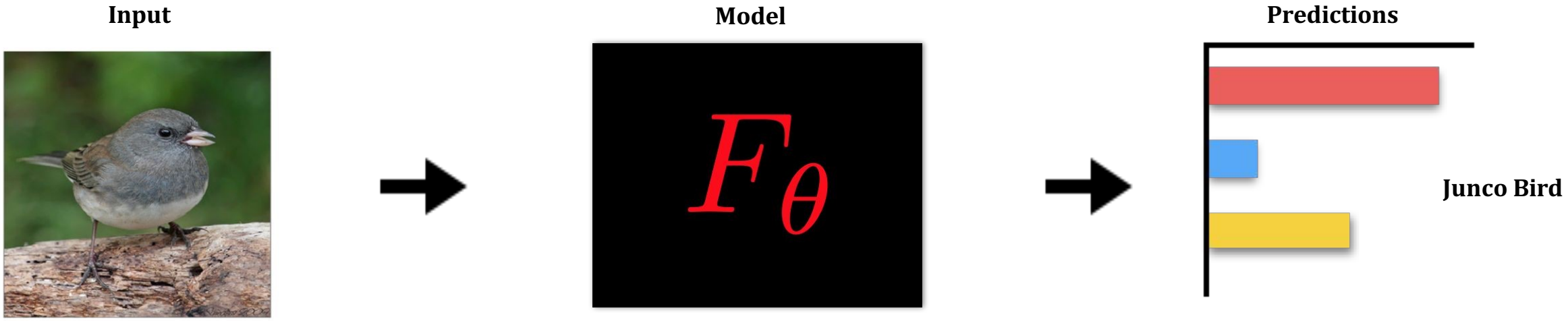
- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

Representation Based Approaches

- Derive model understanding by analyzing intermediate representations of a DNN.
- Determine model's reliance on 'concepts' that are semantically meaningful to humans.

Representation Based Approaches

- Derive model understanding by analyzing intermediate representations of a DNN.
- Determine model's reliance on 'concepts' that are semantically meaningful to humans.



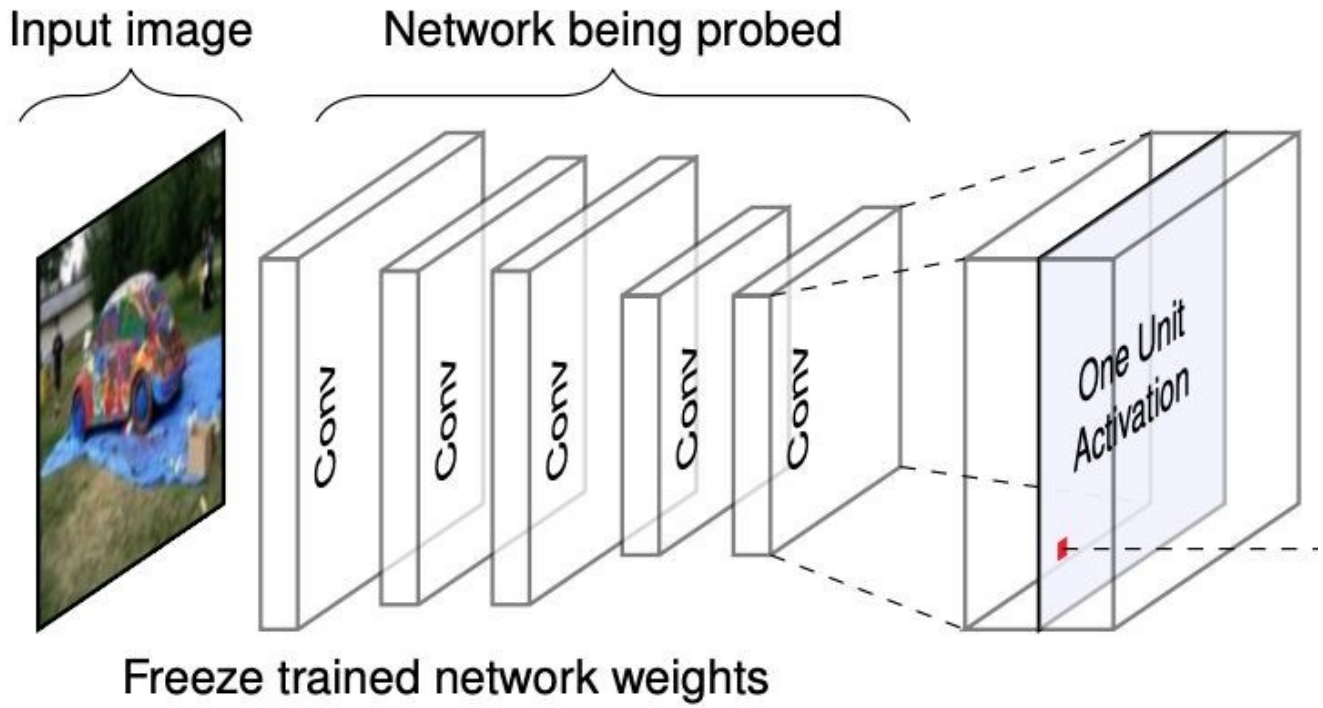
Does the model rely on the **'green background'**?

Network Dissection

1. Identify a broad set of human-labeled visual concepts.

Bau & Zhou et. al. 2017

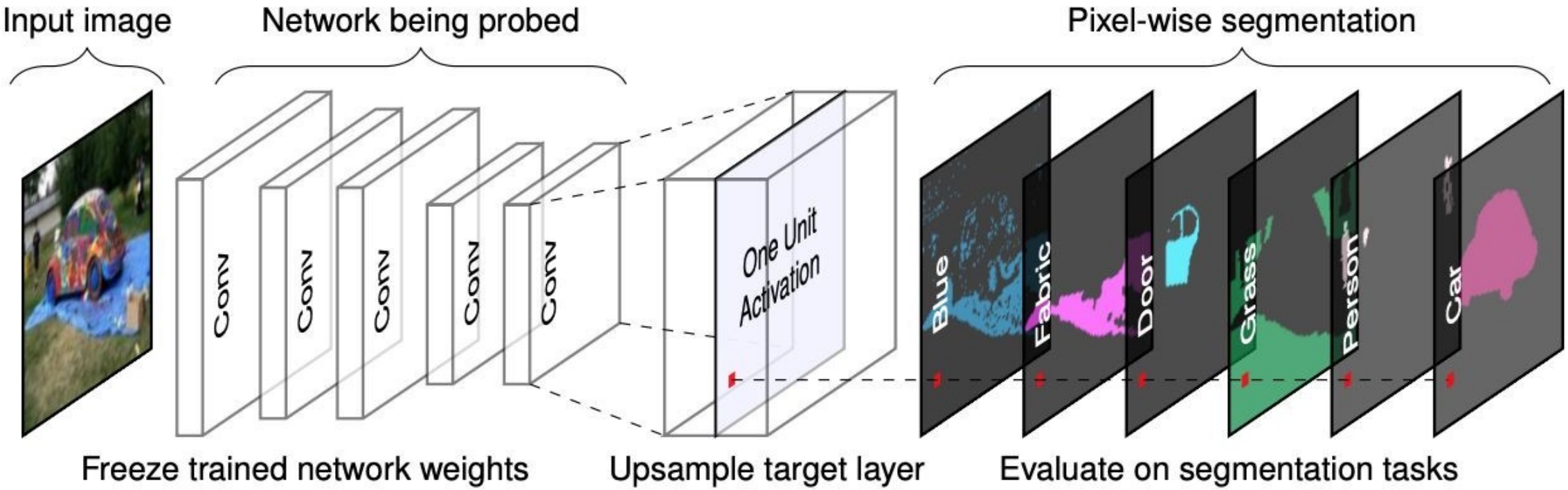
Network Dissection



1. Identify a broad set of human-labeled visual concepts.
2. Gather the response of hidden variables (convolutional filters) to known concepts.

Bau & Zhou et. al. 2017

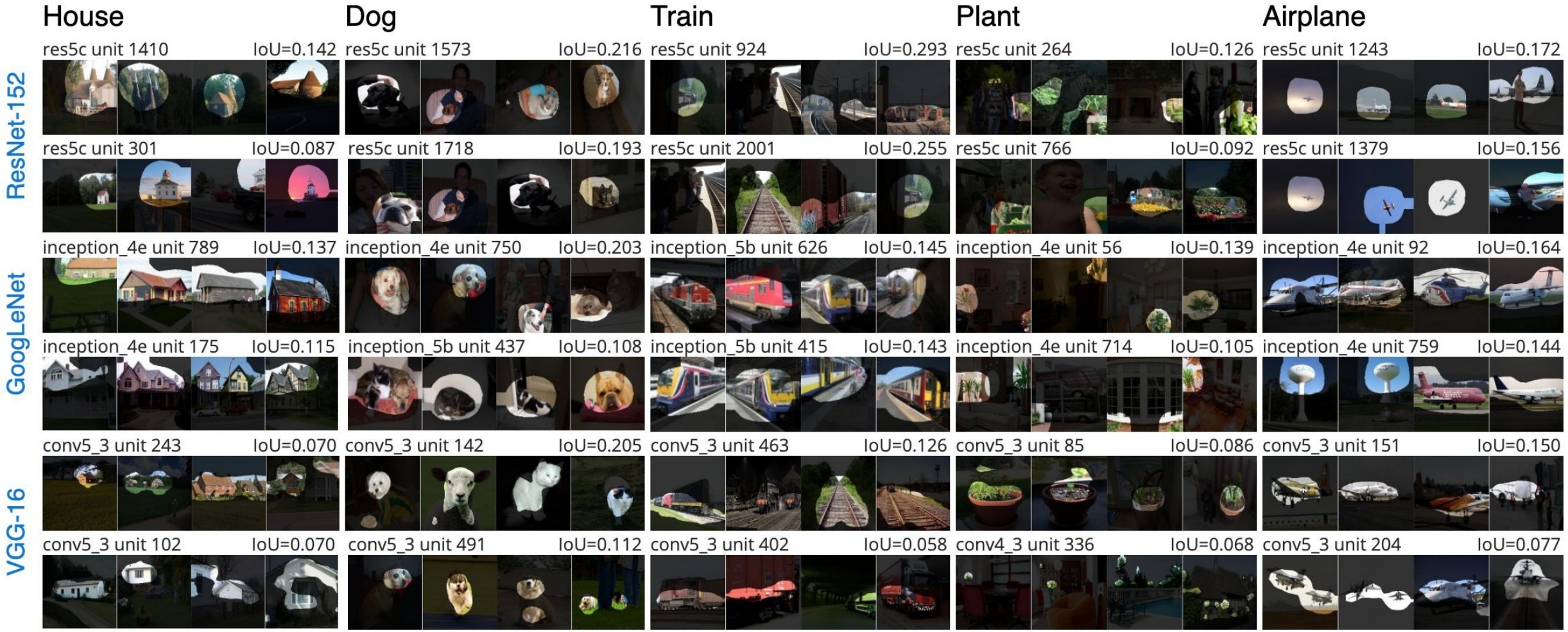
Network Dissection



1. Identify a broad set of human-labeled visual concepts.
2. Gather the response of hidden variables (convolutional filters) to known concepts.
3. Quantify alignment of hidden variable-concept pairs

Bau & Zhou et. al. 2017

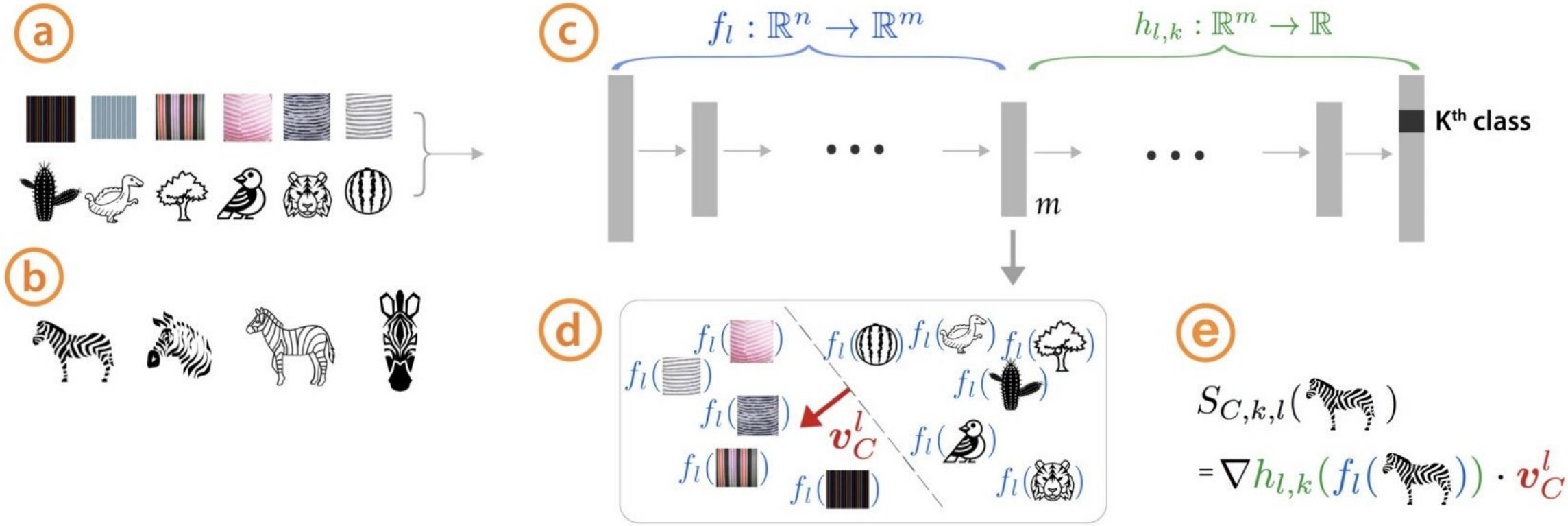
Network Dissection



[Bau & Zhou et. al. 2017]

Quantitative Testing with Concept Activation Vectors (TCAV)

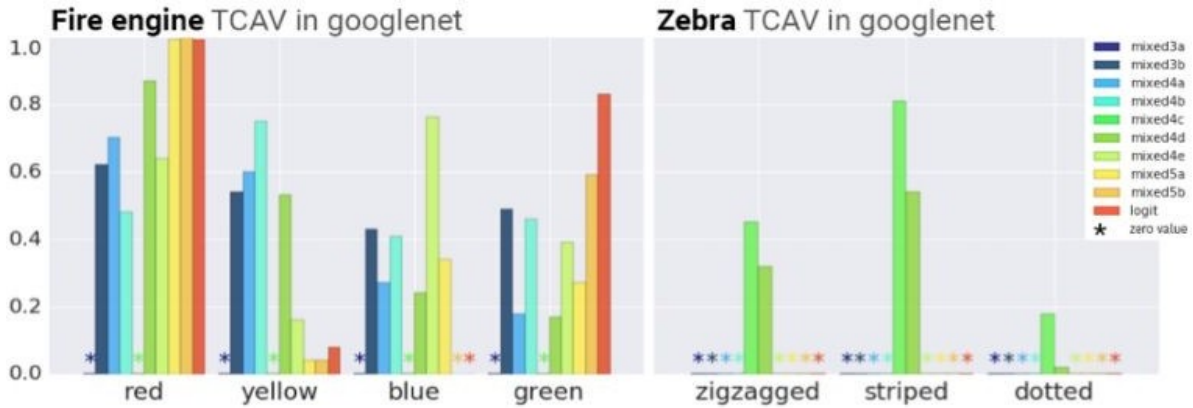
TCAV measures the sensitivity of a model's prediction to **user provided concept** using the model **internal representations**.

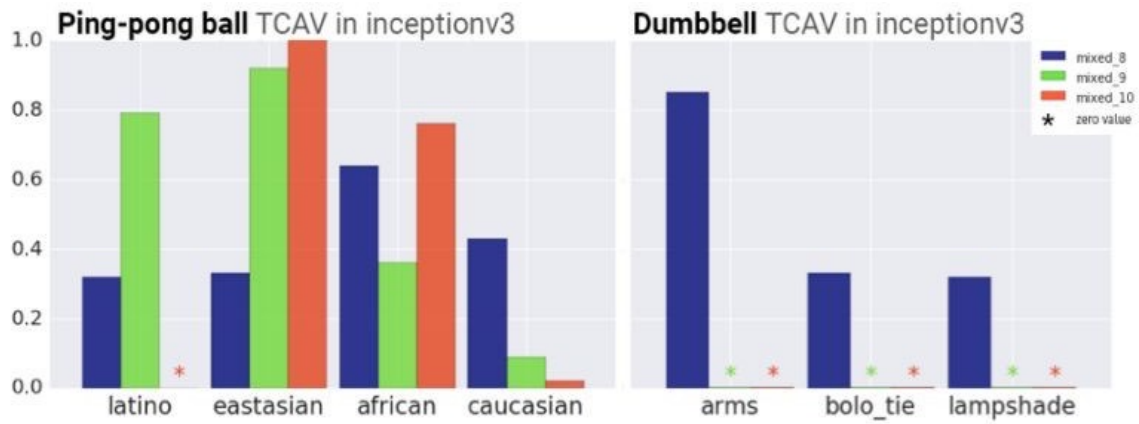


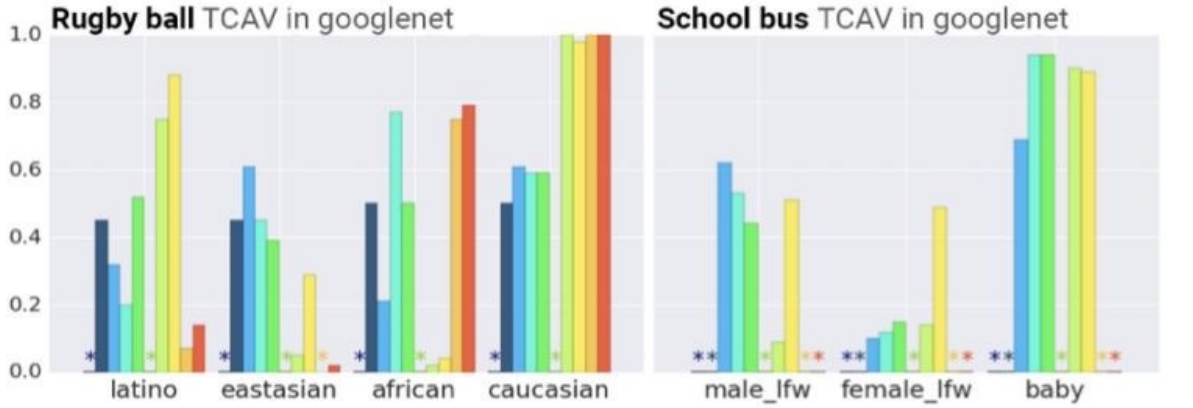
[Kim et. al. 2018]

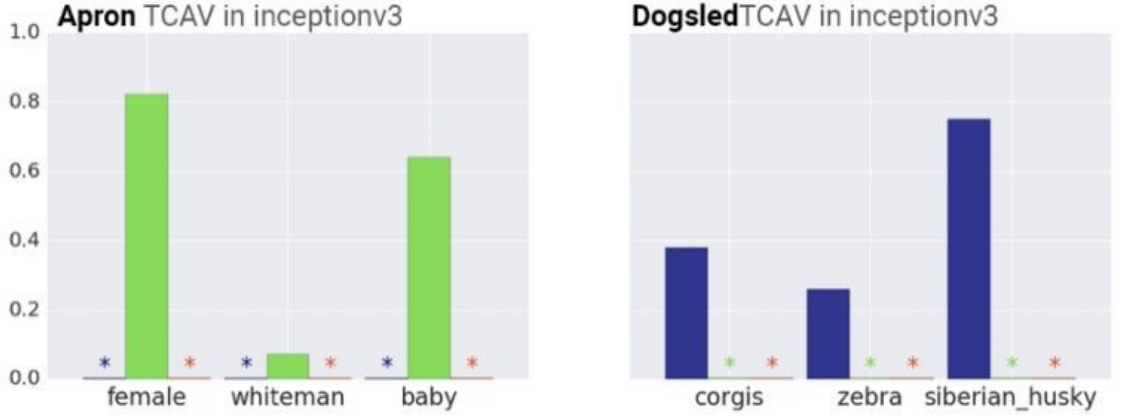
Quantitative Testing with Concept Activation Vectors (TCAV)

Insights from Googlenet and Inceptionv3





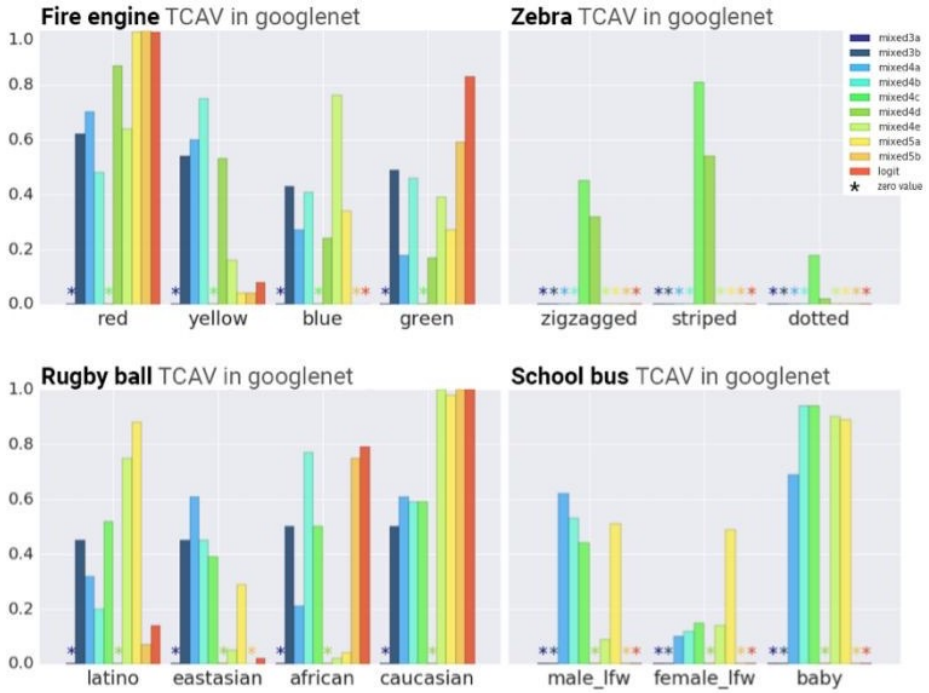


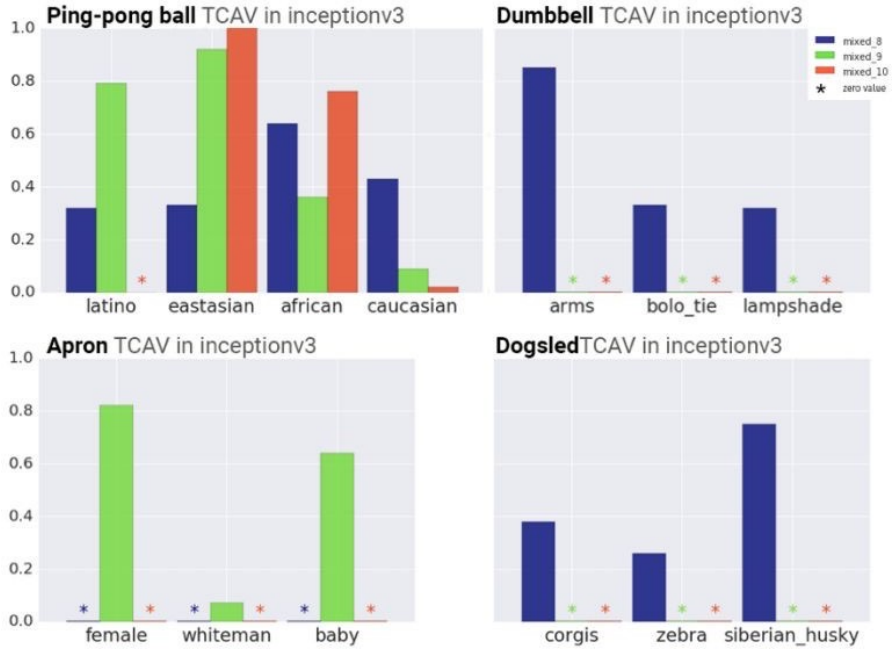


Images from [Kim et. al. 2018]

Quantitative Testing with Concept Activation Vectors (TCAV)

Insights from Googlenet and Inceptionv3

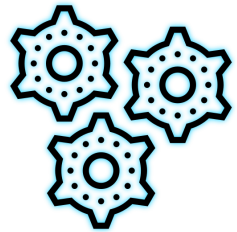




Additional Variants:

- Regression problems in medical domain (Graziani et. al. 2019).
- Automatic extraction of visual concepts (Ghorbani et. al. 2019).

Images from **Kim et. al. 2018**



Approaches for Post hoc Explainability

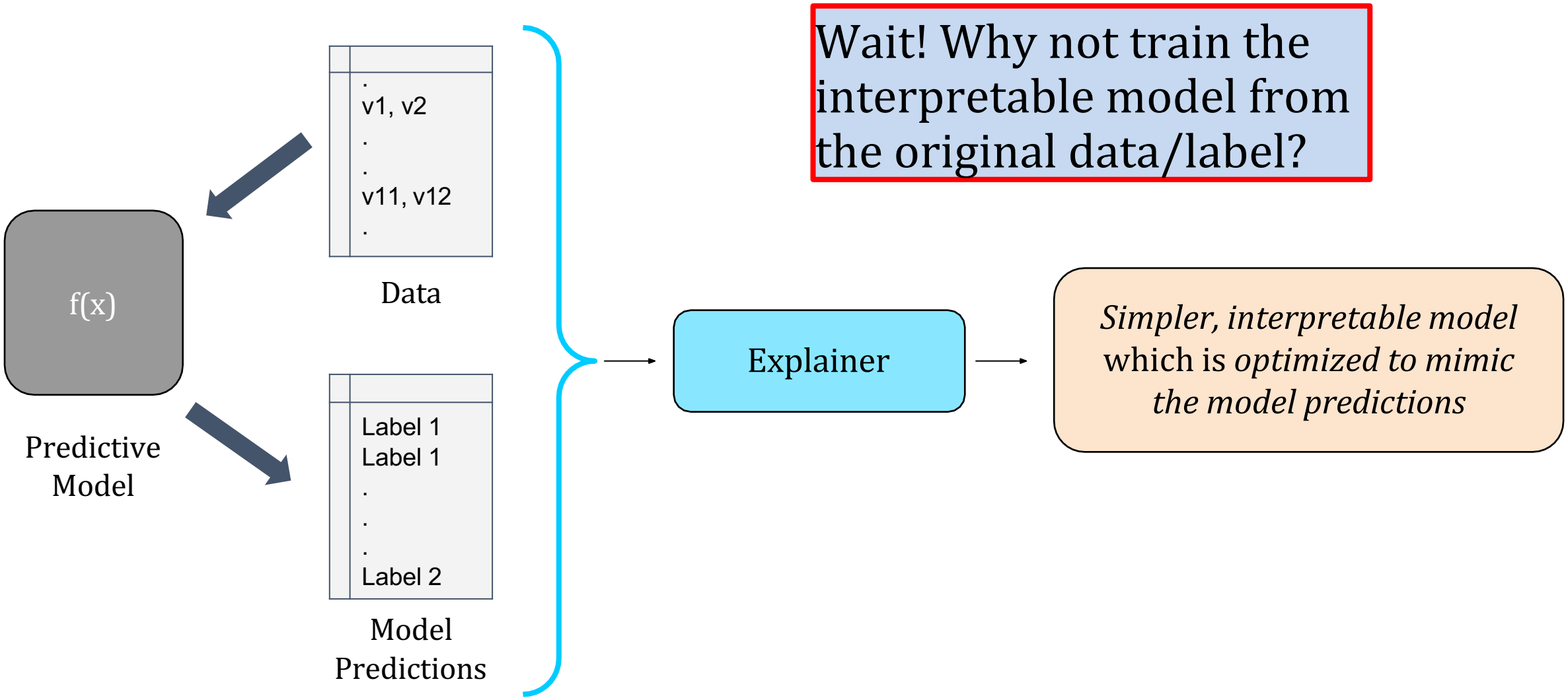
Local Explanations

- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

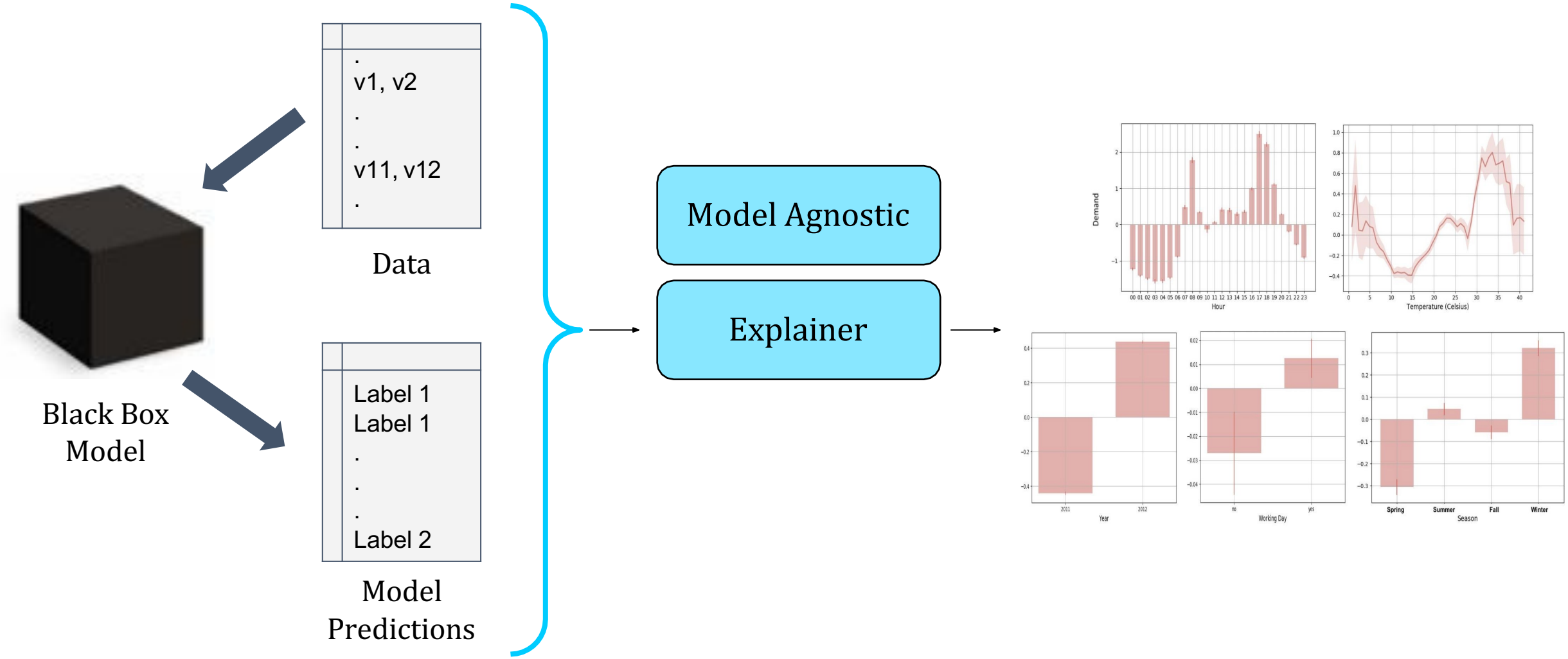
- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

Model Distillation for Generating Global Explanations



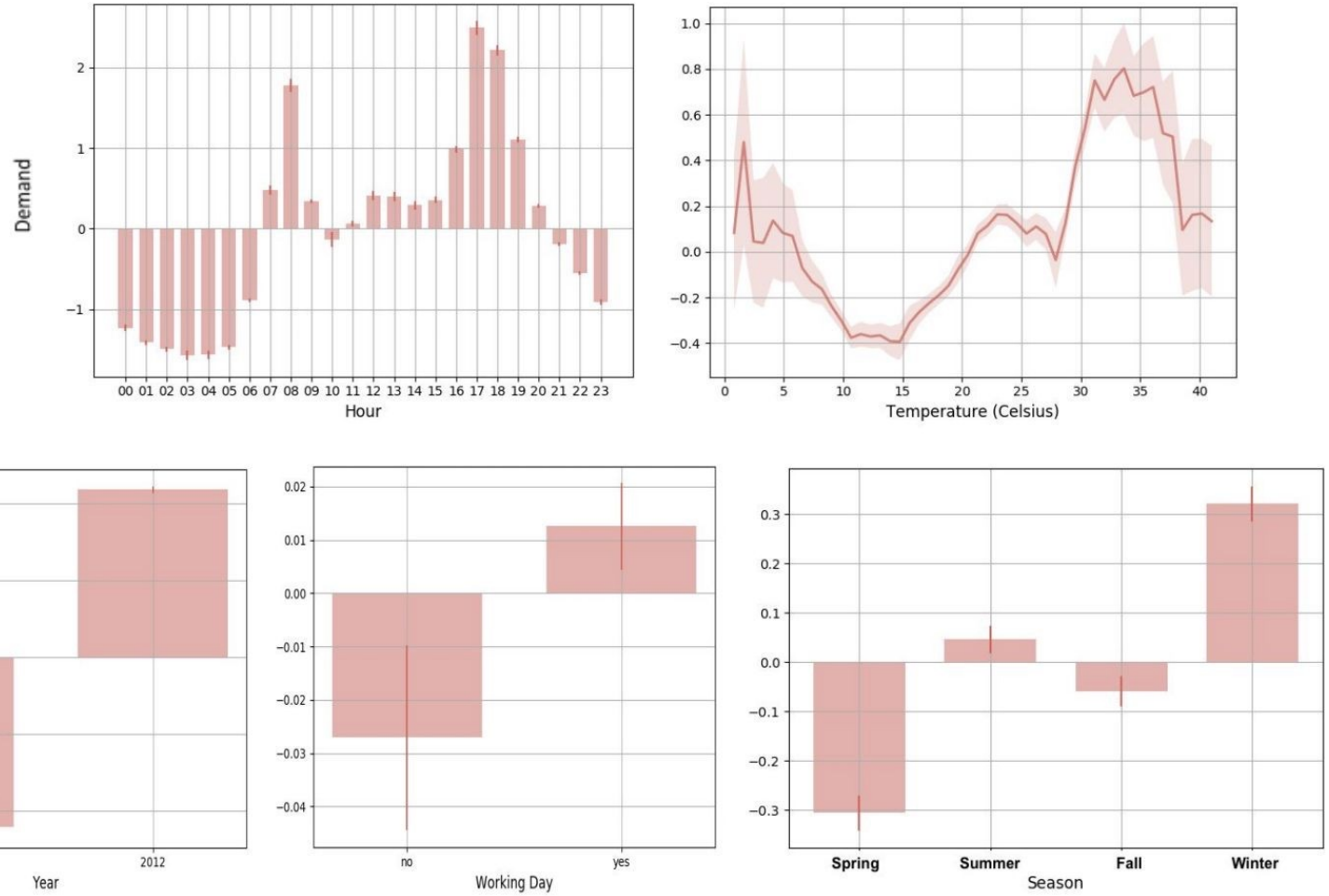
[Tan et. al., 2019]

Generalized Additive Models as Global Explanations



[Tan et. al., 2019]

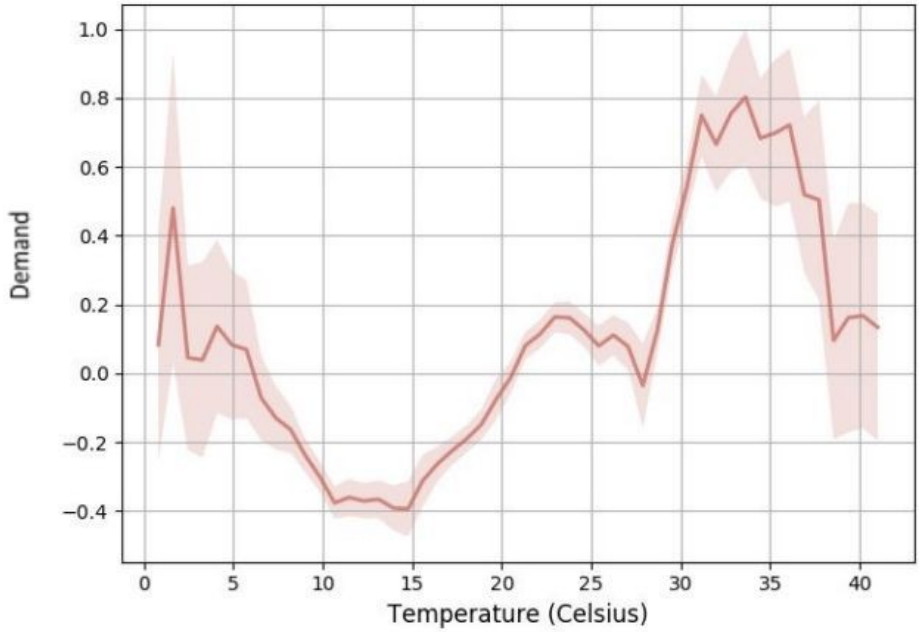
Generalized Additive Models as Global Explanations: *Shape Functions* for Predicting Bike Demand



[Tan et. al., 2019]

Generalized Additive Models as Global Explanations: *Shape Functions* for Predicting Bike Demand

How does bike demand vary as a function of temperature?



[Tan et. al., 2019]

Generalized Additive Models as Global Explanations

Generalized Additive Model (GAM) :

$$\hat{y} = h_0 + \sum_i h_i(x_i) + \sum_{i \neq j} h_{ij}(x_i, x_j) + \sum_{i \neq j} \sum_{j \neq k} h_{ijk}(x_i, x_j, x_k) + \cdots$$

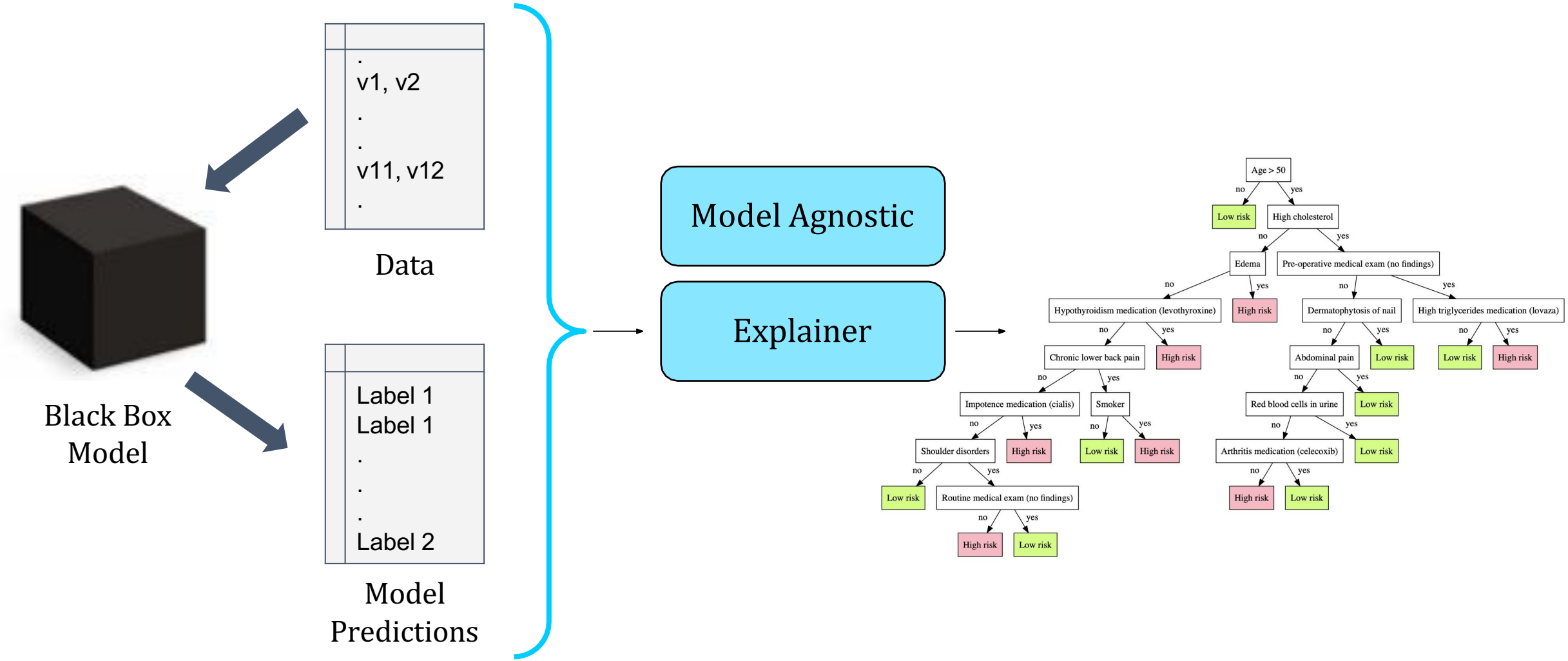
Shape functions of individual features

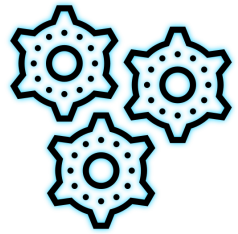
Higher order feature interaction terms

Fit this model to the predictions of the black box to obtain the shape functions.

[Bastani et. al., 2019]

Decision Trees as Global Explanations





Approaches for Post hoc Explainability

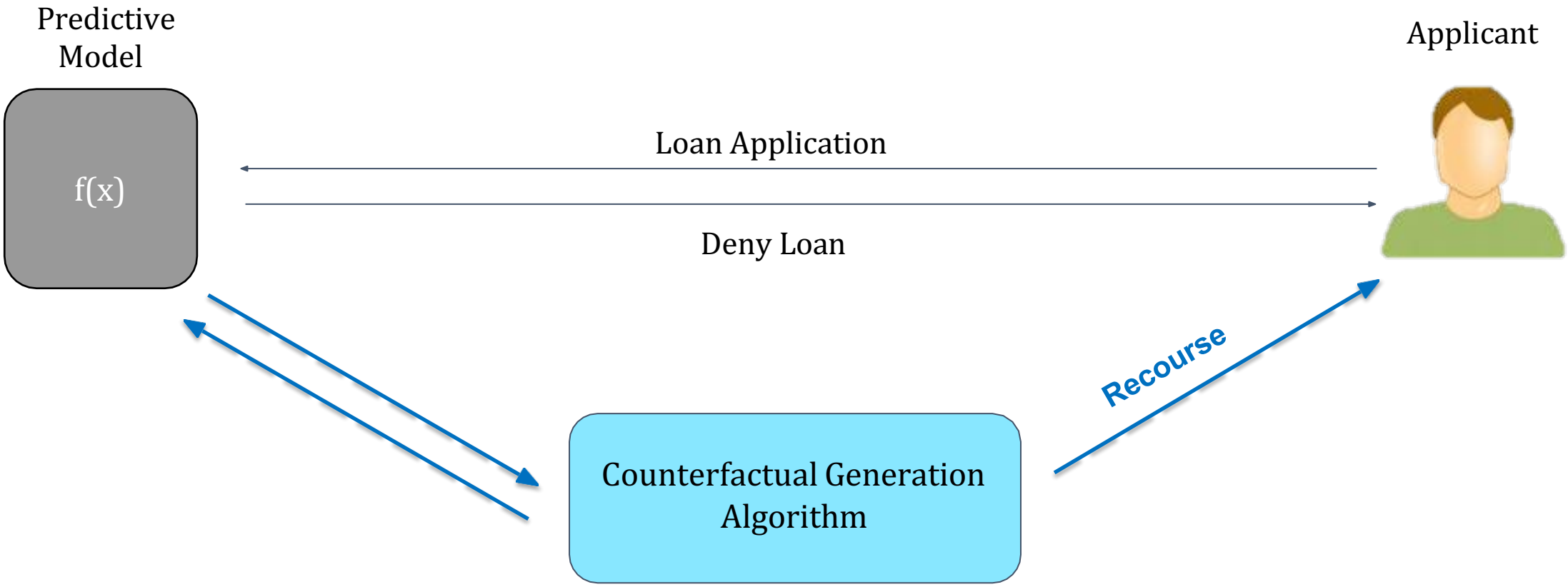
Local Explanations

- Feature Importances
- Rule Based
- Saliency Maps
- Prototypes/Example Based
- Counterfactuals

Global Explanations

- Collection of Local Explanations
- Representation Based
- Model Distillation
- Summaries of Counterfactuals

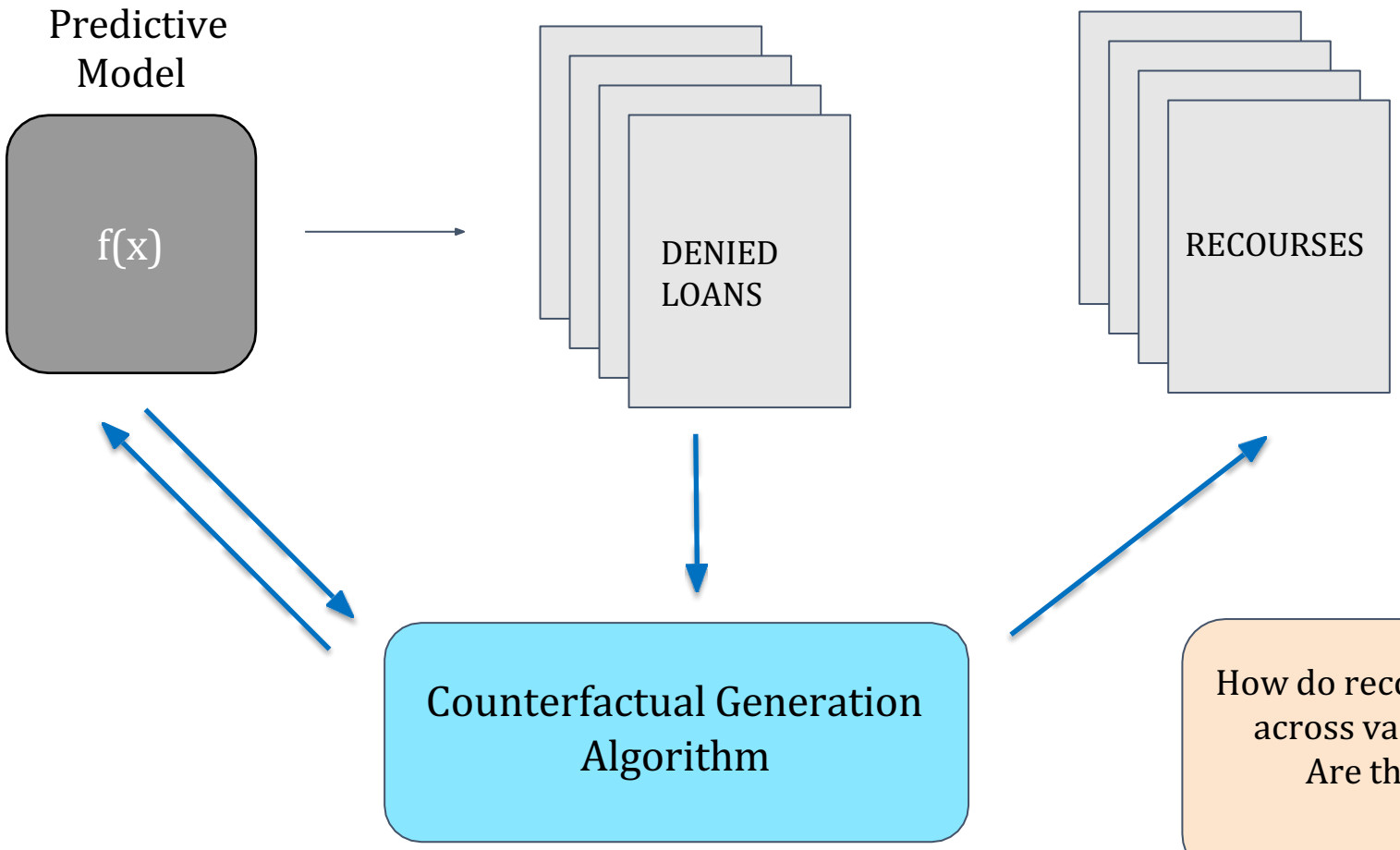
Counterfactual Explanations



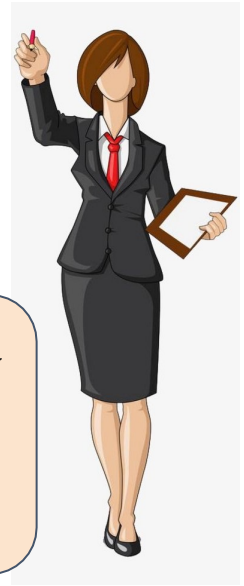
Recourse: Increase your salary by 50K & pay your credit card bills on time for next 3 months

[Rawal and Lakkaraju, 2020]

Counterfactual Explanations



Decision Maker
(or) Regulatory Authority



How do recourses permitted by the model vary across various racial & gender subgroups? Are there any biases against certain demographics?

[Rawal and Lakkaraju, 2020]

Customizable Global Summaries of Counterfactuals

Predictive Model

f(x)

DENIED LOANS

Algorithm for generating global summaries of counterfactuals

If Race = Caucasian **and** Gender = Male:

If Married =No **and** Property =No and Has Job =No, **then** Married =No **and** Property =No and Has Job =Yes

If Drugs =Yes **and** School =No **and** Pays Rent =No , **then** Drugs =No **and** School =No **and** Pays Rent =No

If Race = Caucasian **and** Gender = Female:

If Married =No **and** Property =No and Has Job =No, **then** Married =No **and** Property =No and Has Job =Yes

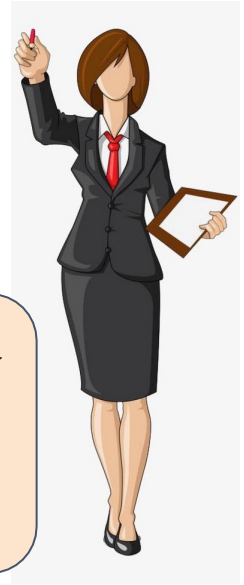
If Drugs =Yes **and** School =No **and** Pays Rent =No , **then** Drugs =No **and** School =No **and** Pays Rent =Yes

If Race ≠ Caucasian:

If Married =No **and** Property =No and Has Job =No, **then** Married =No **and** Property =Yes and Has Job =Yes

If Drugs =Yes **and** School =No **and** Pays Rent =No , **then** Drugs =No **and** School =Yes **and** Pays Rent =Yes

How do recourses permitted by the model vary across various racial & gender subgroups? Are there any biases against certain demographics?



[Rawal and Lakkaraju, 2020]

Customizable Global Summaries of Counterfactuals

Subgroup Descriptor

If Race = Caucasian **and** Gender = Male:

If Married =No **and** Property =No and Has Job =No, **then** Married =No **and** Property =No and Has Job =Yes

If Drugs =Yes **and** School =No **and** Pays Rent =No , **then** Drugs =No **and** School =No **and** Pays Rent =No

If Race = Caucasian **and** Gender = Female:

If Married =No **and** Property =No and Has Job =No, **then** Married =No **and** Property =No and Has Job =Yes

If Drugs =Yes **and** School =No **and** Pays Rent =No , **then** Drugs =No **and** School =No **and** Pays Rent =Yes

Recourse Rules

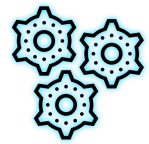
If Race ≠ Caucasian:

If Married =No **and** Property =No and Has Job =No, **then** Married =No **and** Property =Yes and Has Job =Yes

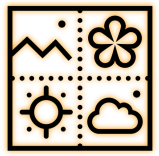
If Drugs =Yes **and** School =No **and** Pays Rent =No , **then** Drugs =No **and** School =Yes **and** Pays Rent =Yes

Omg! this model is biased. It requires certain demographics to "act upon" lot more features than others.

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



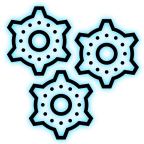
~~Explanations in Different Modalities~~

Evaluation of Explanations

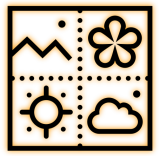
Limits of Post hoc Explainability

Future of Post hoc Explainability

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



~~Explanations in Different Modalities~~

Evaluation of Explanations

Limits of Post hoc Explainability

Future of Post hoc Explainability

Evaluation of Post hoc Explanations

[Doshi-Velez and Kim, 2017]

How we evaluate explanations?

[Lage et al, 2019]

Two Different Factors

What are you evaluating?

How we evaluate it?

	Understand the Behavior	Useful for Debugging	Help make decisions
Application-grounded			
Human-grounded			
Functionally-grounded			

Evaluating Post hoc Explanations

Understand the Behavior

Help make decisions

Useful for Debugging

Evaluating Post hoc Explanations

Understand the Behavior

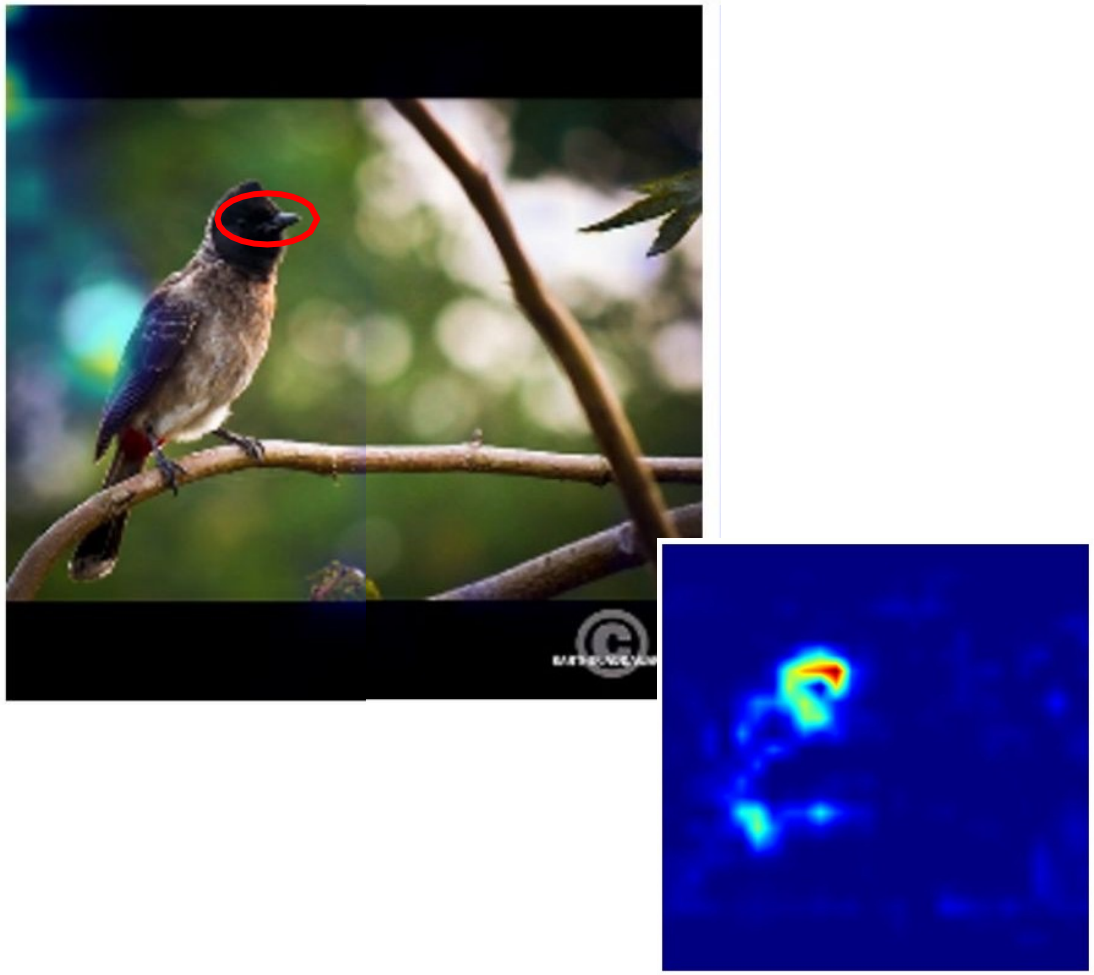
Help make decisions

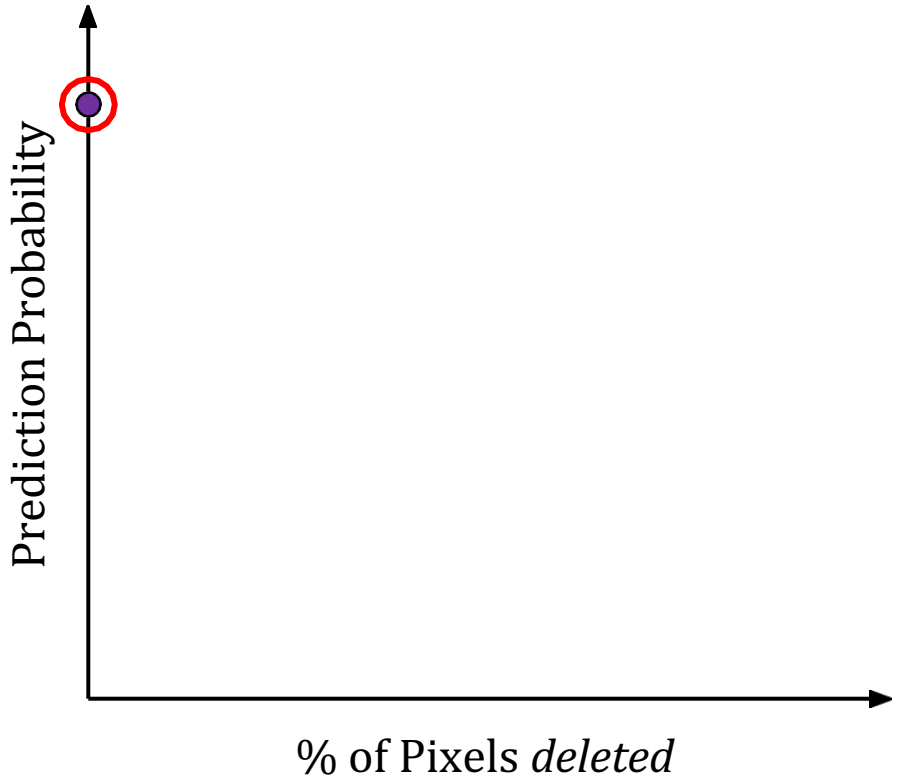
Useful for Debugging

[Qi, Khorram, Fuxin, 2020]

How important are selected features?

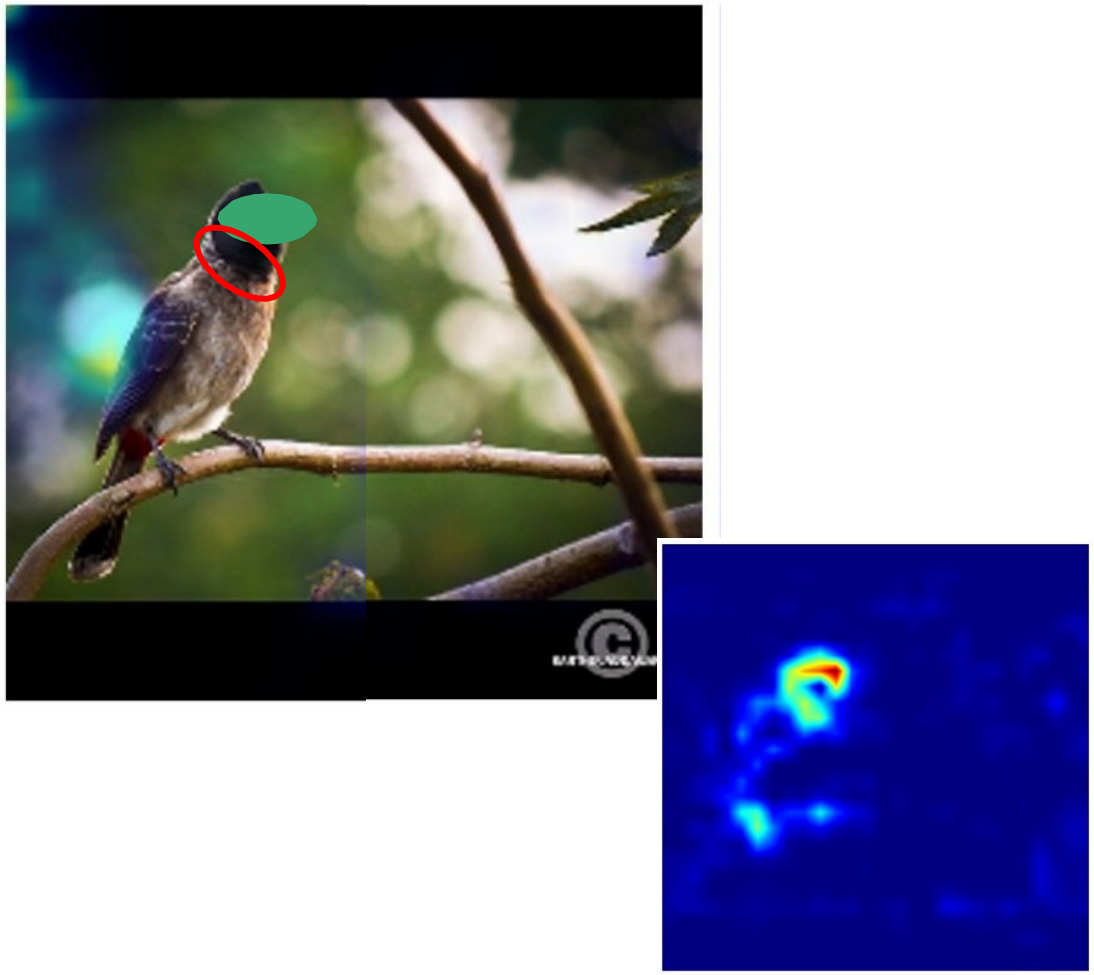
- Deletion: remove important features and see what happens..

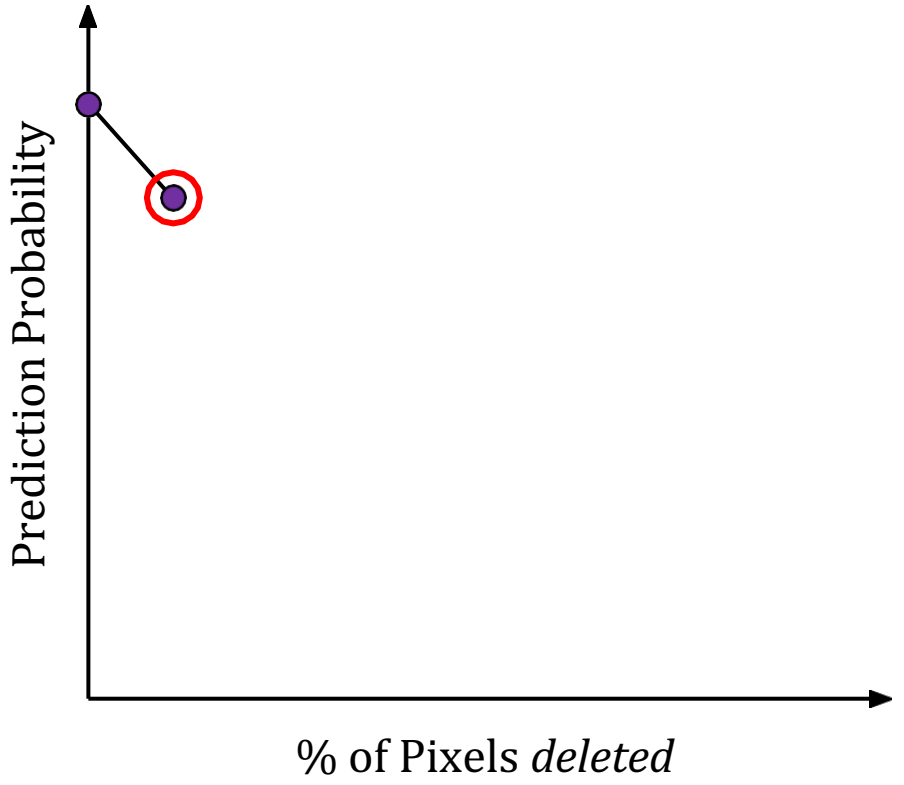




How important are selected features?

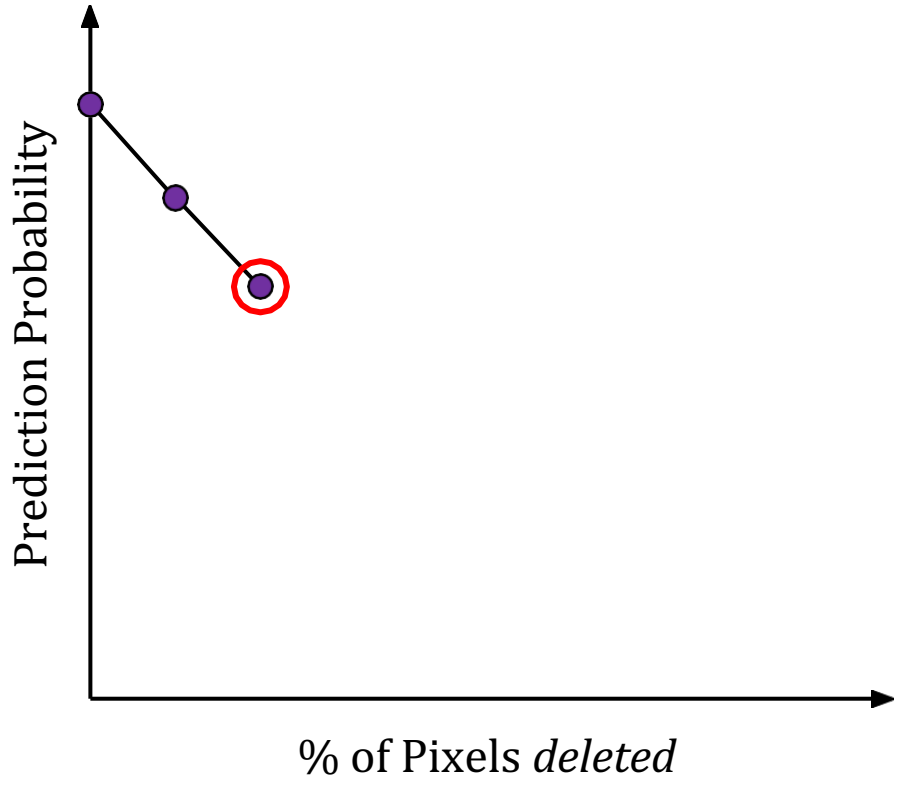
- Deletion: remove important features and see what happens..





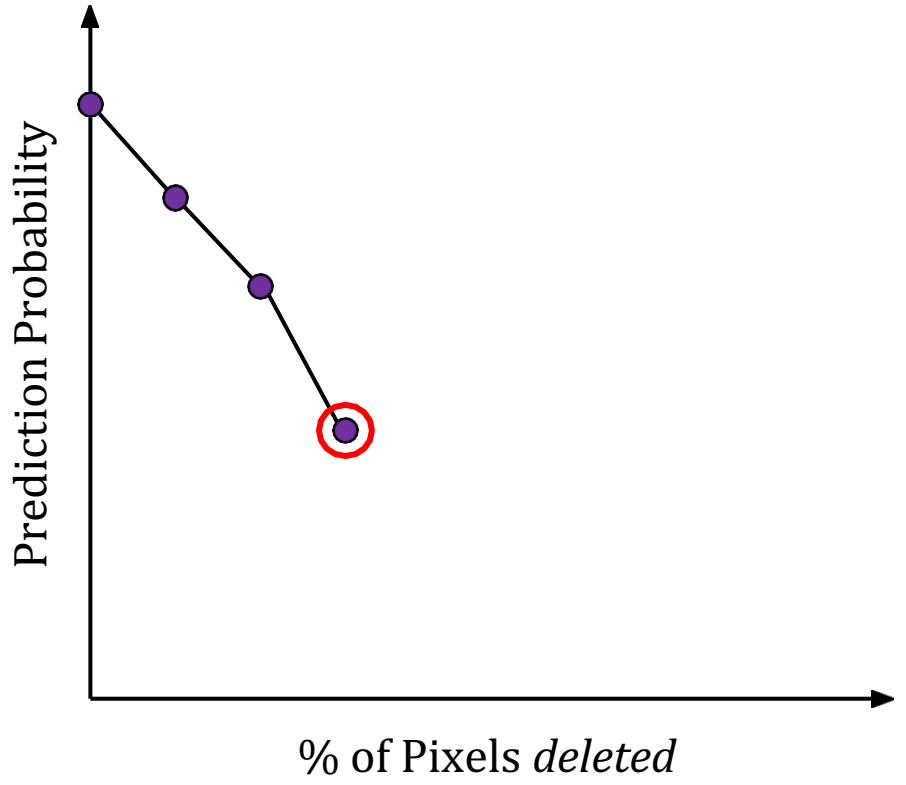
How important are selected features?

- Deletion: remove important features and see what happens..



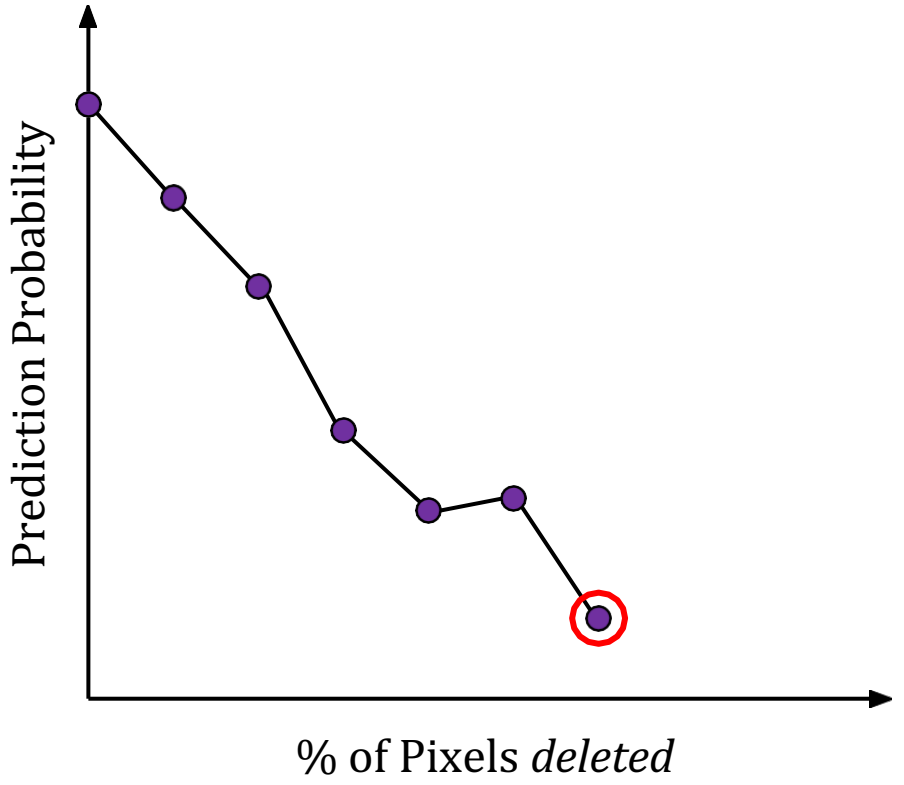
How important are selected features?

- Deletion: remove important features and see what happens..



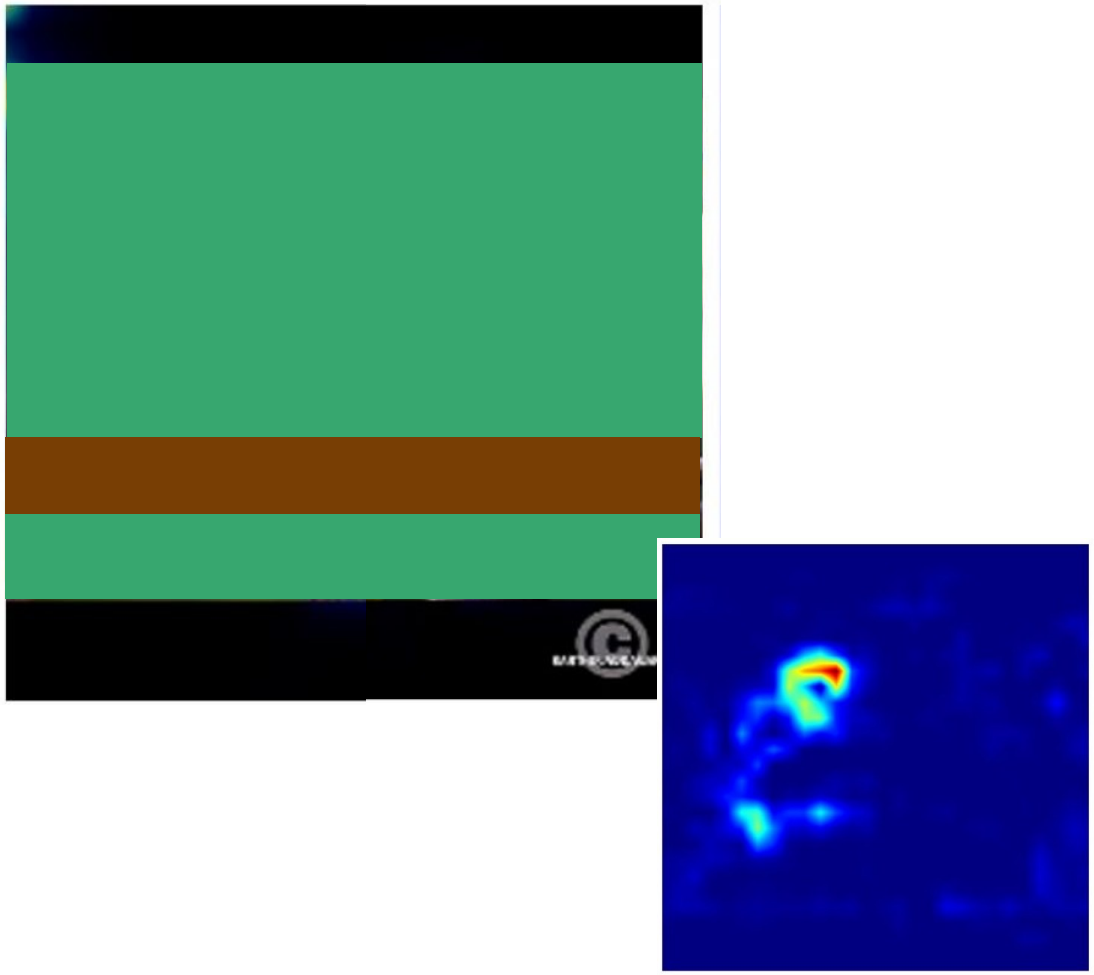
How important are selected features?

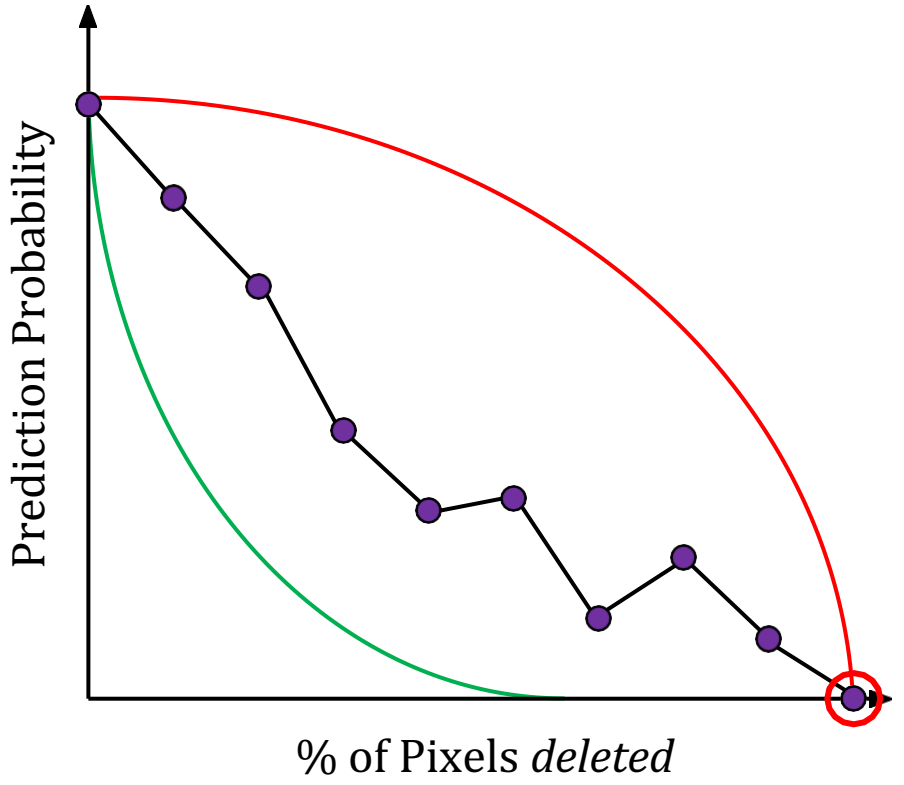
- Deletion: remove important features and see what happens..



How important are selected features?

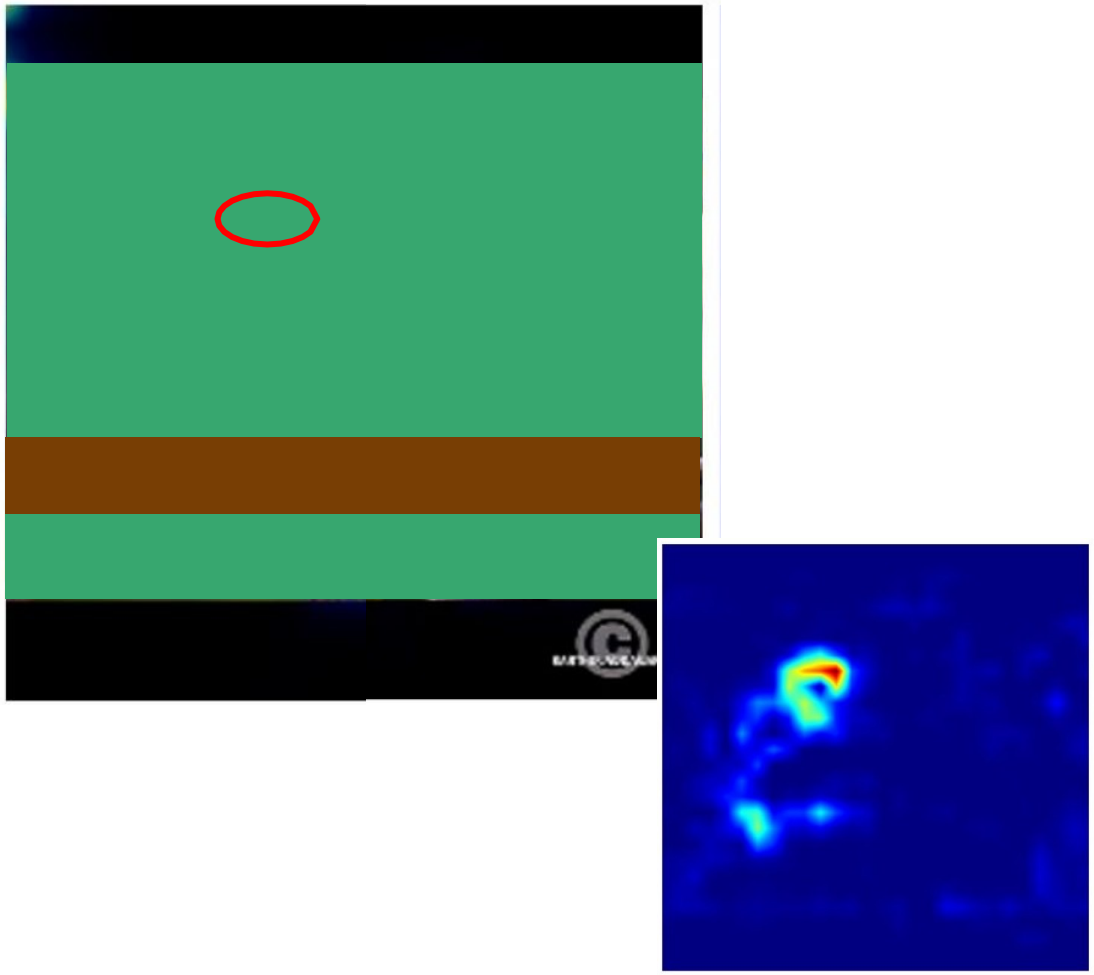
- Deletion: remove important features and see what happens..

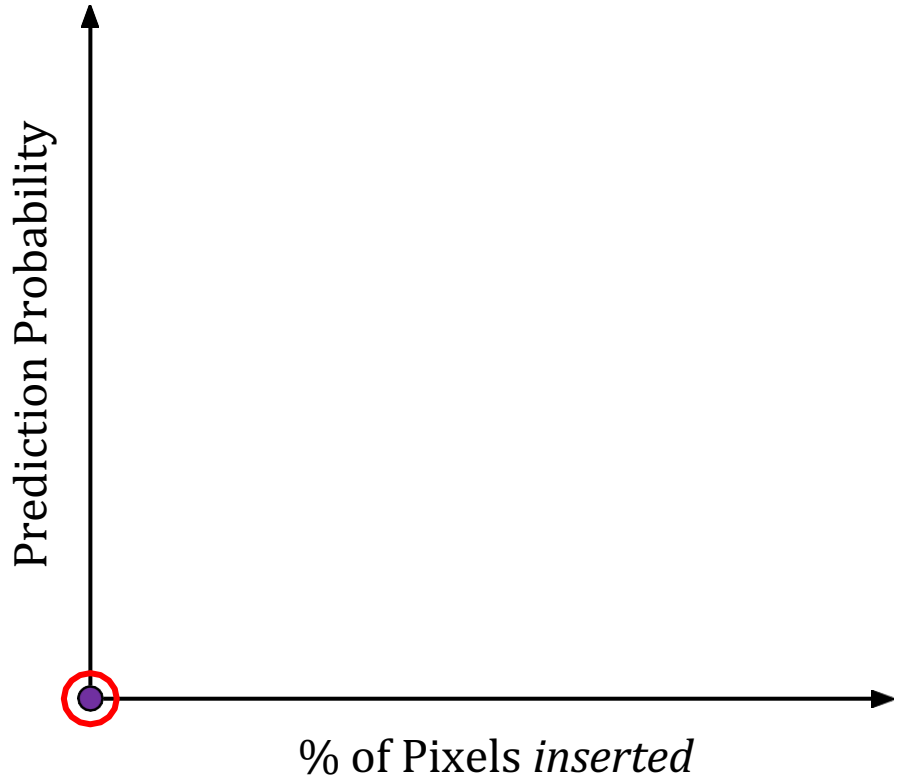




How important are selected features?

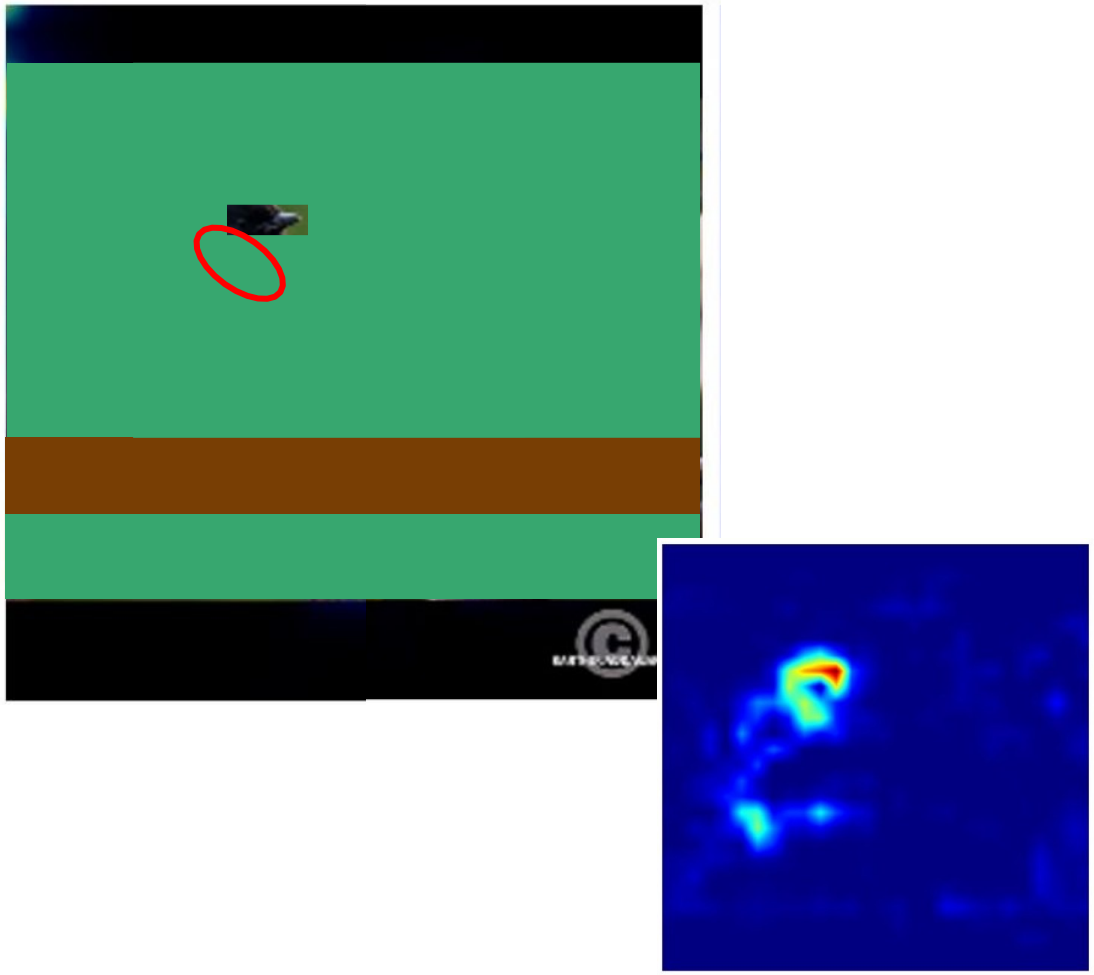
- Insertion: add important features and see what happens..

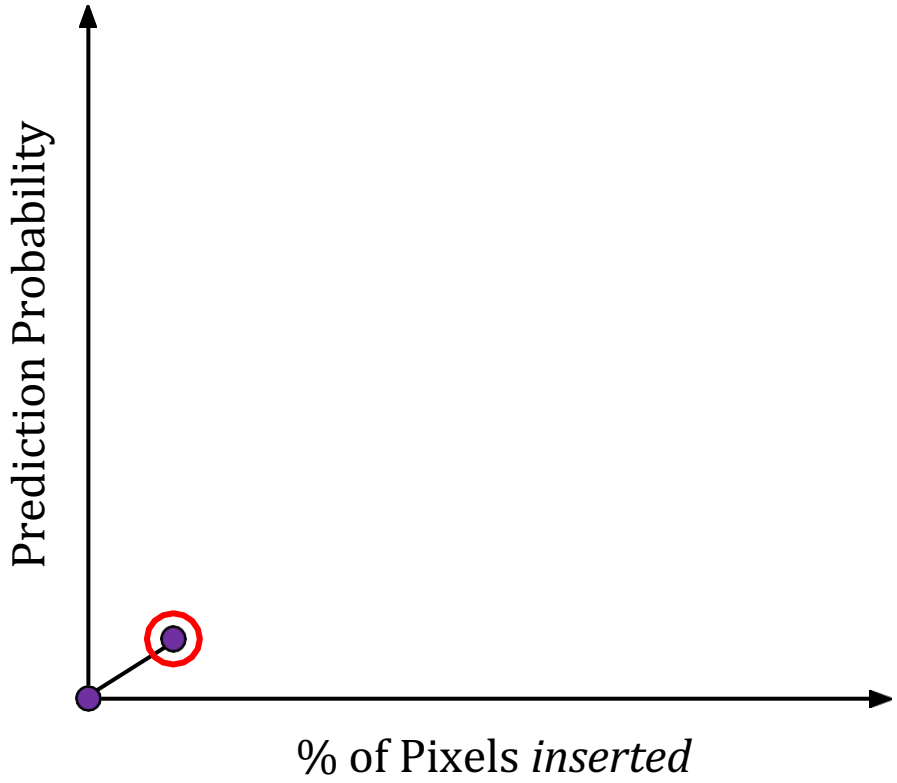




How important are selected features?

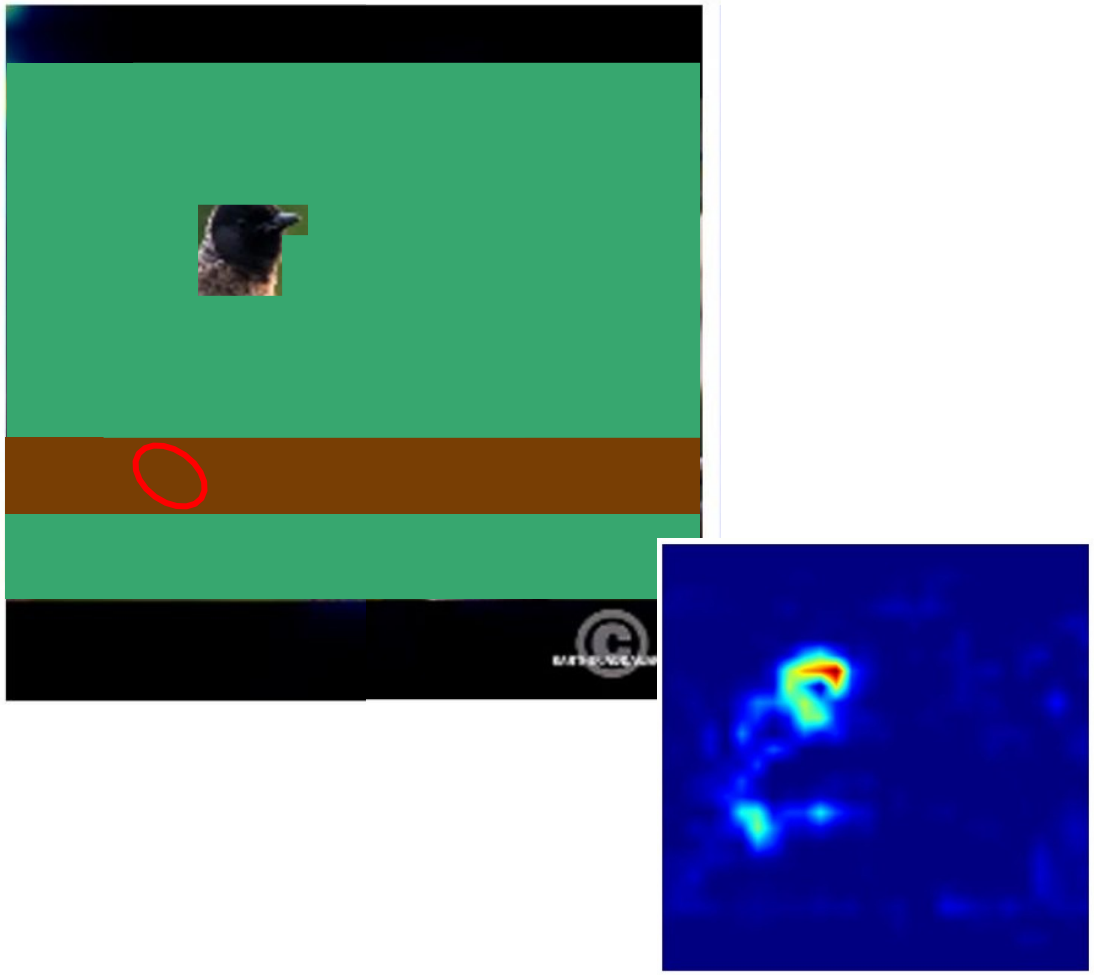
- Insertion: add important features and see what happens..

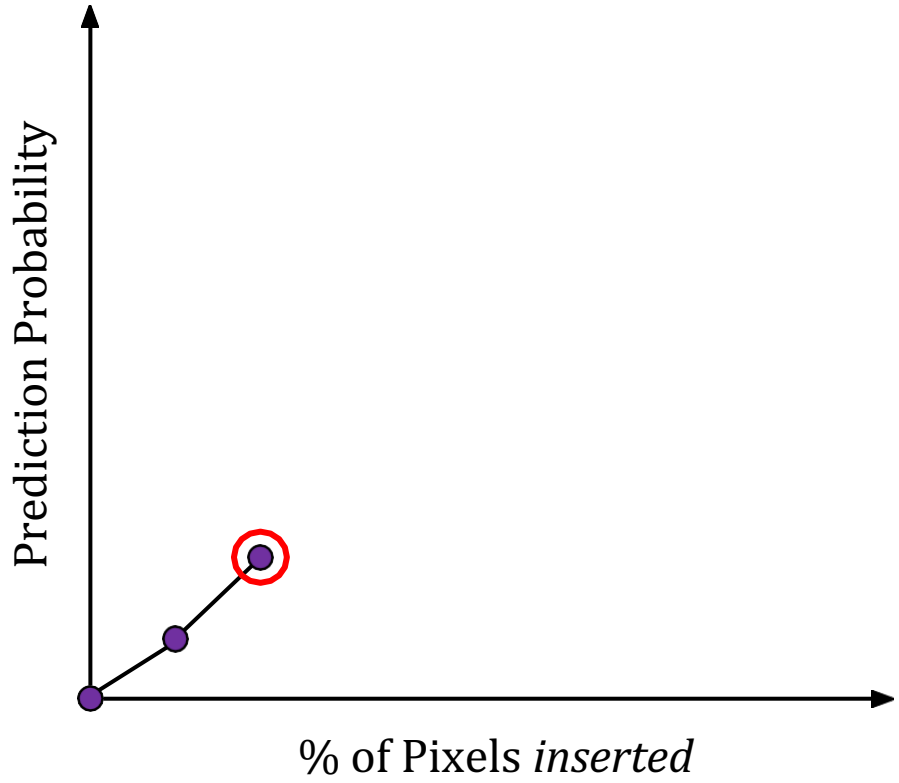




How important are selected features?

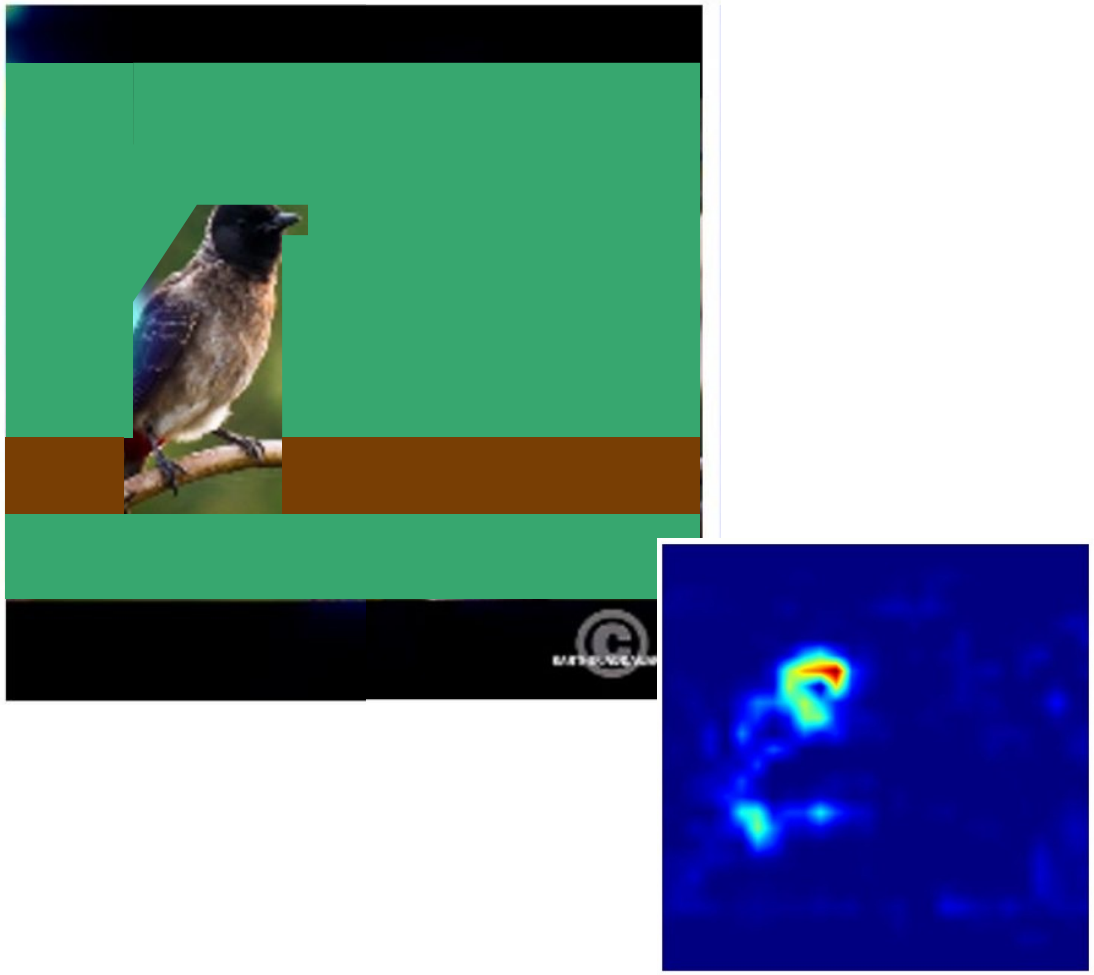
- Insertion: add important features and see what happens..

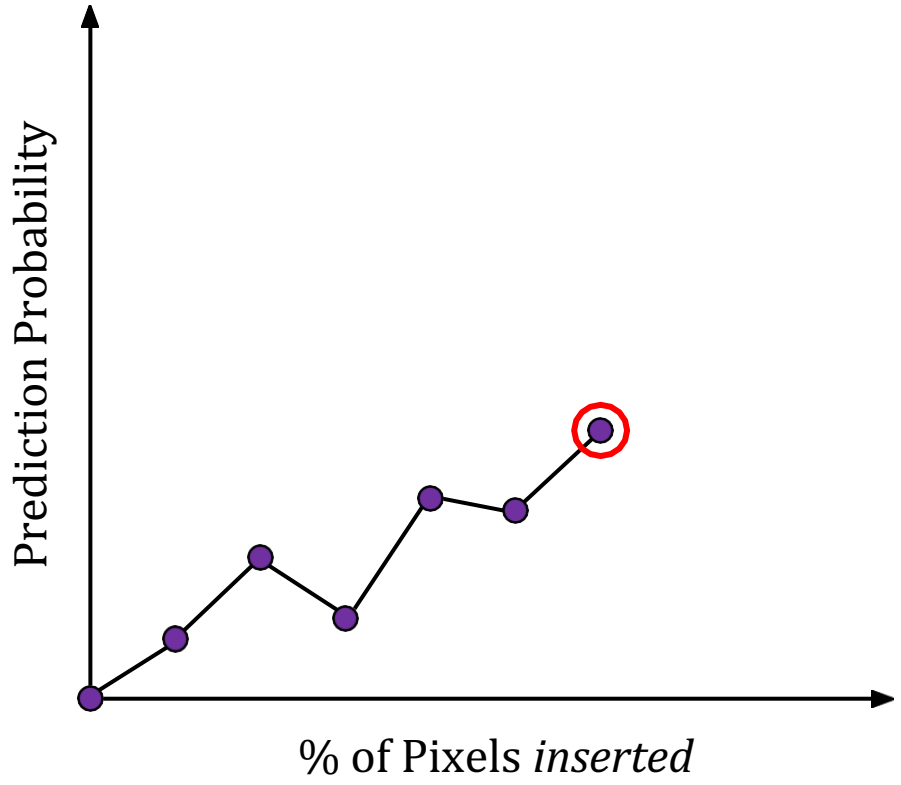




How important are selected features?

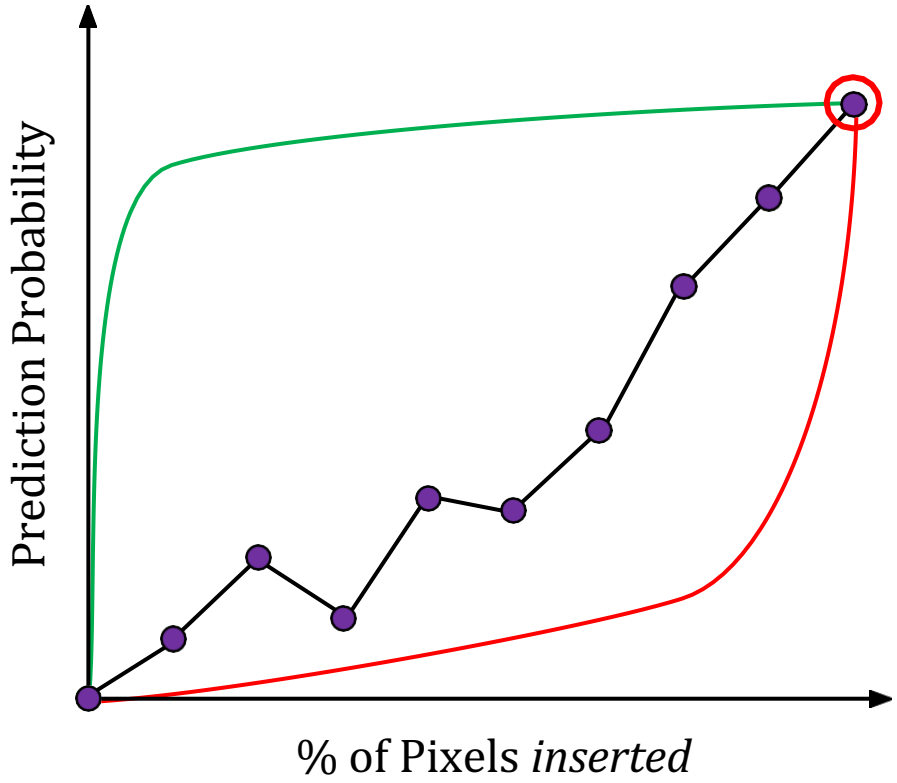
- Insertion: add important features and see what happens..





How important are selected features?

- Insertion: add important features and see what happens..



Limitation of the approach

As discussed earlier, pertubed examples are out of distribution with respect to training examples.

⇒ Loss of performance might be due to the pertubation, not because of the information that is hidden.

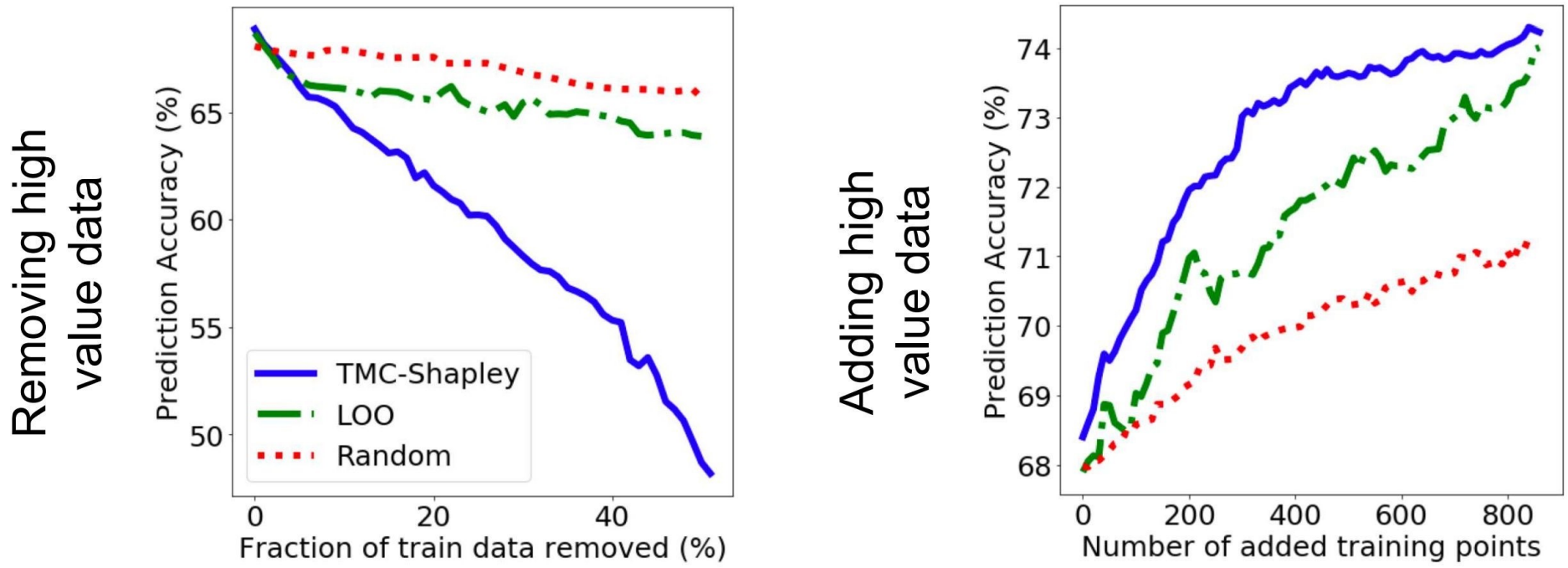
Alternative: retrain the model on modified examples (ROAR, for RemOve And Retrain, Kim et al., NeurIPS 2019)

Still not totally satisfactory: does not handle well redundant features (why?) and shape of removed areas can be informative.

[Ghorbani and Zou 2019]

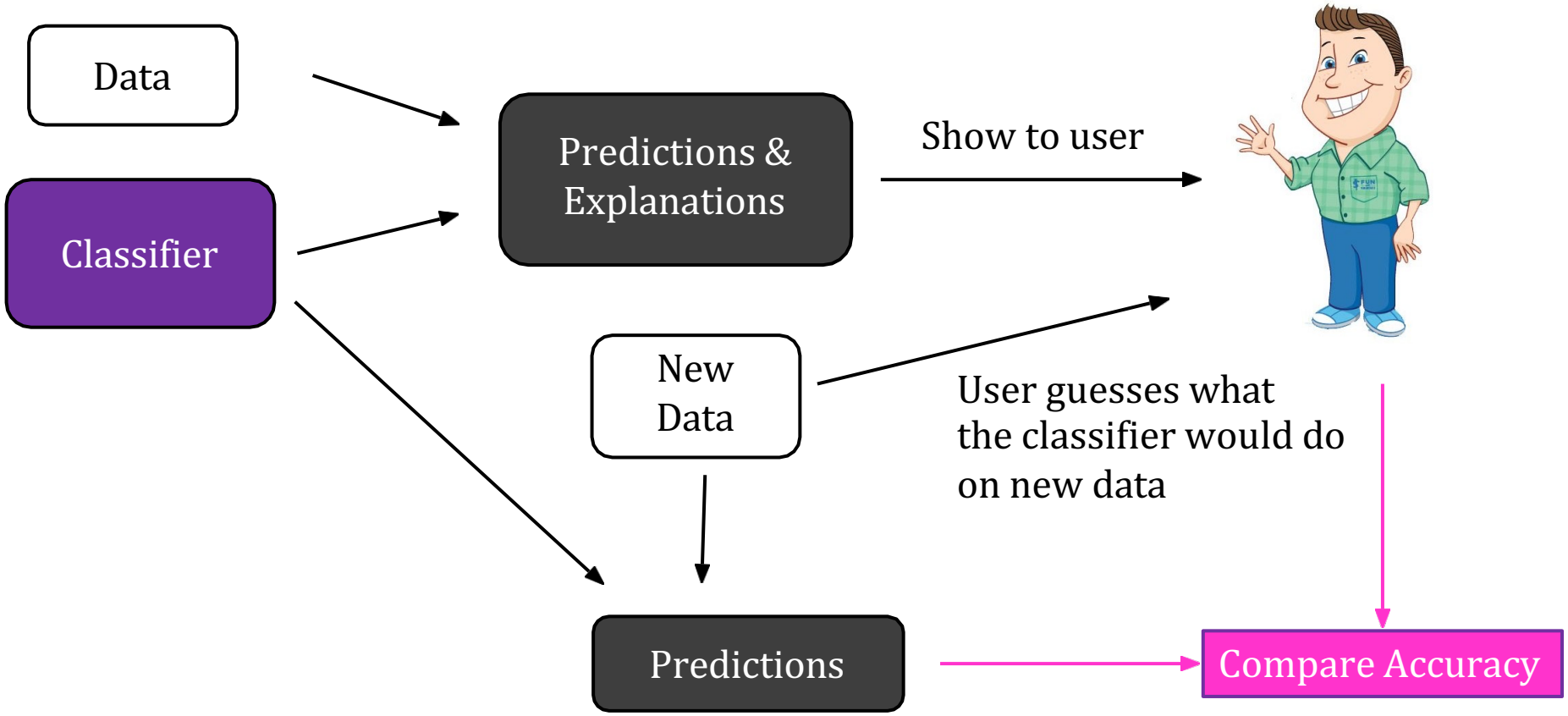
Same Idea: For *Training* Data

Add/remove influential training data, see what happens



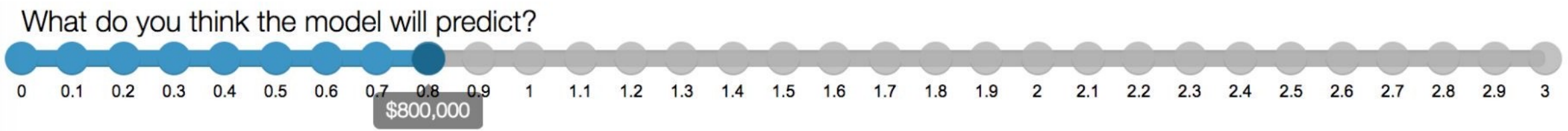
[Ribeiro et al. 2018, Hase and Bansal 2020]

Predicting Behavior ("Simulation")

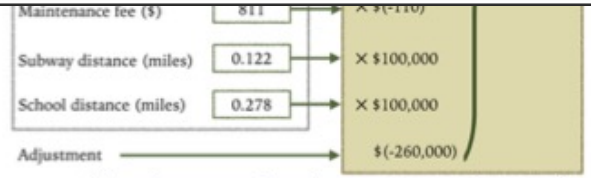


[Poursabzi-Sangdeh et al. 2021]

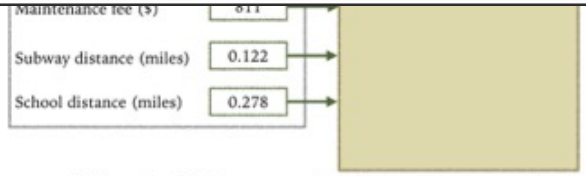
Predicting Behavior ("Simulation")



(a) Step 1: Participants were asked to guess the model's prediction and state their confidence.



(c) Clear, eight-feature condition (CLEAR-8).



(d) Black-box, eight-feature condition (BB-8).

Evaluating Post hoc Explanations

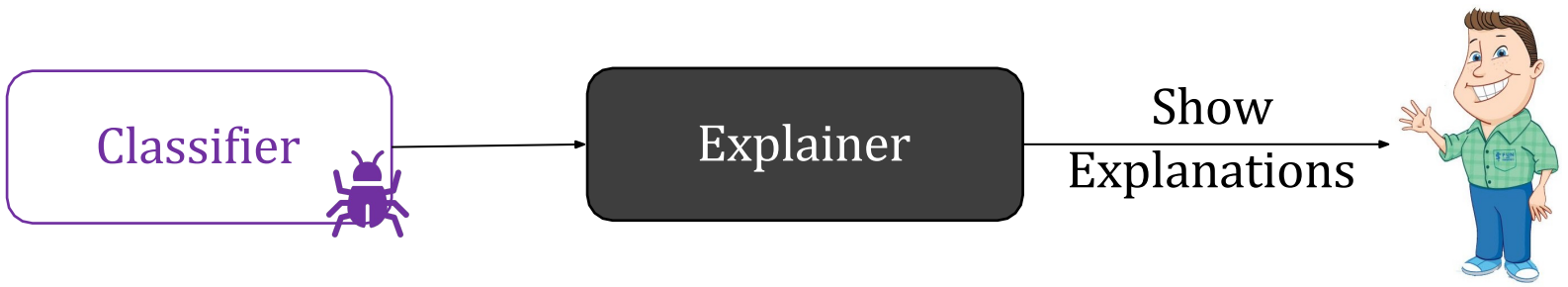
Understand the Behavior

Help make decisions

Useful for Debugging

[Ribeiro et al. 2016]

1. Detecting Problems in Classifiers



Question 1

Would you trust this model?

Did they say no?

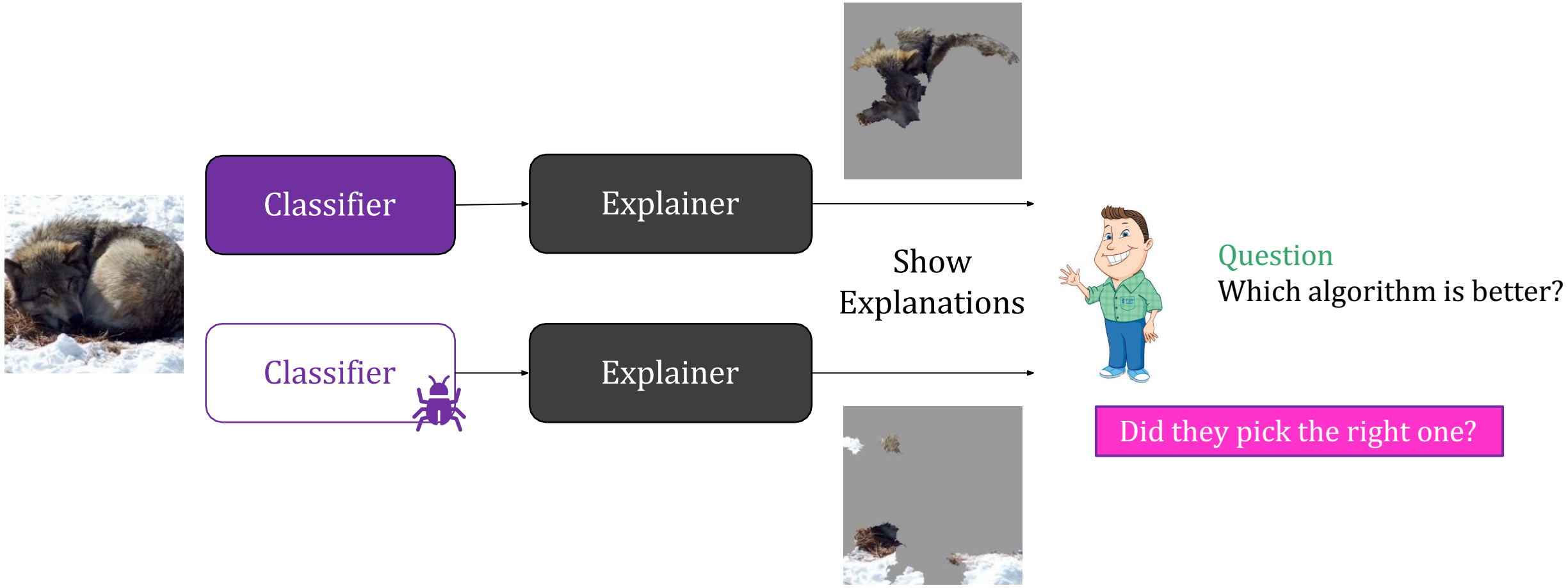
Question 2

What is the classifier doing?

Did they get it right?

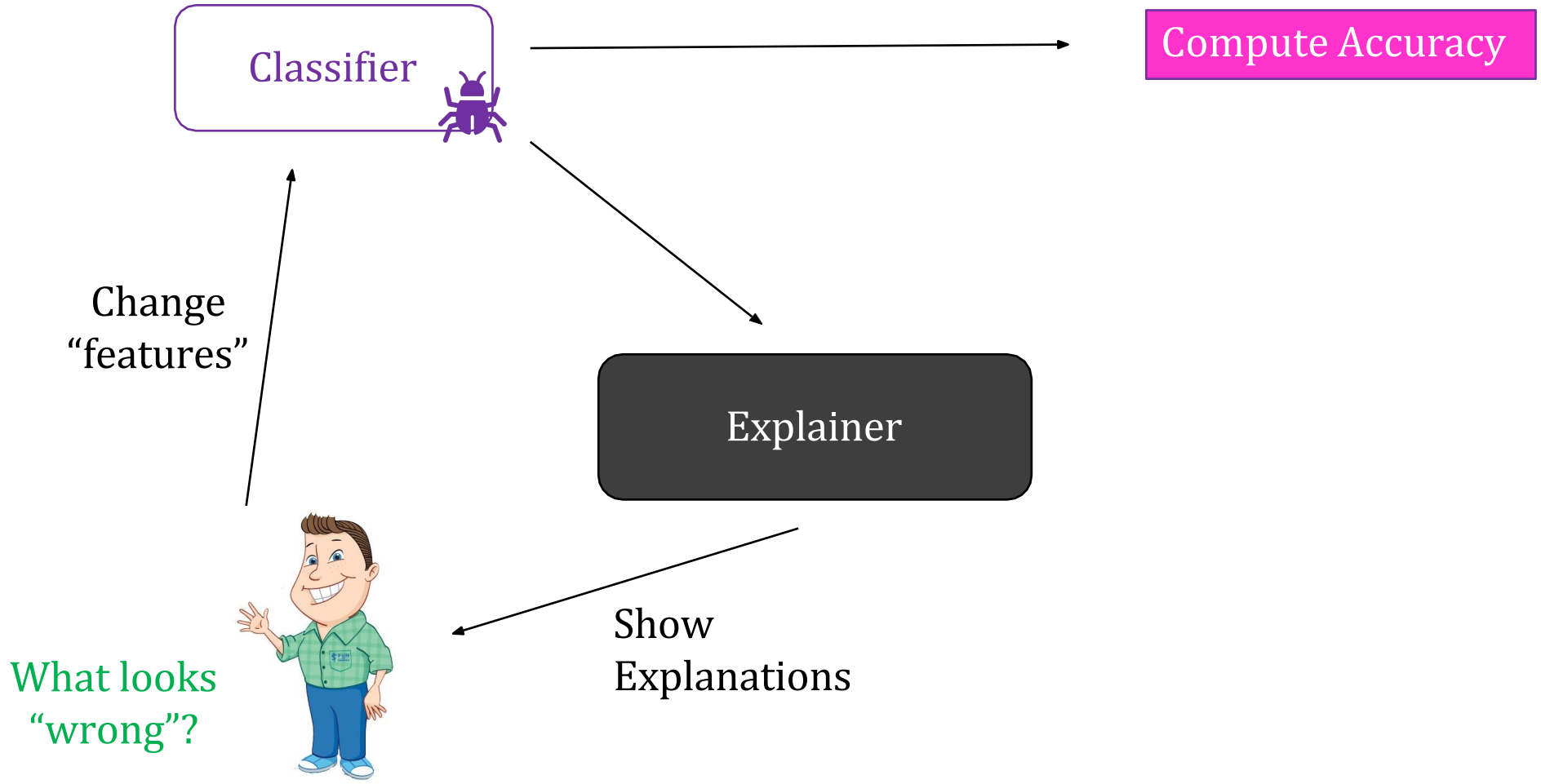
[Ribeiro et al. 2016]

2. Comparing Classifiers



[Ribeiro et al. 2016, Lertvittayakumjorn et al. 2020]

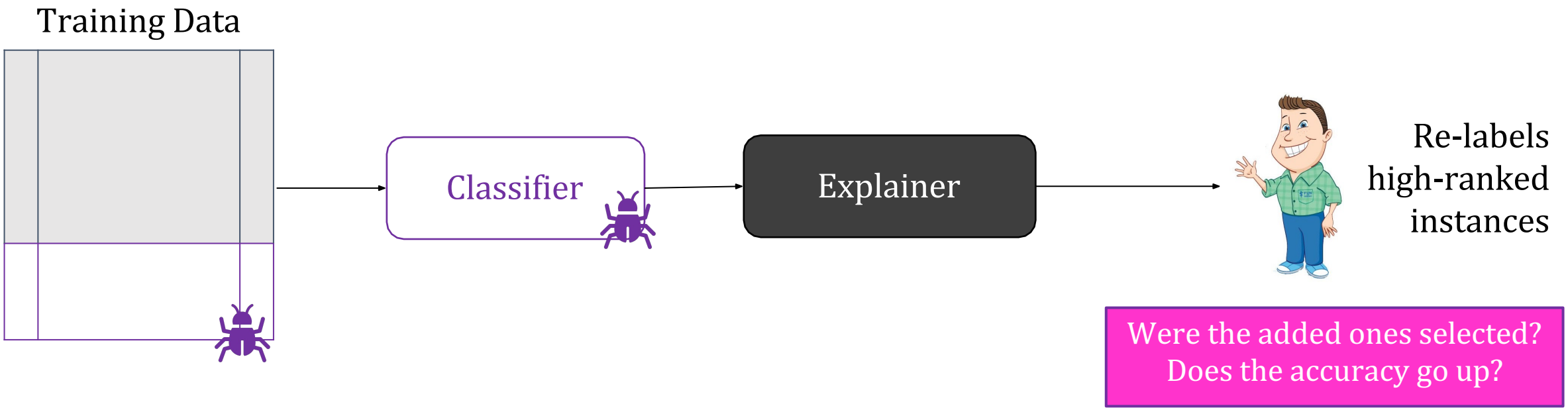
3. “Fixing” Features of Classifiers



[Koh and Liang et al. 2017, Pezeshkpour et al 2019]

4. Finding Errors in Training Data

- Prototypical Explanations: important instances from training data



Evaluating Posthoc Explanations

Understand the Behavior

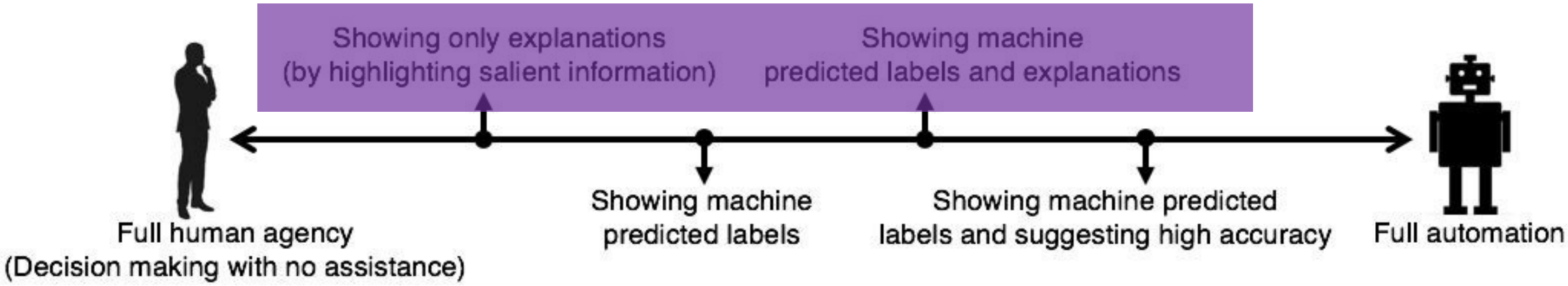
Help make decisions

Useful for Debugging

[Lai and Tan, 2019]

Human-AI Collaboration

- Are Explanations Useful for Making Decisions?
  - For tasks where the algorithms are not reliable by themselves



[Lai and Tan, 2019]

Human-AI Collaboration

- Deception Detection: Identify fake reviews online
  - Are Humans better detectors with explanations?

Note: The highlighted words are important words which machine learning classifiers use to decide if a review is genuine or deceptive. The below scale shows level of importance of each word.

I would not stay at this hotel again. The rooms had a fowl odor. It seemed as though the carpets have never been cleaned. The neighborhood was also less than desirable. The housekeepers seemed to be snooping around while they were cleaning the rooms. I will say that the front desk staff was friendly albeit slightly dimwitted.

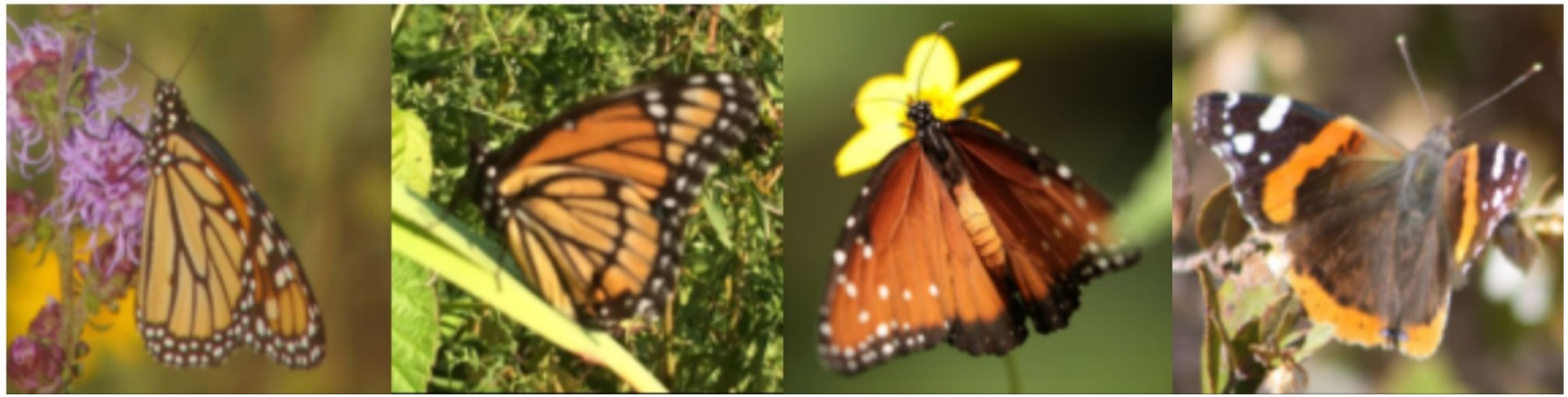
Genuine Deceptive

https://machineintheloop.com/deception/

[Aodha et al, 2018]

Machine Teaching

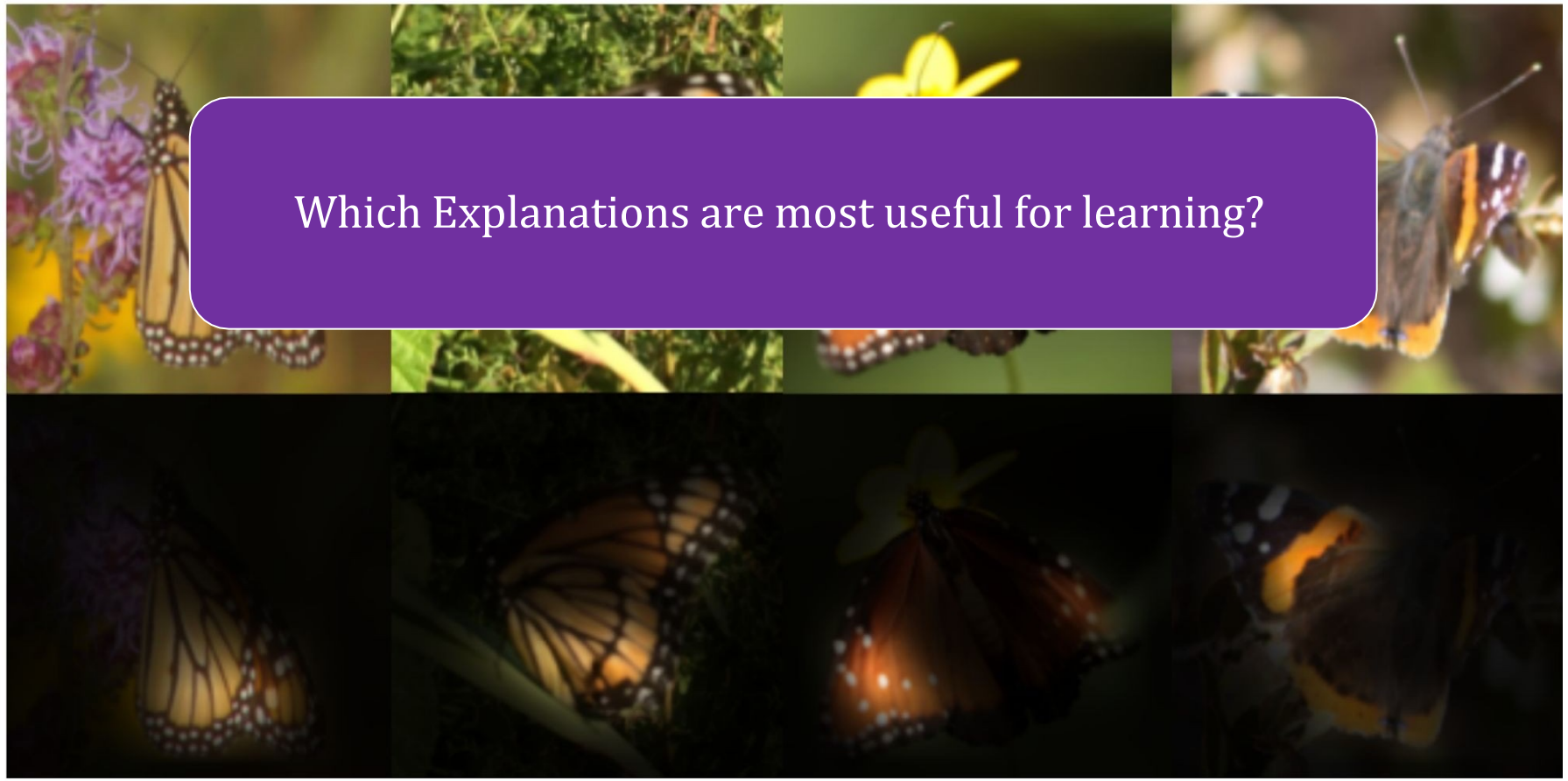
Monarch Viceroy Queen Red Admiral



[Aodha et al, 2018]

Machine Teaching

Monarch Viceroy Queen Red Admiral



Evaluating Posthoc Explanations

Understand the Behavior

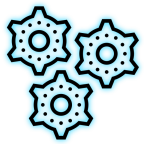
Help make decisions

Useful for Debugging

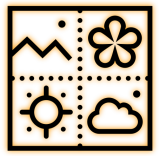
Limitations of Evaluating Explanations

- Evaluation setup is often very easy/simple (or unrealistic)
  - E.g. “bugs” are obvious artifacts, classifiers are different from each other
  - Instances/perturbations create out-of-domain points
- Sometimes flawed
  - E.g. is model explanation same as human explanation?
- Automated metrics can be *optimized*
- User studies are not consistent
  - Affected by choice of: UI, phrasing, visualization, population, incentives, …
  - ML researchers are not trained for this 😢
- Conclusions are difficult to generalize

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



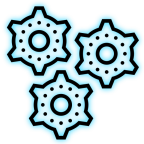
Explanations in Different Modalities

Evaluation of Explanations

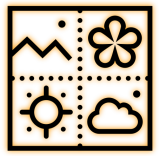
Limits of Post hoc Explainability

Future of Post hoc Explainability

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



Explanations in Different Modalities

Evaluation of Explanations

Limits of Post hoc Explainability

Future of Post hoc Explainability

Limits of Post hoc Explanations

Limitations

- **Faithfulness/Fidelity**
  - Some explanation methods do not '*reflect*' the underlying model.

Limitations

- **Faithfulness/Fidelity**
  - Some explanation methods do not '*reflect*' the underlying model.

- **Fragility**
  - Post-hoc explanations can be easily manipulated.

Limitations

- **Faithfulness/Fidelity**
  - Some explanation methods do not '*reflect*' the underlying model.
- **Fragility**
  - Post-hoc explanations can be easily manipulated.
- **Stability**
  - Slight changes to inputs can cause large changes in explanations.

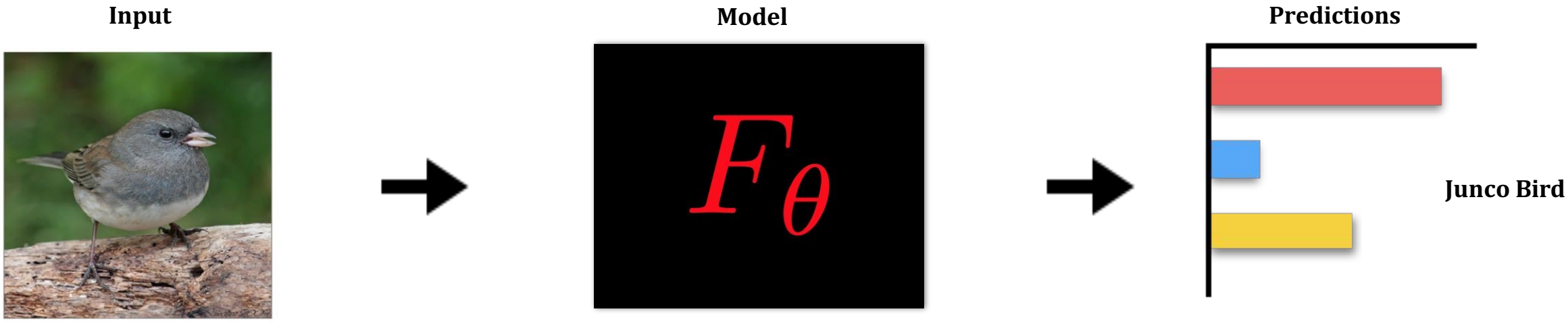
Limitations

- **Faithfulness/Fidelity**
  - Some explanation methods do not '*reflect*' the underlying model.
- **Fragility**
  - Post-hoc explanations can be easily manipulated.
- **Stability**
  - Slight changes to inputs can cause large changes in explanations.
- **Useful in practice?**
  - Unclear if a data scientist (ML engineer)/end-user can use explanations to isolate errors, improve 'trust' or simulate the model.

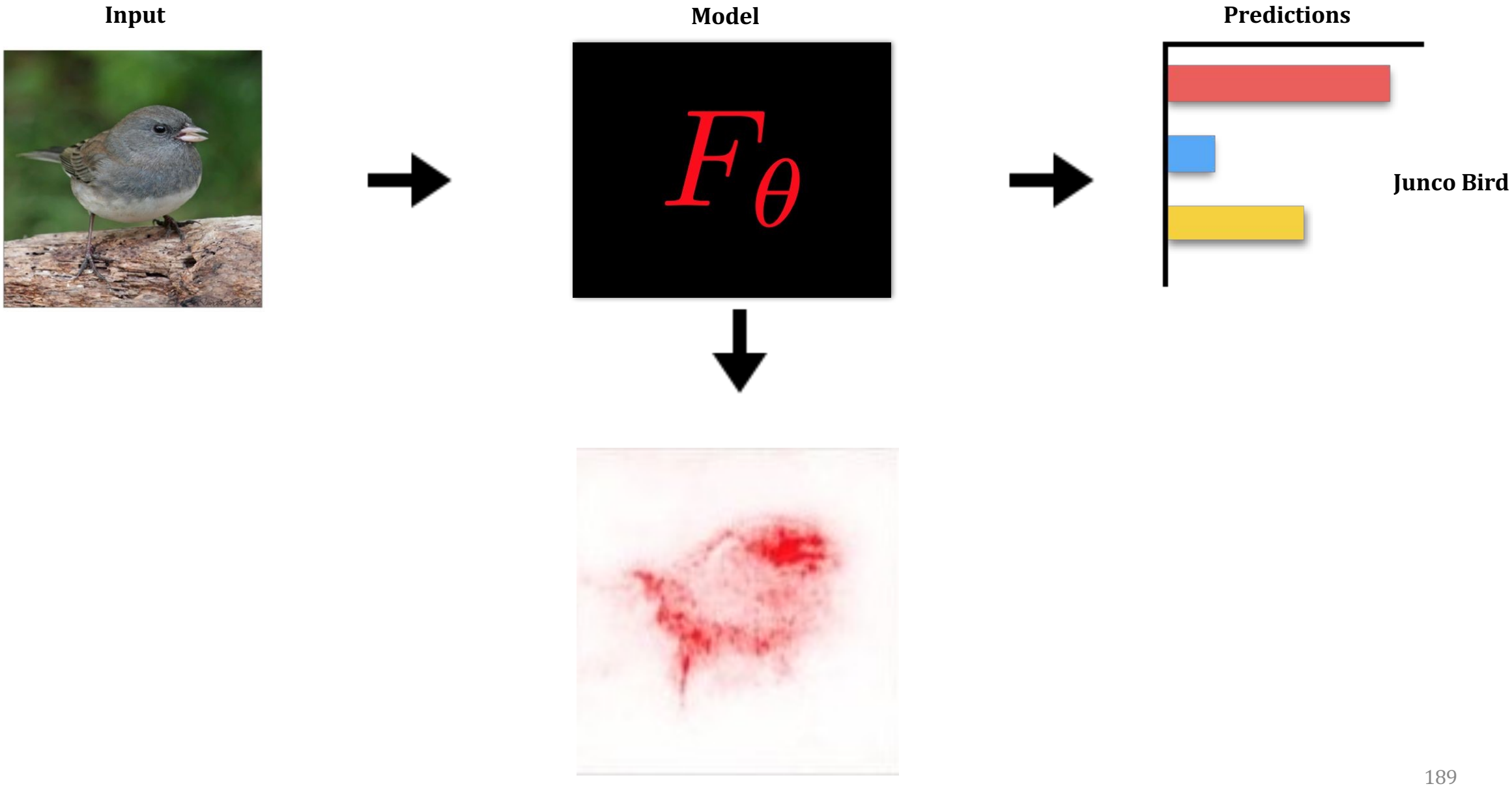
Limitations

- **Faithfulness/Fidelity**
  - Some explanation methods do not '*reflect*' the underlying model.

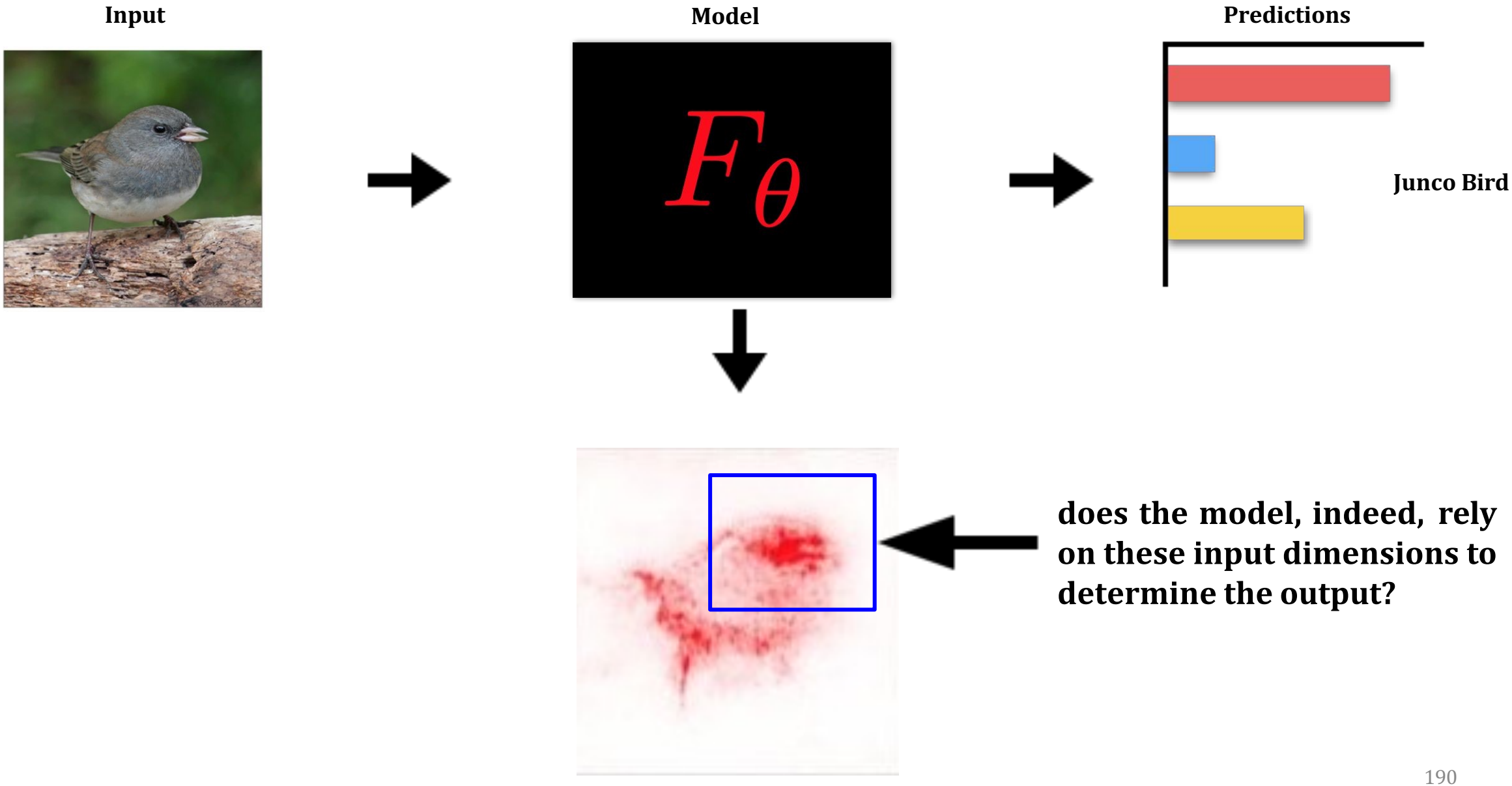
Do Explanations Capture Model-based Discriminative Signals?



Do Explanations Capture Model-based Discriminative Signals?



Do Explanations Capture Model-based Discriminative Signals?



Faithfulness/Fidelity

Does the output of an explanation method reflect the underlying ***'computation or behavior'*** of the black-box model?

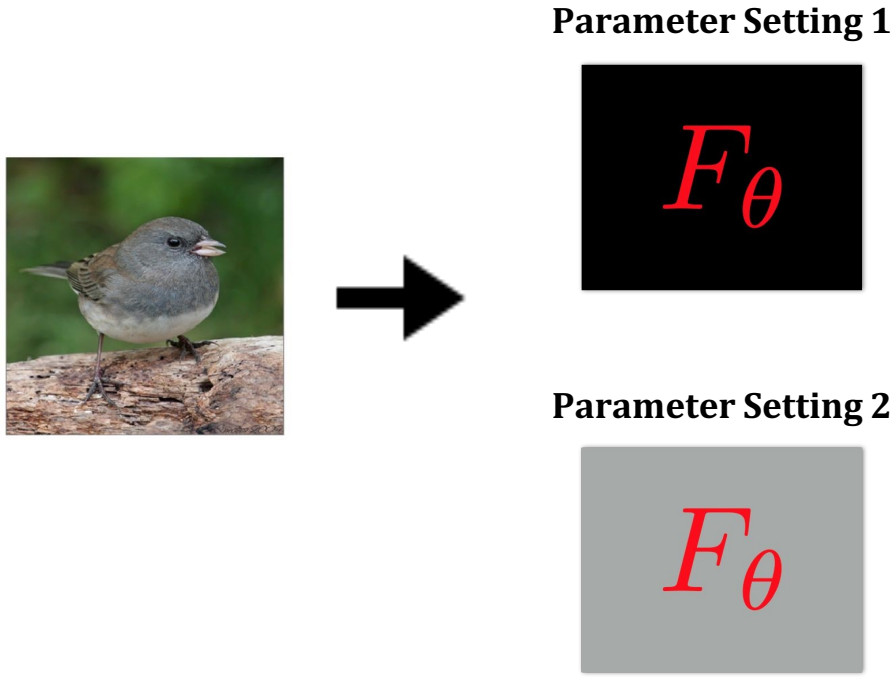
Sanity Check for Faithfulness/Fidelity

- **Sensitivity to Model Parameters:** if the parameter settings change, the explanations should change.

Adebayo et. al. 2018

Sanity Check for Faithfulness/Fidelity

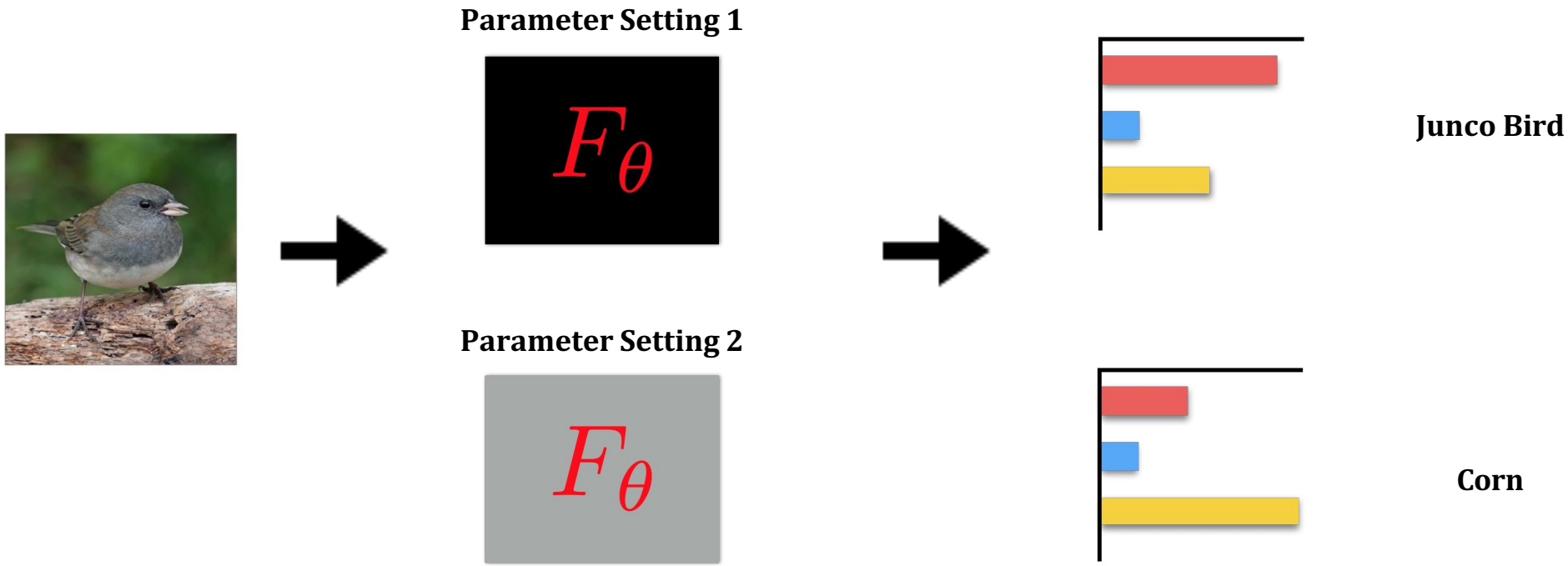
- **Sensitivity to Model Parameters:** if the parameter settings change, the explanations should change.



Adebayo et. al. 2018

Sanity Check for Faithfulness/Fidelity

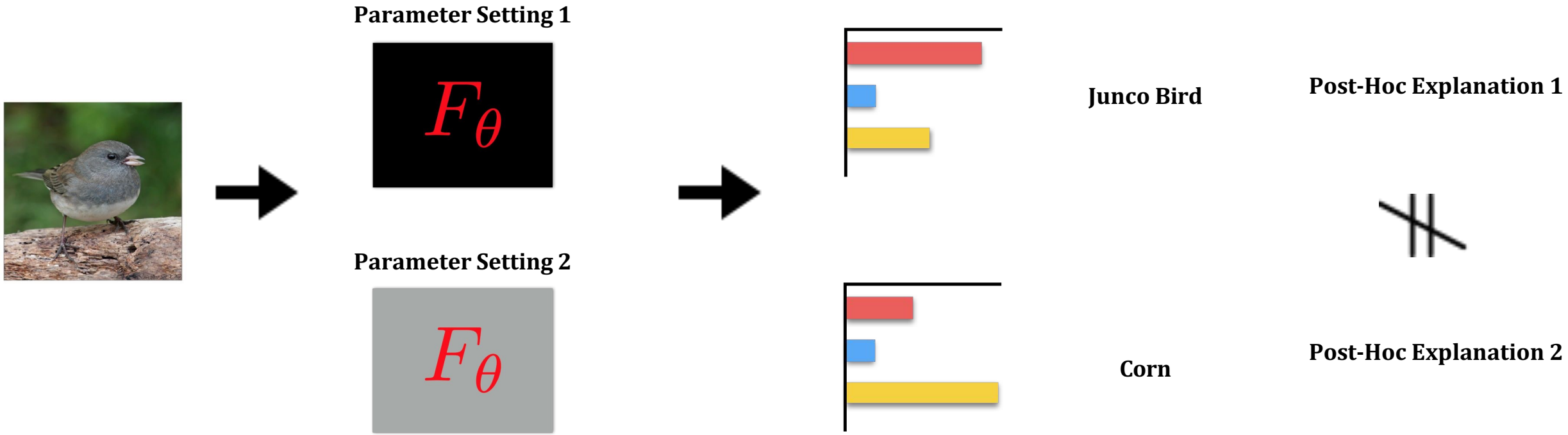
- **Sensitivity to Model Parameters:** if the parameter settings change, the explanations should change.



Adebayo et. al. 2018

Sanity Check for Faithfulness/Fidelity

- **Sensitivity to Model Parameters:** if the parameter settings change, the explanations should change.



Adebayo et. al. 2018

Cascading Randomization Inception-V3

- **Randomize (re-initialize)** model parameters starting from top layer all the way to the input.

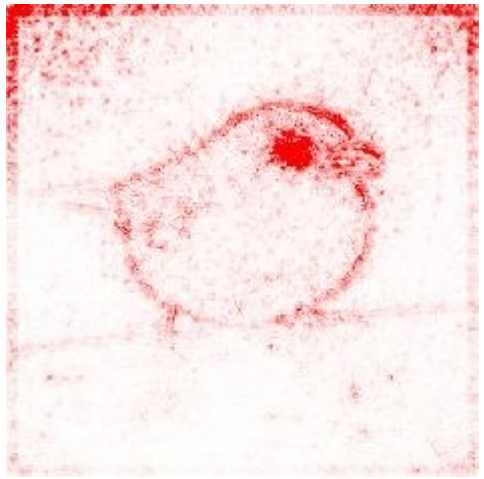
Guided BackProp Explanation Inception-V3 ImageNet

Adebayo et. al. 2018

Cascading Randomization Inception-V3

- **Randomize (re-initialize)** model parameters starting from top layer all the way to the input.

Normal Model Explanation



Guided BackProp Explanation Inception-V3 ImageNet

[Adebayo et. al. 2018]

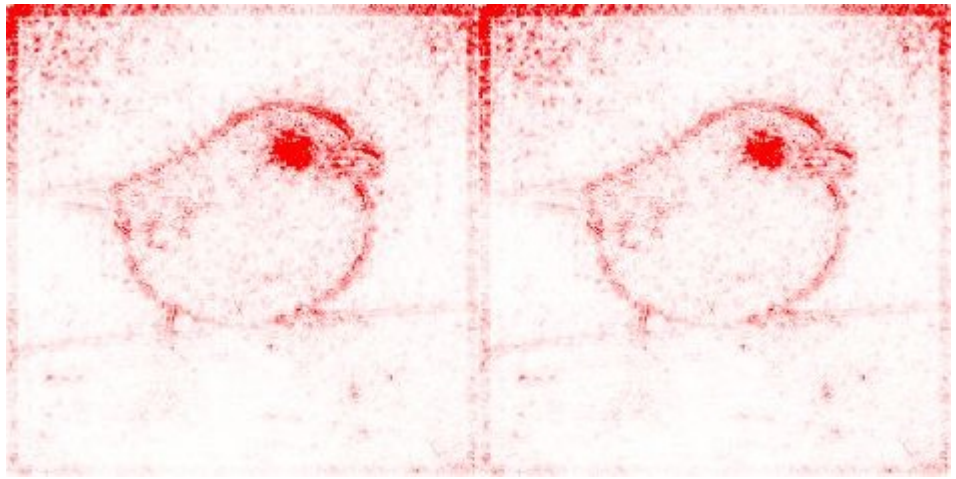
Cascading Randomization Inception-V3

- **Randomize (re-initialize)** model parameters starting from top layer all the way to the input.

Guided BackProp Explanation Inception-V3 ImageNet

Normal Model Explanation

Top Layer Randomized



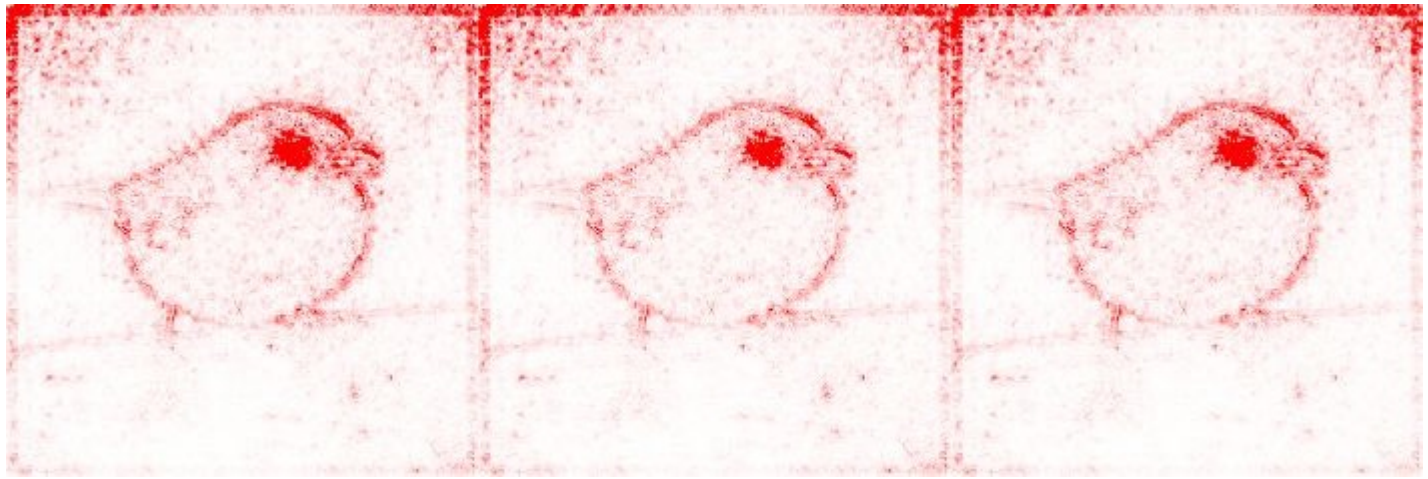
[Adebayo et. al. 2018]

Cascading Randomization Inception-V3

- **Randomize (re-initialize)** model parameters starting from top layer all the way to the input.

Guided BackProp Explanation Inception-V3 ImageNet

Normal Model Explanation | **Top Layer Randomized**



[Adebayo et. al. 2018]

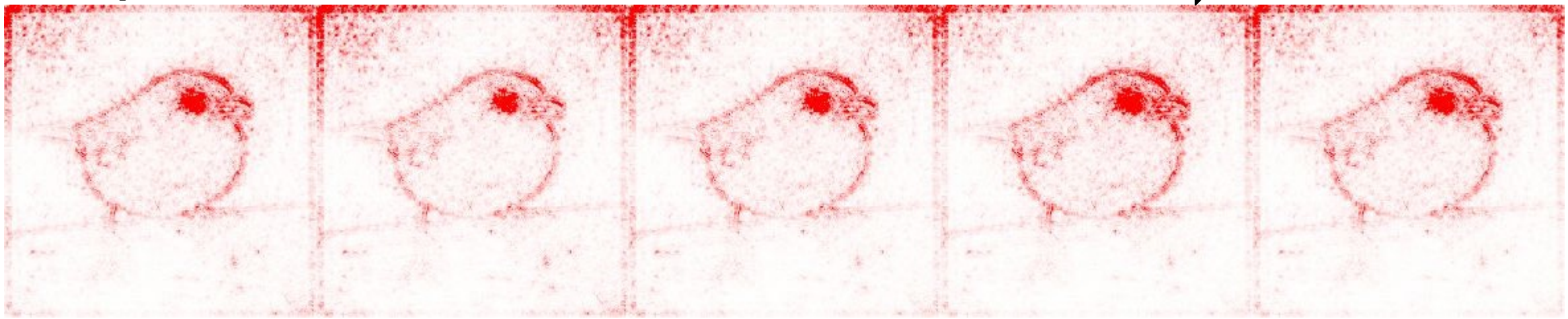
Cascading Randomization Inception-V3

- **Randomize (re-initialize)** model parameters starting from top layer all the way to the input.

Guided BackProp Explanation Inception-V3 ImageNet

Normal Model Explanation

Top Layer Randomized

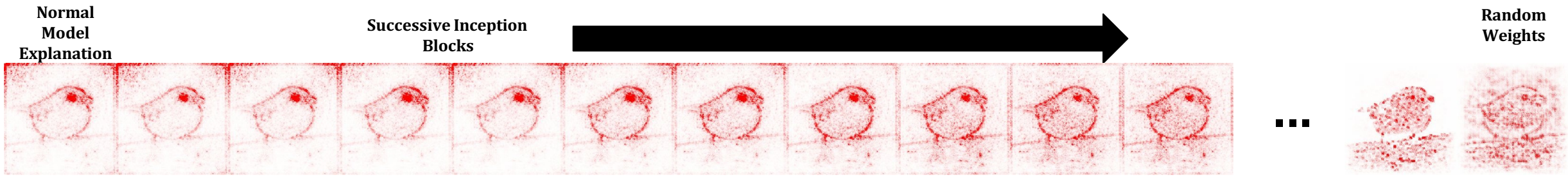


[Adebayo et. al. 2018]

Cascading Randomization Inception-V3

- **Randomize (re-initialize)** model parameters starting from top layer all the way to the input.

Guided BackProp Explanation Inception-V3 ImageNet

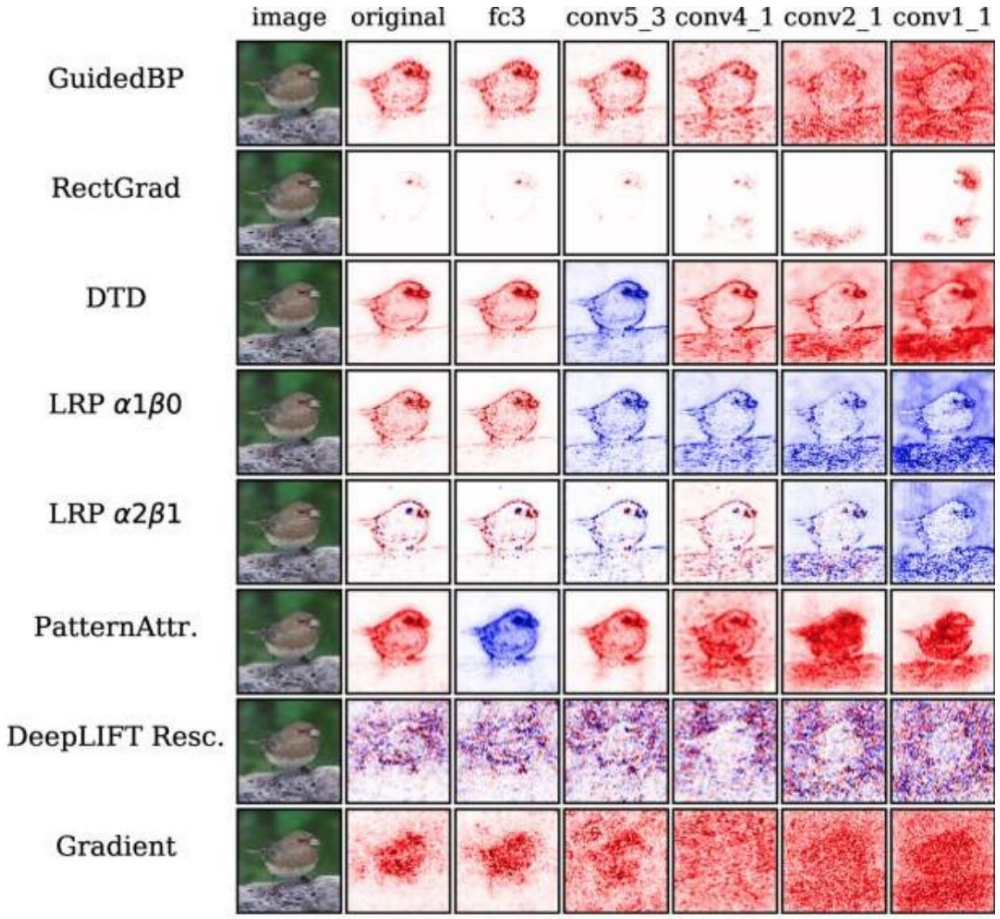


Guided BackProp is invariant to the higher level weights.

Adebayo et. al. 2018

‘Modified backprop approaches’ are invariant

Method that compute relevance via modified backpropagation and performance positive aggregation along the way are invariant to higher layers.



Sixt et. al. 2020

Source of Invariance

- **Guided BackProp and DeConvNet** **seek to approximately reconstruct the input** (Nie et. al. 2018).

- **These modified backprop methods** **converge to a rank-1 matrix**! This is because the product of a sequence of non-negative matrices (non-orthogonal columns, along with other assumptions) converges to a rank-1 matrix (*Theorem 1 in Sixt et. al. 2020*).

Source of Invariance

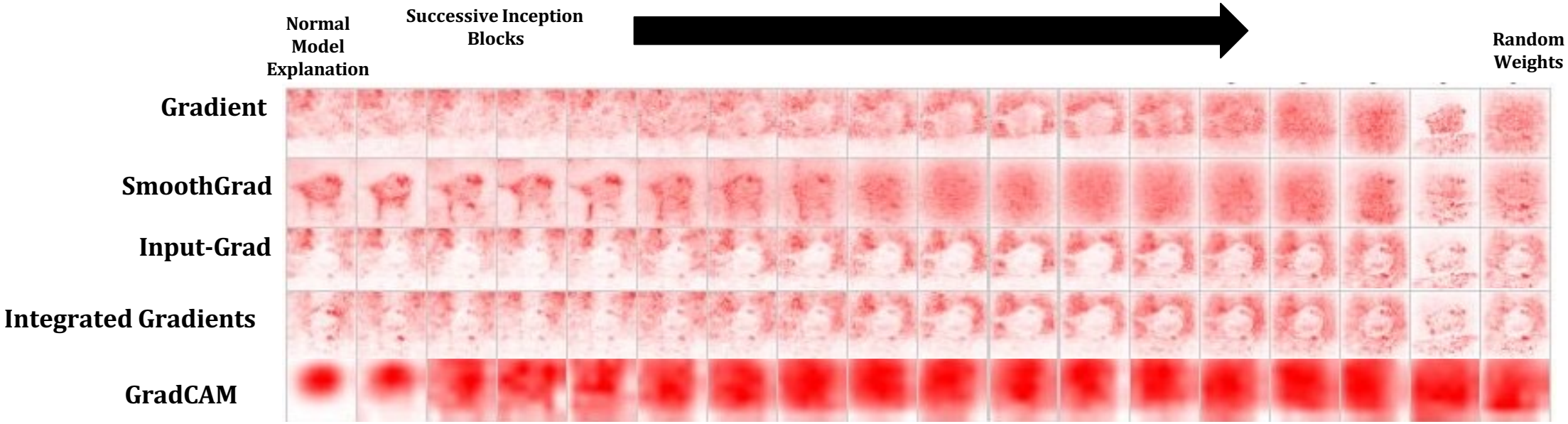
- **Guided BackProp and DeConvNet seek to approximately reconstruct the input** (Nie et. al. 2018).

- **These modified backprop methods converge to a rank-1 matrix**! This is because the product of a sequence of non-negative matrices (non-orthogonal columns, along with other assumptions) converges to a rank-1 matrix (*Theorem 1 in Sixt et. al. 2020*).

- **DeConvNet**
- **Guided BackProp**
- **Guided GradCAM**

- **Deep Taylor Decomposition**
- **Pattern Net and Pattern Attribution (empirically)**
- **RectGrad**

Cascading Randomization Inception-V3



[Adebayo et. al. 2018]

Limitations

- ~~**Faithfulness/Fidelity**~~
  - ~~Some explanation methods do not '*reflect*' the underlying model.~~

- **Fragility**
  - Post-hoc explanations can be easily manipulated.

Post-hoc Explanations are Fragile

Post-hoc explanations can be easily manipulated.

Dombrowski et. al. 2019

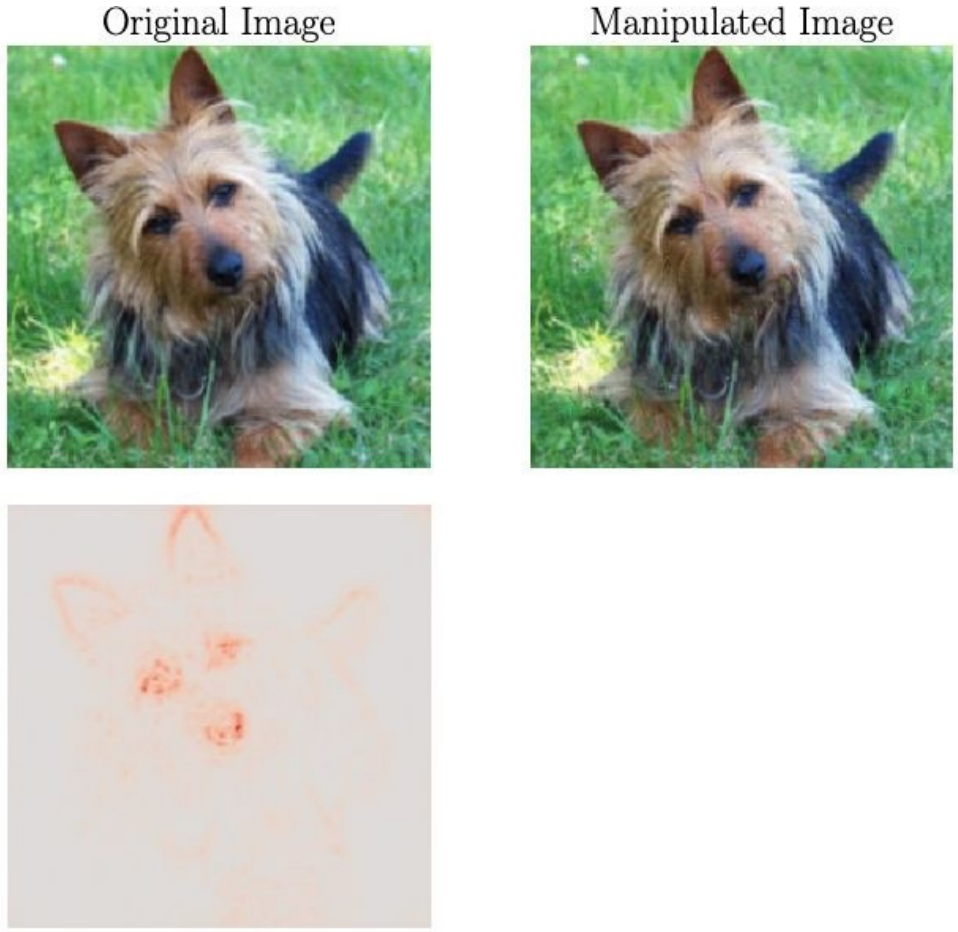
Post-hoc Explanations are Fragile

Post-hoc explanations can be easily manipulated.

Dombrowski et. al. 2019

Post-hoc Explanations are Fragile

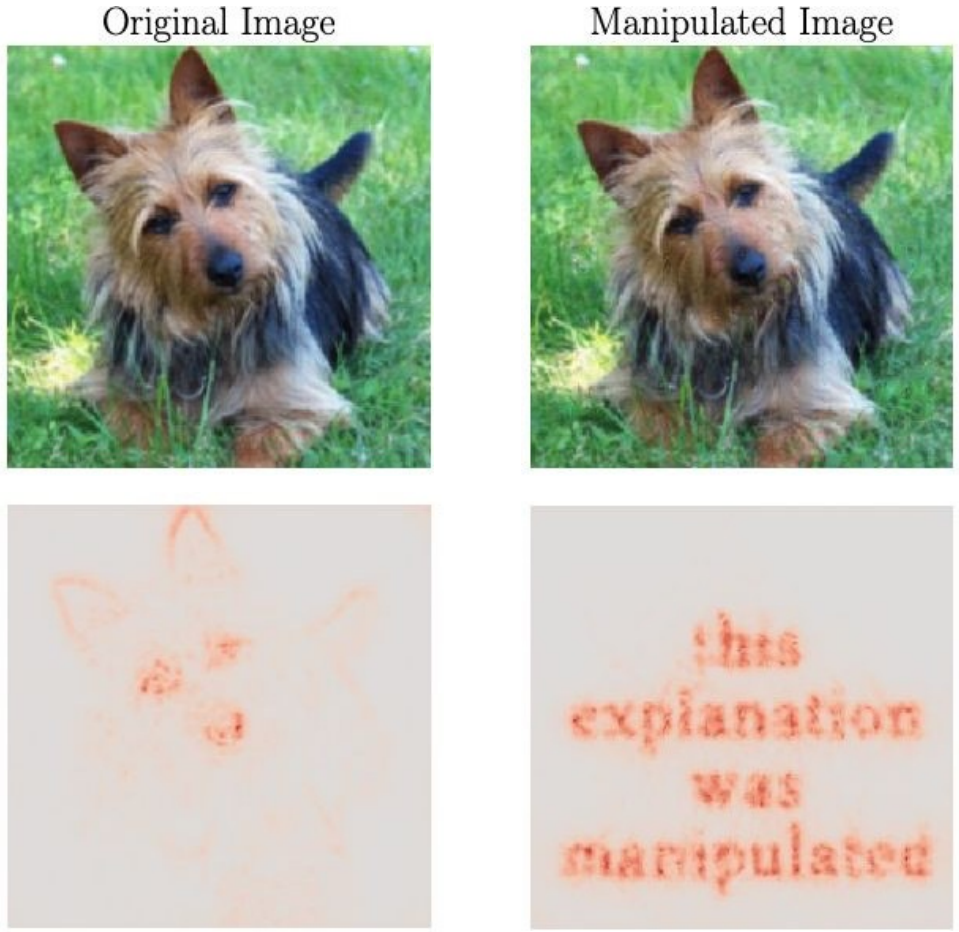
Post-hoc explanations can be easily manipulated.



Dombrowski et. al. 2019

Post-hoc Explanations are Fragile

Post-hoc explanations can be easily manipulated.



Dombrowski et. al. 2019

Adversarial Attack on Explanations

Minimally modify the input with a **small perturbation without changing the model prediction**.

$$\arg\max_{\boldsymbol{\delta}} \mathcal{D}\left(\boldsymbol{I}(\boldsymbol{x}_t; \mathcal{N}), \boldsymbol{I}(\boldsymbol{x}_t + \boldsymbol{\delta}; \mathcal{N})\right)$$

Adversarial Attack of Ghorbani et. al. 2018

Adversarial Attack on Explanations

Minimally modify the input with a **small perturbation without changing the model prediction**.

$$\arg\max_{\boldsymbol{\delta}} \mathcal{D}\left(\boldsymbol{I}(\boldsymbol{x}_t; \mathcal{N}), \boldsymbol{I}(\boldsymbol{x}_t + \boldsymbol{\delta}; \mathcal{N})\right)$$

$$\text{subject to: } ||\boldsymbol{\delta}||_\infty \leq \epsilon,$$

Adversarial Attack of Ghorbani et. al. 2018

Adversarial Attack on Explanations

Minimally modify the input with a **small perturbation without changing the model prediction**.

$$\arg\max_{\boldsymbol{\delta}} \mathcal{D}\left(\boldsymbol{I}(\boldsymbol{x}_t; \mathcal{N}), \boldsymbol{I}(\boldsymbol{x}_t + \boldsymbol{\delta}; \mathcal{N})\right)$$

$$\text{subject to: } ||\boldsymbol{\delta}||_\infty \leq \epsilon,$$

$$\text{Prediction}(\boldsymbol{x}_t + \boldsymbol{\delta}; \mathcal{N}) = \text{Prediction}(\boldsymbol{x}_t; \mathcal{N})$$

Adversarial Attack of Ghorbani et. al. 2018

Other Attacks

- Shift attack by Kindermans & Hooker et. al. (2017).
- Augmented loss function attack by Dombrowski et. al. (2019).
- Passive and Active fooling loss augmentation attack by Heo et. al. (2019).

Other Attacks

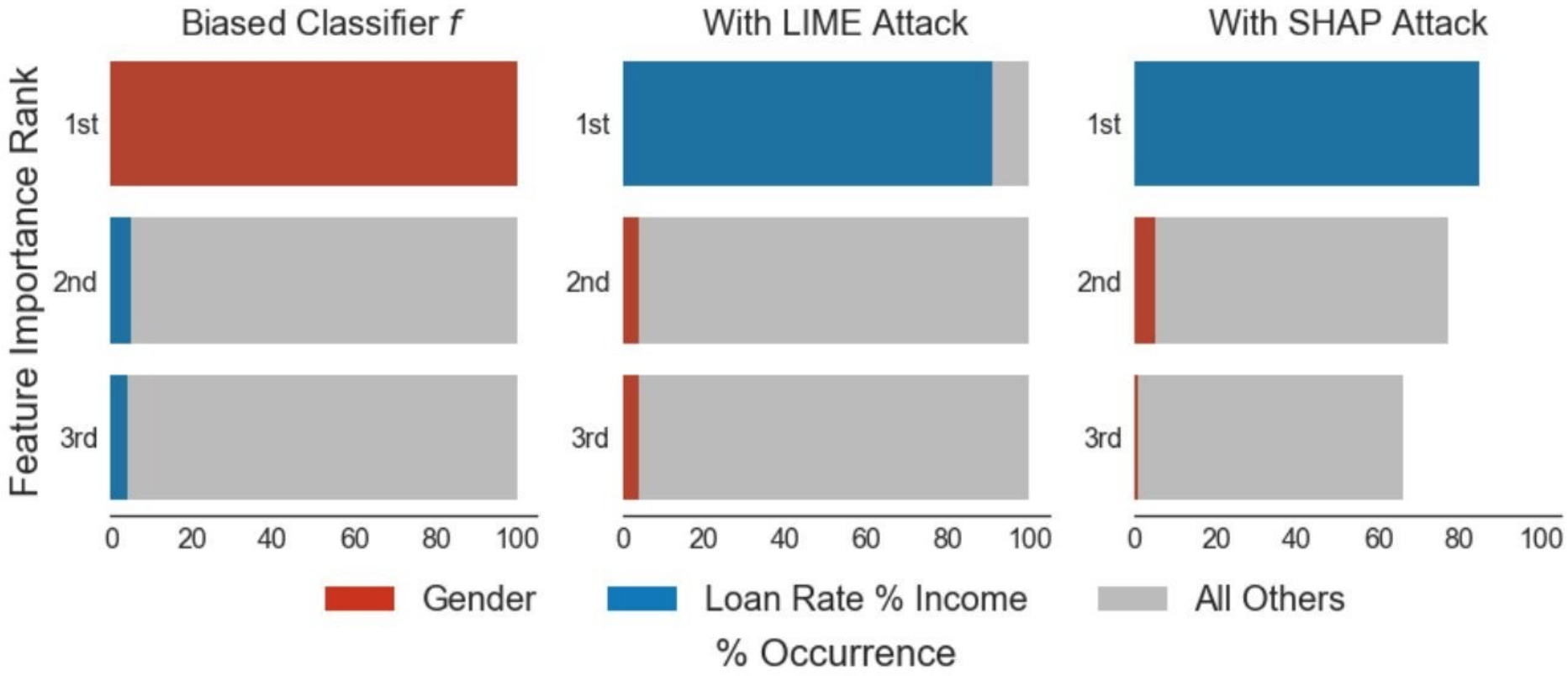
- Shift attack by Kindermans & Hooker et. al. (2017).
- Augmented loss function attack by Dombrowski et. al. (2019).
- Passive and Active fooling loss augmentation attack by Heo et. al. (2019).

Methods Affected

- **LIME**
- **Gradient**
- **Input-Gradient**
- **DeConvNet**
- **Guided BackProp**
- **GradCAM**
- **SHAP**
- **Integrated Gradients**
- **LRP**
- **Deep Taylor Decomposition**
- **Pattern Attribution**
- **Training Point Ranking**

Scaffolding Attack on LIME & SHAP

Scaffolding attack used to **hide classifier dependence on gender**.



[Slack and Hilgard et. al. 2020]

Limitations

- ~~**Faithfulness/Fidelity**~~
  - ~~Some explanations do not reflect the underlying model.~~
- ~~**Fragility**~~
  - ~~Post-hoc explanations can be easily manipulated.~~
- **Stability**
  - Slight changes to inputs can cause large changes in explanations.

Limitations: Stability

Post-hoc explanations can be unstable to small, **non-adversarial**, perturbations to the input.

Alvarez et. al. 2018.

Limitations: Stability

Post-hoc explanations can be unstable to small, **non-adversarial**, perturbations to the input.

'Local Lipschitz Constant'

Explanation function: LIME, SHAP, Gradient...etc.

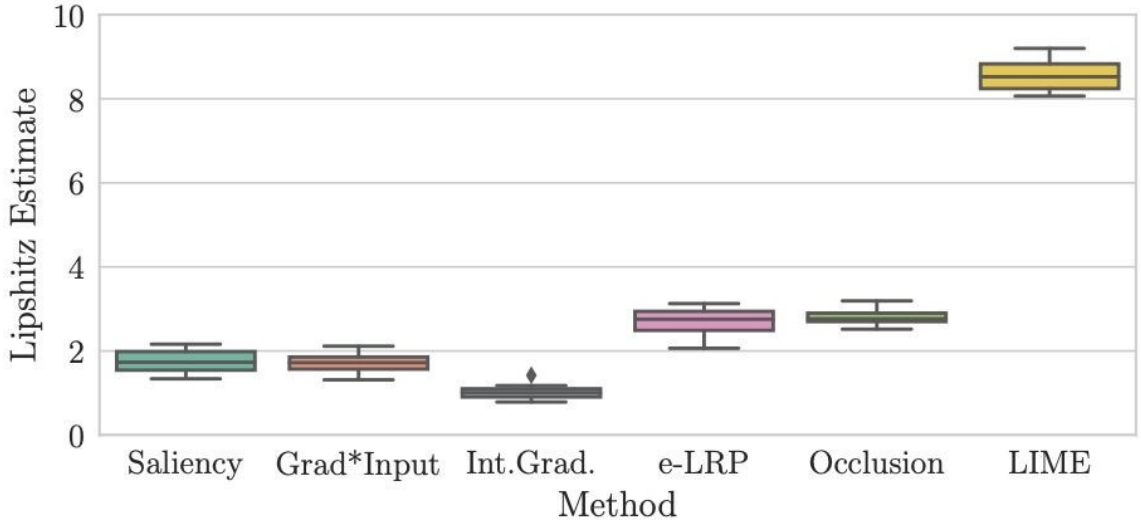
$$\hat{L}(x_i) = \underset{x_j \in B_\epsilon(x_i)}{\operatorname{argmax}} \frac{\|f(x_i) - f(x_j)\|_2}{\|x_i - x_j\|_2}$$

Input

[Alvarez et. al. 2018.]

Limitations: Stability

- Perturbation approaches like LIME can be unstable.
- Yeh et. al. (2019) analytically derive bounds on explanations sensitive for certain popular methods and propose stable variants.

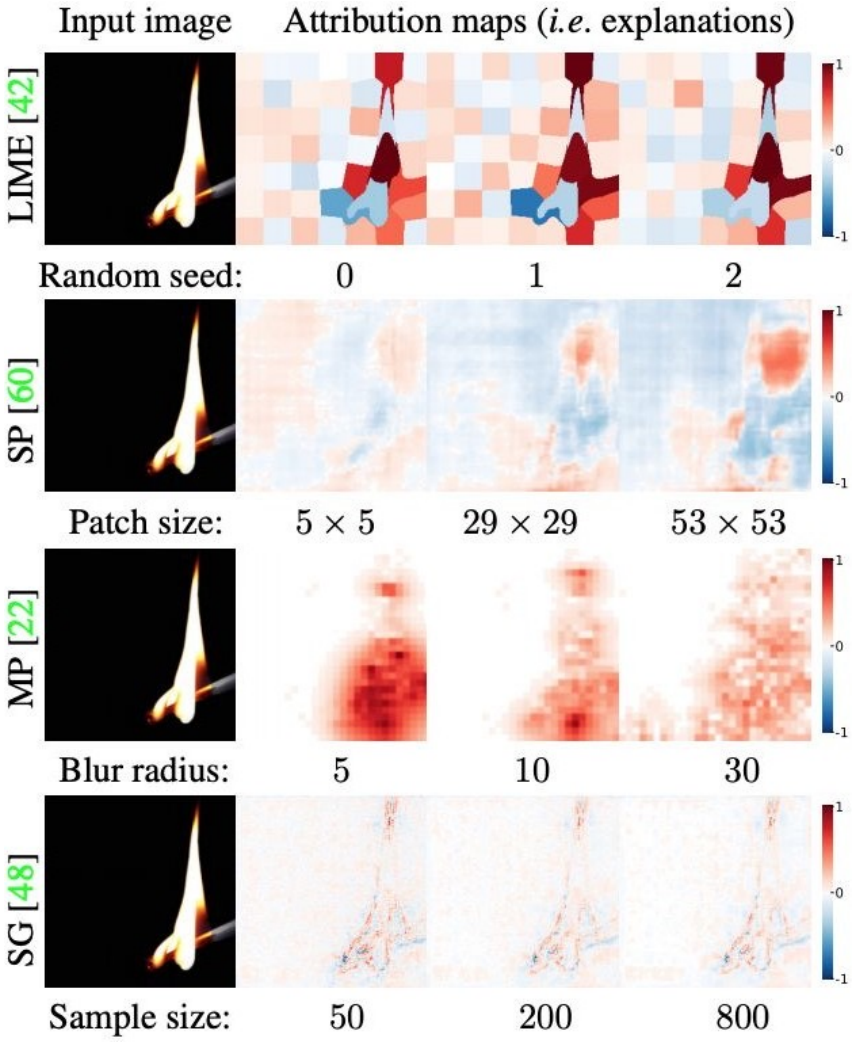


Estimate for 100 tests for an MNIST Model.

Alvarez et. al. 2018.

Sensitivity to Hyperparameters

Explanations can be highly sensitive to hyperparameters such as **random seed**, number of perturbations, patch size, etc.



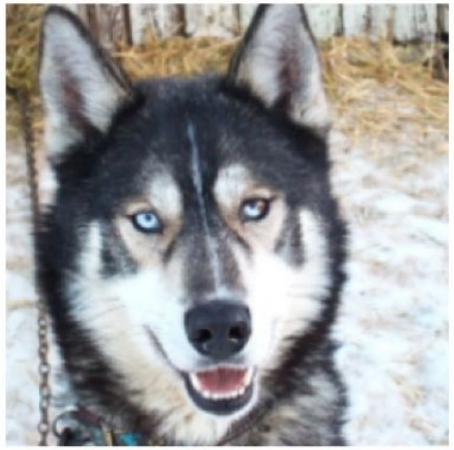
Bansal, Agarwal, & Nguyen, 2020.

Limitations

- ~~**Faithfulness/Fidelity**~~
  - ~~Some explanations do not reflect the underlying model.~~
- ~~**Fragility**~~
  - ~~Post-hoc explanations can be easily manipulated.~~
- ~~**Stability**~~
  - ~~Slight changes to inputs can cause large changes in explanations.~~
- **Useful in practice?**
  - Unclear if a data scientist (ML engineer)/lay person use explanations to isolate errors, improve 'trust', and 'simulatability' in practice?

Model Debugging: Spurious Signals

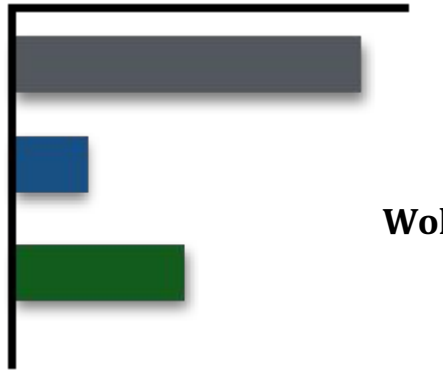
True Label: Siberian Husky



Model

F_θ

Predictions



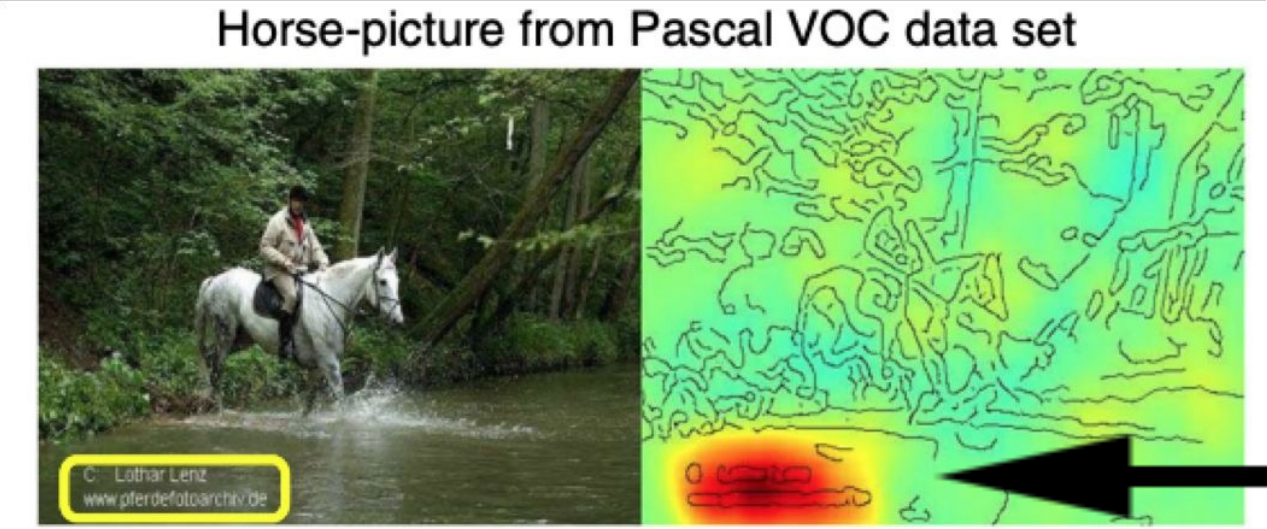
Wolf

LIME

Relying on snow background

[Riberio et. al. 2017.]

Model Debugging: Spurious Signals



Relying on Image Captions to find horses.

[Lapuschkin et. al. 2020]

Explanations with perfect fidelity can still mislead

In a bail adjudication task, **misleading** high-fidelity explanations improve end-user (domain experts) trust.

True Classifier relies on race

If Race ≠ African American:
    If Prior-Felony = Yes and Crime-Status = Active, then Risky
    If Prior-Convictions = 0, then Not Risky

If Race = African American:
    If Pays-rent = No and Gender = Male, then Risky
    If Lives-with-Partner = No and College = No, then Risky
    If Age ≥35 and Has-Kids = Yes, then Not Risky
    If Wages ≥70K, then Not Risky

Default: Not Risky

Lakkaraju & Bastani 2019.

Explanations with perfect fidelity can still mislead

In a bail adjudication task, **misleading** high-fidelity explanations improve end-user (domain experts) trust.

True Classifier relies on race

If Race ≠ African American:
- If Prior-Felony = Yes and Crime-Status = Active, then Risky
- If Prior-Convictions = 0, then Not Risky

If Race = African American:
- If Pays-rent = No and Gender = Male, then Risky
- If Lives-with-Partner = No and College = No, then Risky
- If Age ≥35 and Has-Kids = Yes, then Not Risky
- If Wages ≥70K, then Not Risky

Default: Not Risky

High fidelity 'misleading' explanation

If Current-Offense = Felony:
- If Prior-FTA = Yes and Prior-Arrests ≥ 1, then Risky
- If Crime-Status = Active and Owns-House = No and Has-Kids = No, then Risky
- If Prior-Convictions = 0 and College = Yes and Owns-House = Yes, then Not Risky

If Current-Offense = Misdemeanor and Prior-Arrests > 1:
- If Prior-Jail-Incarcerations = Yes, then Risky
- If Has-Kids = Yes and Married = Yes and Owns-House = Yes, then Not Risky
- If Lives-with-Partner = Yes and College = Yes and Pays-Rent = Yes, then Not Risky

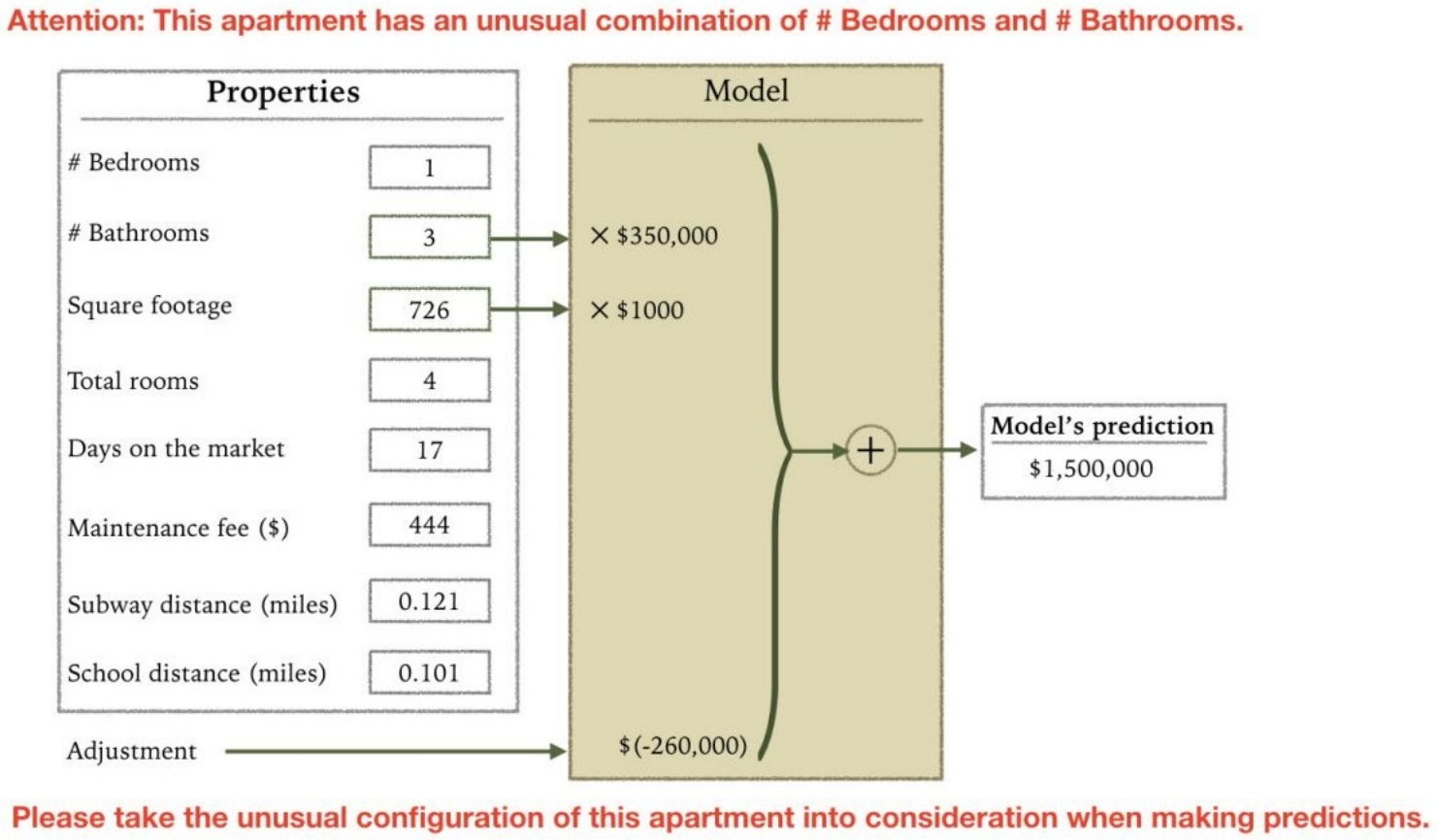
If Current-Offense = Misdemeanor and Prior-Arrests ≤ 1:
- If Has-Kids = No and Owns-House = No and Prior-Jail-Incarcerations = Yes, then Risky
- If Age ≥ 50 and Has-Kids = Yes and Prior-FTA = No, then Not Risky

Default: Not Risky

Lakkaraju & Bastani 2019.

Difficulty using explanations for debugging

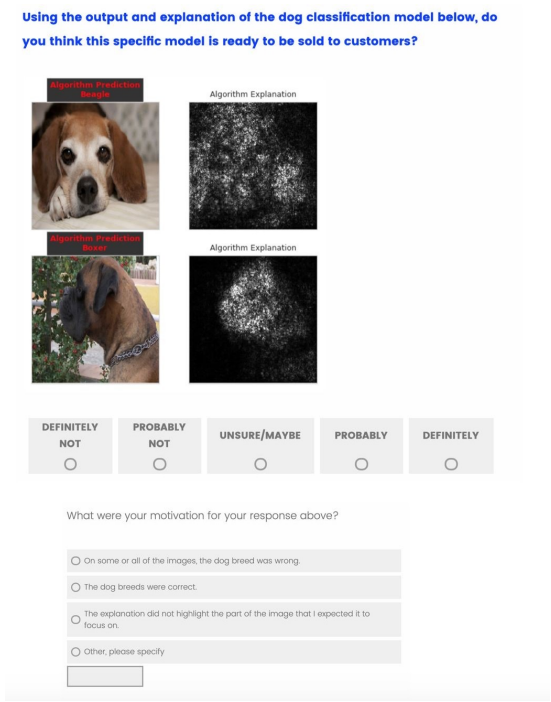
In a housing price prediction task, Amazon mechanical turkers are unable to use linear model coefficients to diagnose model mistakes.



[Poursabzi-Sangdeh et. al. 2019]

Difficulty using explanations for debugging

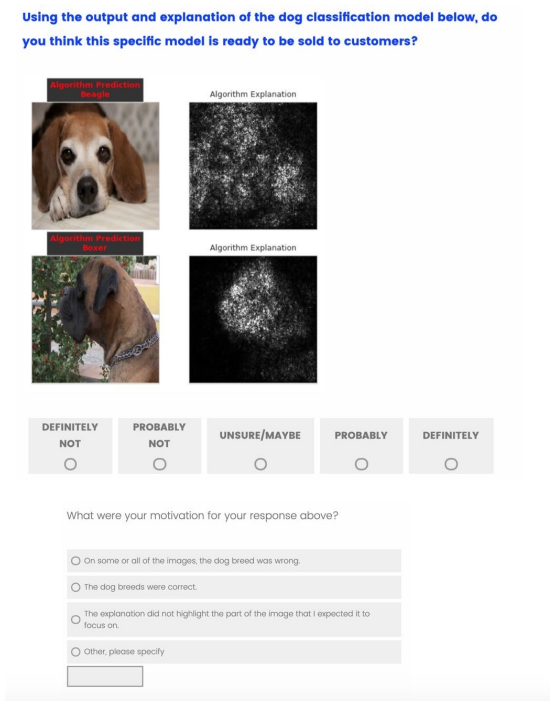
In a dog breeds classification task, users familiar with machine learning **rely on labels, instead of saliency maps,** for diagnosing model errors.

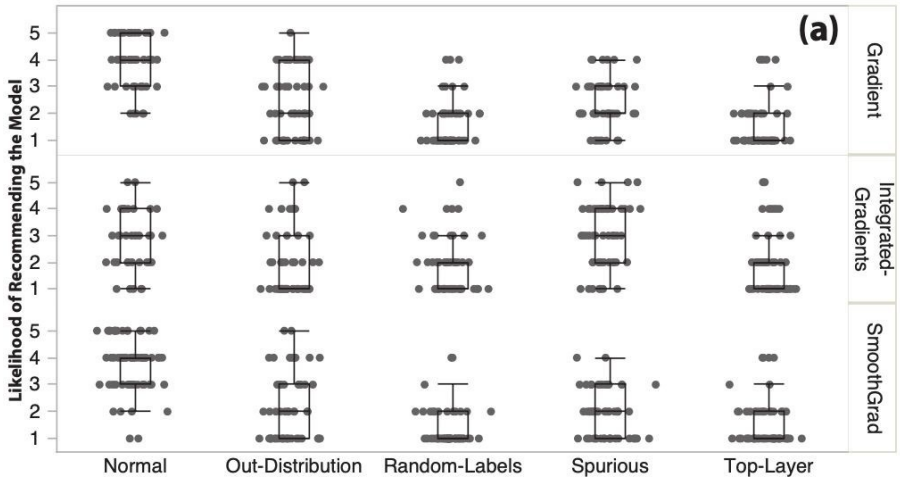


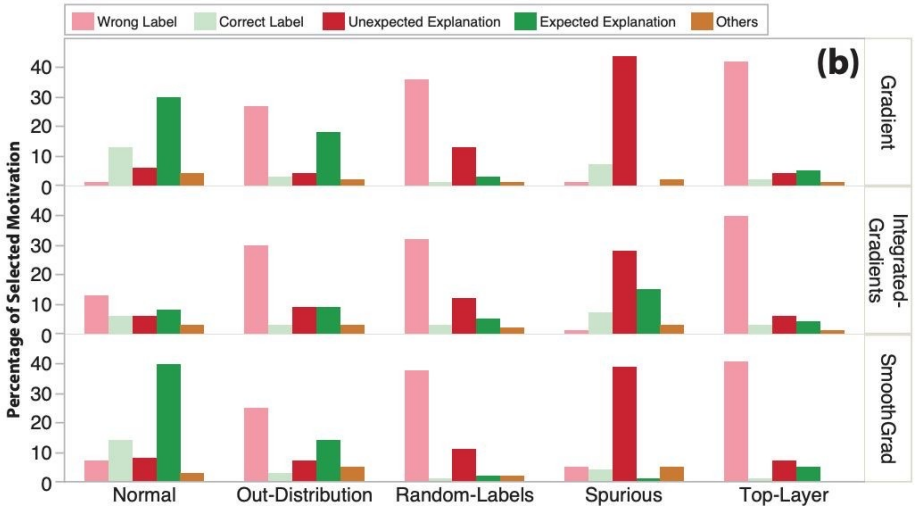
[Adebayo et. al., 2020.]

Difficulty using explanations for debugging

In a dog breeds classification task, users familiar with machine learning **rely on labels, instead of saliency maps,** for diagnosing model errors.



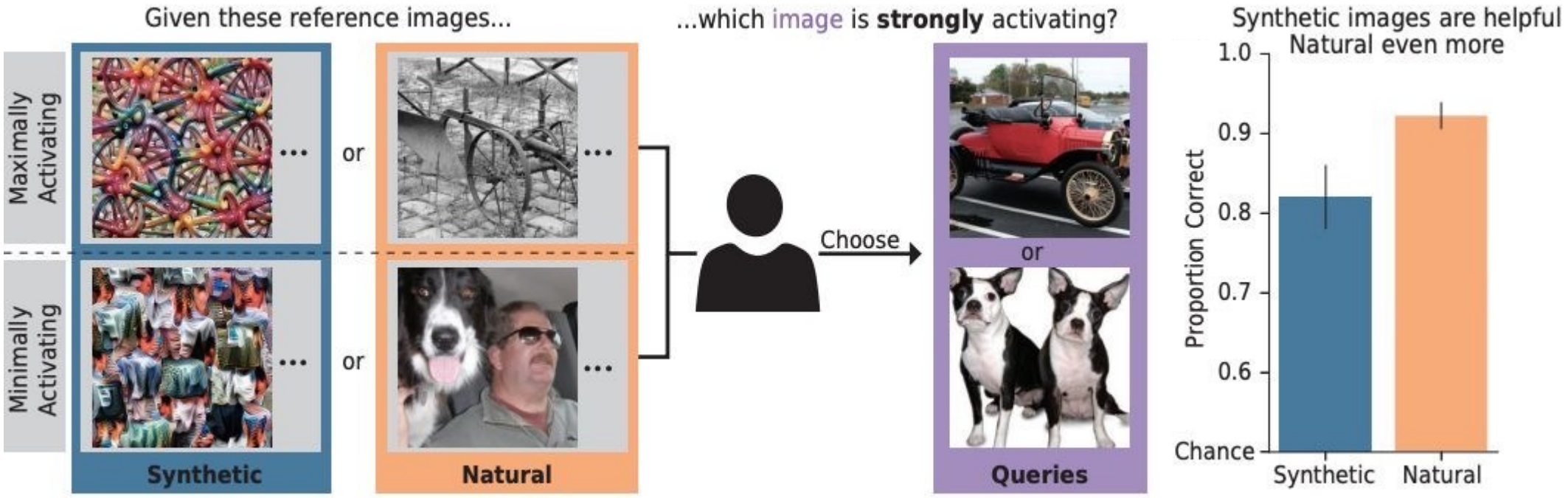




[Adebayo et. al., 2020.]

Natural images more helpful than feature visualization

Users found natural images more helpful than feature visualization in deciding whether in image strongly activated a neuron.



[Borowski & Zimmermann et. al. 2020]

Conflicting Evidence on Utility of Explanations

- **Mixed evidence**:
  - simulation and benchmark studies show that explanations are useful for debugging;
  - however, recent user studies show limited utility in practice.

Conflicting Evidence on Utility of Explanations

- **Mixed evidence**:
  - simulation and benchmark studies show that explanations are useful for debugging;
  - however, recent user studies show limited utility in practice.

- Rigorous **user studies** and **pilots with end-users** can continue to help provide feedback to researchers on what to address (see: Alqaraawi et. al. 2020, Bhatt et. al. 2020 & Kaur et. al. 2020).

[Kaur et. al., 2020]

Responses from Data Scientists Using Explainability Tools (GAM and SHAP)

"I didn't fully grasp what SHAP values were. This is a pretty popular tool and I get the log-odds concept in general. I figure they were showing SHAP values for a reason. Maybe it's easier to judge relationships using log-odds instead of predicted value. Anyway, so it made sense I suppose." (P6, SHAP)

"Age 38 seems to have the highest positive influence on income based on the plot. Not sure why, but the explanation clearly shows it... makes sense." (P9, GAMs)

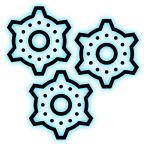
"[The tool] assigns a value that is important to know, but it's showing that in a way that makes you misinterpret that value. Now I want to go back and check all my answers"... [later] "Okay, so, it's not showing me a whole lot more than what I can infer on my own. Now I'm thinking... is this an 'interpretability tool'?" (P4, SHAP)

"[The tool] shows visualizations of ML models, which is not something anything else I have worked with has done. It's very transparent, and that makes me trust it more" (P9, GAMs).

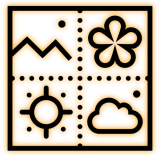
Limitations

- **Faithfulness/Fidelity**
  - Some explanation methods do not '*reflect*' the underlying model.
- **Fragility**
  - Post-hoc explanations can be easily manipulated.
- **Stability**
  - Slight changes to inputs can cause large changes in explanations.
- **Useful in practice?**
  - Unclear if a data scientist (ML engineer)/end-user can use explanations to isolate errors, improve 'trust' or simulate the model.

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



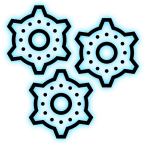
Explanations in Different Modalities

Evaluation of Explanations

Limits of Post hoc Explainability

Future of Post hoc Explainability

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability

Explanations in Different Modalities

Evaluation of Explanations

Limits of Post hoc Explainability

Future of Post hoc Explainability

Future of Post hoc Explainability

Emerging Topics in Explainability Research

Future of Post hoc Explainability

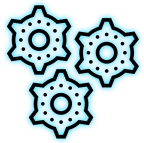
Towards Better Post hoc Explanations

- Methods for More Reliable Post hoc Explanations
- Theoretical Analysis of Post hoc Explanation Methods
- Rigorous Evaluation of the Utility of Post hoc Explanations

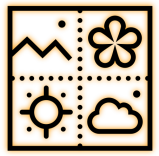
Other Emerging Directions

- Post hoc Explainability Beyond Classification
- Intersections with Differential Privacy
- Intersections with Fairness

Tutorial on Post hoc Explanations



Approaches for Post hoc Explainability



Explanations in Different Modalities

Evaluation of Explanations

Limits of Post hoc Explainability

Future of Post hoc Explainability

Parting Thoughts…

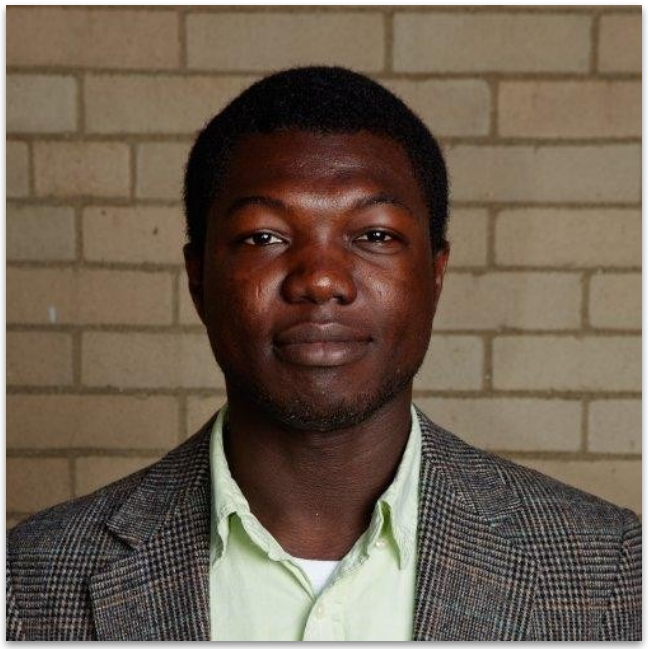
[Leavitt and Morcos, 2020; Roth and Kearns, 2019]

When introducing a new explanation method:

- Who are the target end users that the method will help?
- A clear statement about what capability and/or insight the method aims to provide to its end users
- Careful analysis and exposition of the limitations and vulnerabilities of the proposed method
- Rigorous user studies (preferably with actual end users) to evaluate if the method is achieving the desired effect
- Use quantitative metrics (and not anecdotal evidence) to make claims about explainability

Thank You!

Hima Lakkaraju
Harvard University

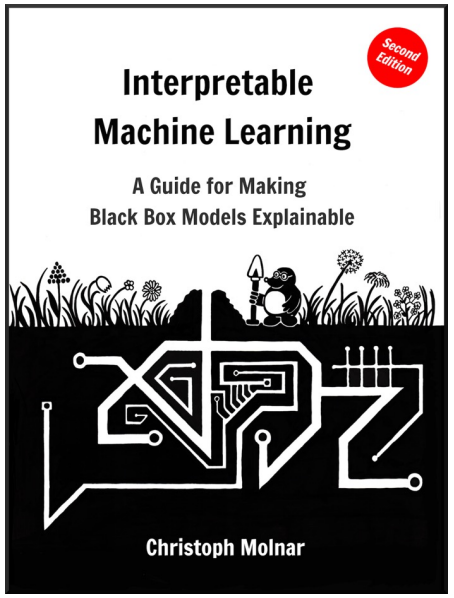


Julius Adebayo
MIT

Sameer Singh
UC Irvine

Slides and Video: explainml-tutorial.github.io

Other interesting references



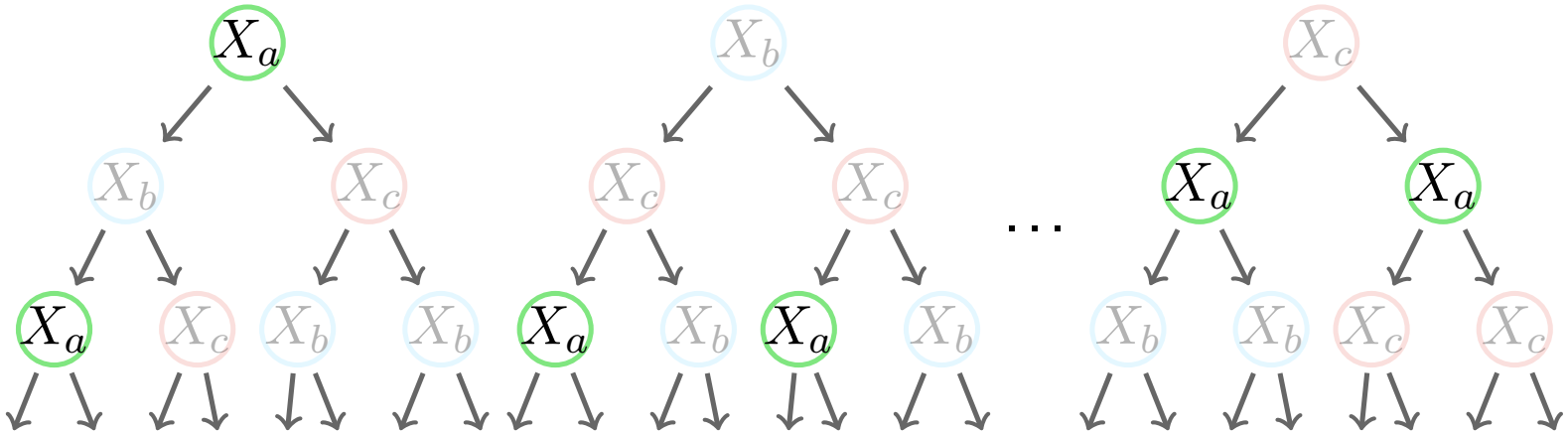
- Interpretable Machine Learning. Christoph Molnar. https://christophm.github.io/interpretable-ml-book/
- Definitions, methods, and applications in interpretable machine learning. Murdoch, Singh, Kumbier, Yu. PNAS 116 (44) 22071-22080, 2019.
- Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Rudin. Nature Machine Intelligence, 1, 206-215 (2019).
- Interpretable machine learning: Fundamental principles and 10 grand challenges. Rudin et al. Statistics Surveys. 16, 1-85, 2022.

Some related works at ULiège

- *From global to local MDI variable importances for random forests and when they are Shapley values*. Sutera, Huynh-Thu, Louppe, Wehenkel and Geurts, NeurIPS 2021.
  - A theoretical analysis and an extension (from global to local) of an existing feature importance measures for random forest.

- *Optimizing Model-Agnostic Random Subspace Ensembles*. Huynh-Thu and Geurts, Submitted, 2022.
  - A model-agnostic method for training an ensemble and selecting important features using gradient-descent, even when base models are non differentiable.

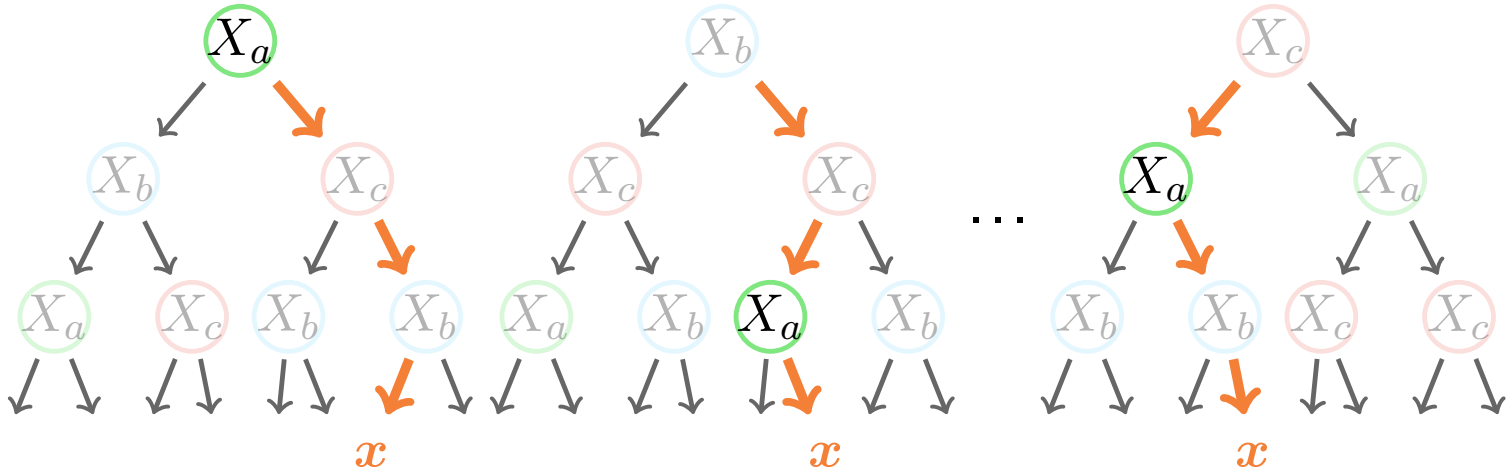
From **global** to local **MDI variable importances** for random forests



Mean Decrease of Impurity (MDI) importance of a variable X_a for predicting the output Y is

$$Imp(X_a) = \frac{1}{N_T} \sum_{T} \underbrace{\sum_{t \in T : \nu(s_t) = X_a}}_{\text{Sum over all nodes splitting on } X_a} p(t) \Delta i(s_t, t)$$

From **global** to local **MDI variable importances** for random forests



Local Mean Decrease of Impurity (MDI) importance of a variable X_a for predicting the output Y ***for a given instance*** $\boldsymbol{x}$ is

$$Imp(X_a, \boldsymbol{x}) = \frac{1}{N_T} \sum_{T} \underbrace{\sum_{t \in T : \nu(s_t) = X_a \wedge \boldsymbol{x} \in t}}_{\text{Sum over all nodes splitting on } X_a \text{ in branches followed by } \boldsymbol{x}} i(t) - i(t_{x_a})$$

Link with global MDI

$$Imp(X_a) = \frac{1}{N} \sum_{i=1}^{N} Imp(X_a, \boldsymbol{x}^i)$$

...and when they are Shapley values: **global MDI**

With ***totally randomized trees*** :

$$Imp_\infty(X_a) = \phi_v^{Sh}(X_a)$$

where ϕ_v^{Sh} is the Shapley value with $v(\cdot) = I(Y;\cdot)$.

Imp_∞ ***does satisfy*** Shapley values properties :

- ✓ Efficiency
- ✓ Symmetry
- ✓ Null player
- ✓ Strong monotonicity

...and when they are Shapley values: **local MDI**

With ***totally randomized trees*** :

$$Imp_\infty(X_a, \boldsymbol{x}) = \phi^{Sh}_{v_{loc(\cdot;\boldsymbol{x})}}(X_a)$$

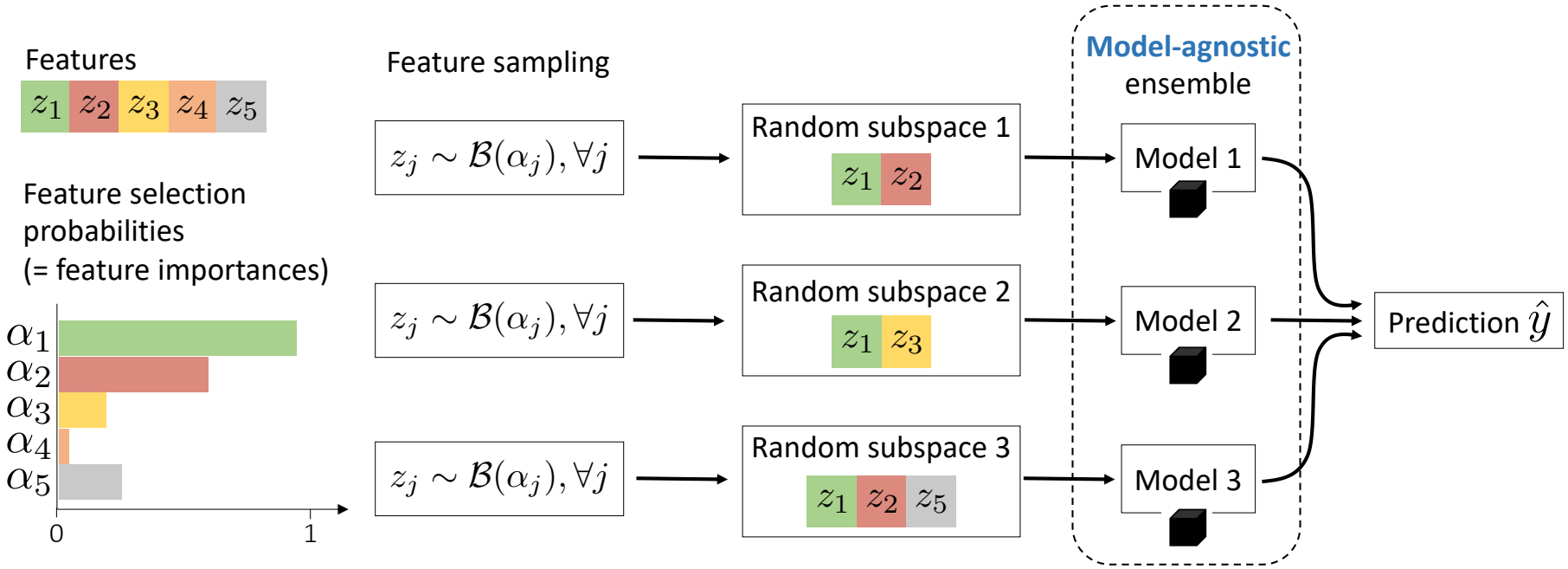
where $\phi^{Sh}_{v_{loc(\cdot;\boldsymbol{x})}}$ is the Shapley value with $v(\cdot;\boldsymbol{x}) = H(Y) - H(y|\cdot = \boldsymbol{x}.)$.

$Imp_\infty(\cdot, \boldsymbol{x})$ ***does satisfy*** Shapley values properties :

- ✓ Efficiency
- ✓ Symmetry
- ✓ Null player
- ✓ Strong monotonicity

Optimizing Model-Agnostic Random Subspace Ensembles: **model**

$$\mathbb{E}[f_z(x)]_{p(z|\alpha)} = \sum_z p(z|\alpha) f_z(x)$$



Optimizing Model-Agnostic Random Subspace Ensembles: **training**

$$\frac{\partial}{\partial \alpha_j}\mathbb{E}[f_z(x_i)]_{p(z|\boldsymbol{\alpha})} = \mathbb{E}\left[f_z(x_i)\frac{\partial}{\partial \alpha_j}\log p(z|\boldsymbol{\alpha})\right]_{p(z|\boldsymbol{\alpha})}$$

(score function estimator)

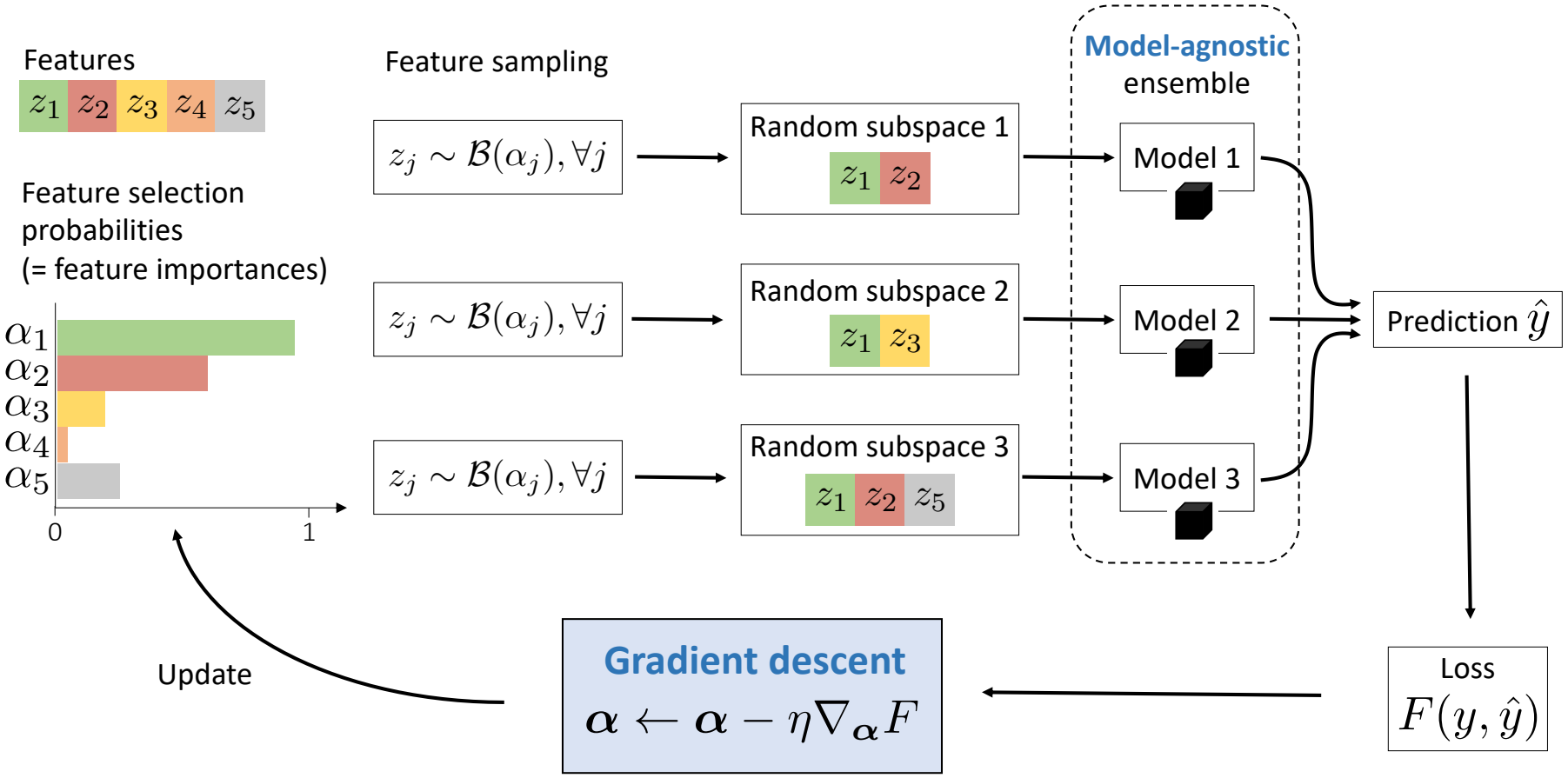


Illustration on noisy MNIST

Objective function:

Sparsity

$$F(\boldsymbol{\alpha}) = \frac{1}{N}\sum_{i=1}^{N} L\left(y_i, \mathbb{E}[f_z(x_i)]_{p(z|\boldsymbol{\alpha})}\right) + \lambda_1 \sum_{j=1}^{W}\sum_{k=1}^{H} \alpha_{j,k}$$

$$+ \lambda_2 \left(\sum_{j=2}^{H} |\alpha_{j,k} - \alpha_{j-1,k}| + \sum_{k=2}^{W} |\alpha_{j,k} - \alpha_{j,k-1}| \right)$$

Spatial smoothness