Trustworthy AI for Business and Society

Ilker Birbil

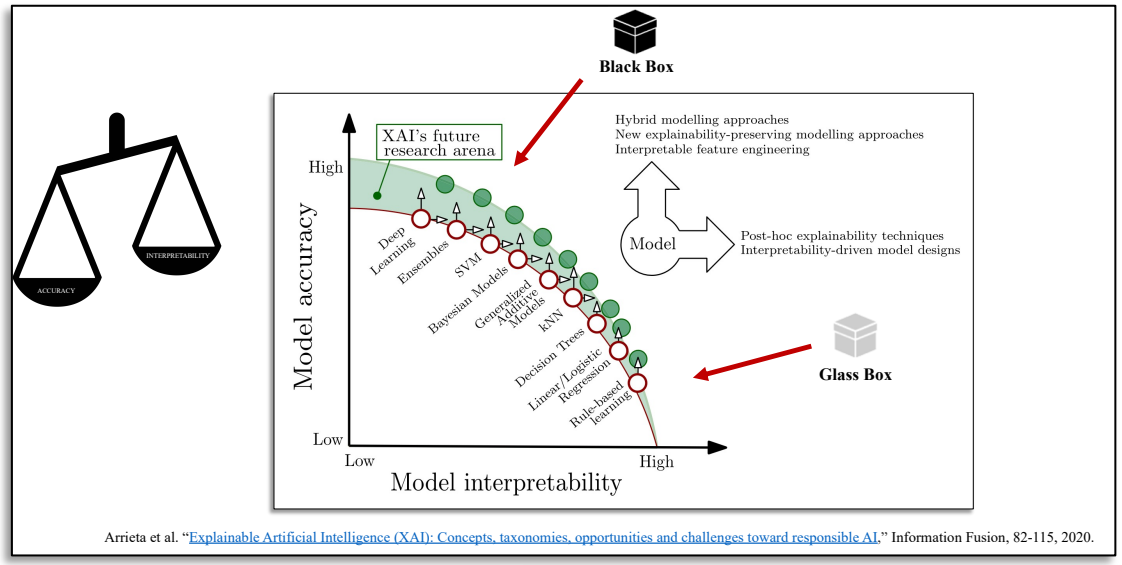
Unboxing Methods

Why Unboxing?

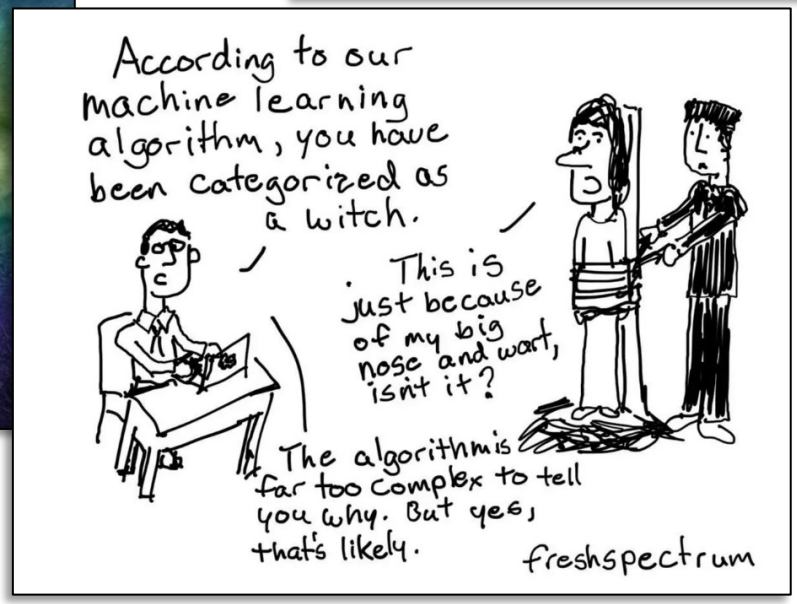
$$g(\mathbb{E}[Y]) = f_0 + \sum_{j=1}^{p} f_j(X_j)$$

contribution of each feature

(link)



Arrieta et al. "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," Information Fusion, 82-115, 2020.



(link)

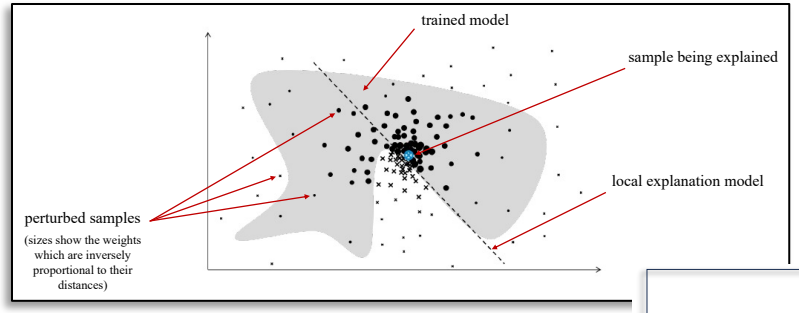
PERSPECTIVE
https://doi.org/10.1038/s42256-019-0048-x
nature machine intelligence

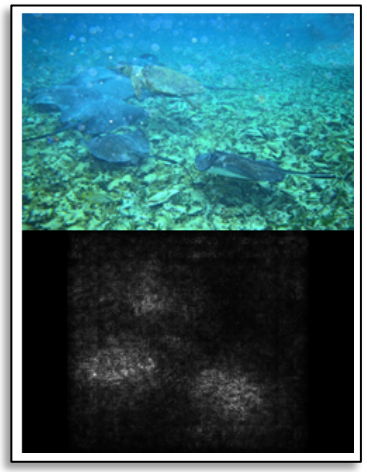
Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead

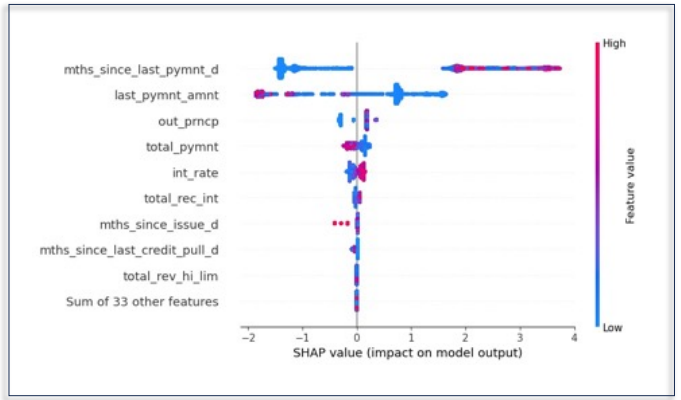
Cynthia Rudin

Big Picture

- LIME
- SHAP
- Gradient Methods
  - Vanilla Gradients
  - SmoothGrad
  - Other Methods







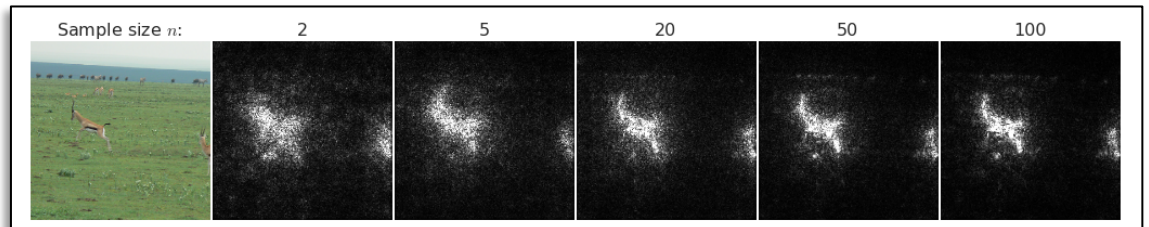


Figure 4. Effect of sample size on the estimated gradient for inception. 10% noise was applied to each image.

LIME (Local Interpretable Model-Agnostic Explanations)

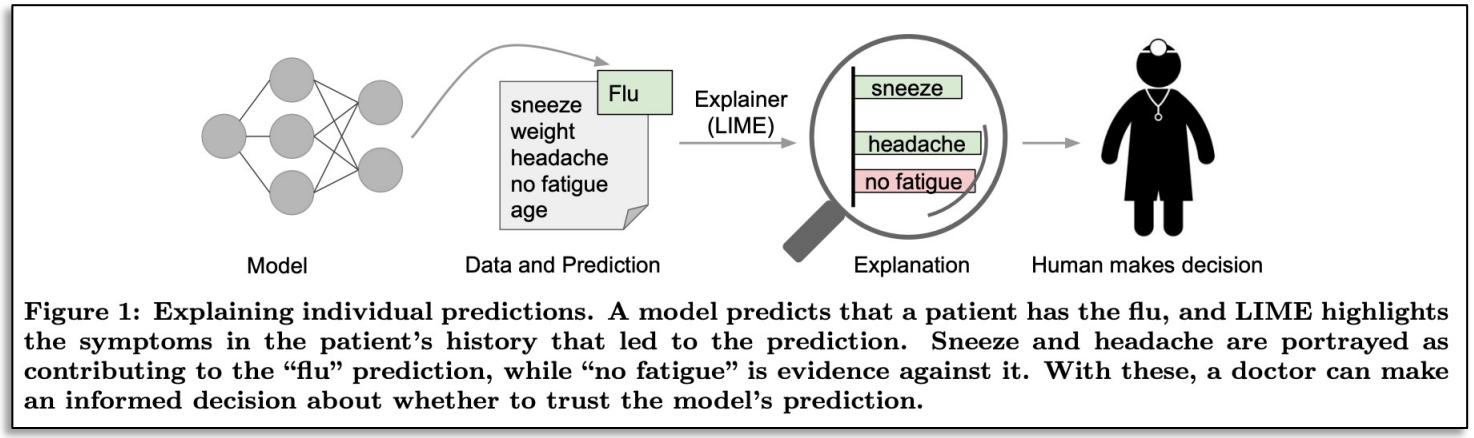


Figure 1: Explaining individual predictions. A model predicts that a patient has the flu, and LIME highlights the symptoms in the patient's history that led to the prediction. Sneeze and headache are portrayed as contributing to the "flu" prediction, while "no fatigue" is evidence against it. With these, a doctor can make an informed decision about whether to trust the model's prediction.

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

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

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

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

"LIME, an algorithm that can explain the predictions of any classifier or regressor in a faithful way, by approximating it locally with an interpretable model."

"[…] an explainer should be able to explain any model, and thus be model-agnostic (*i.e.*, treat the original model as a black box)."

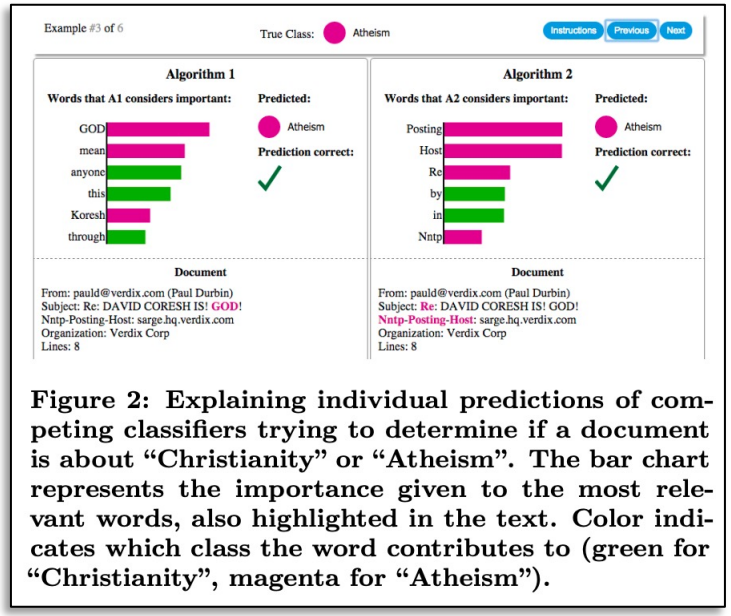
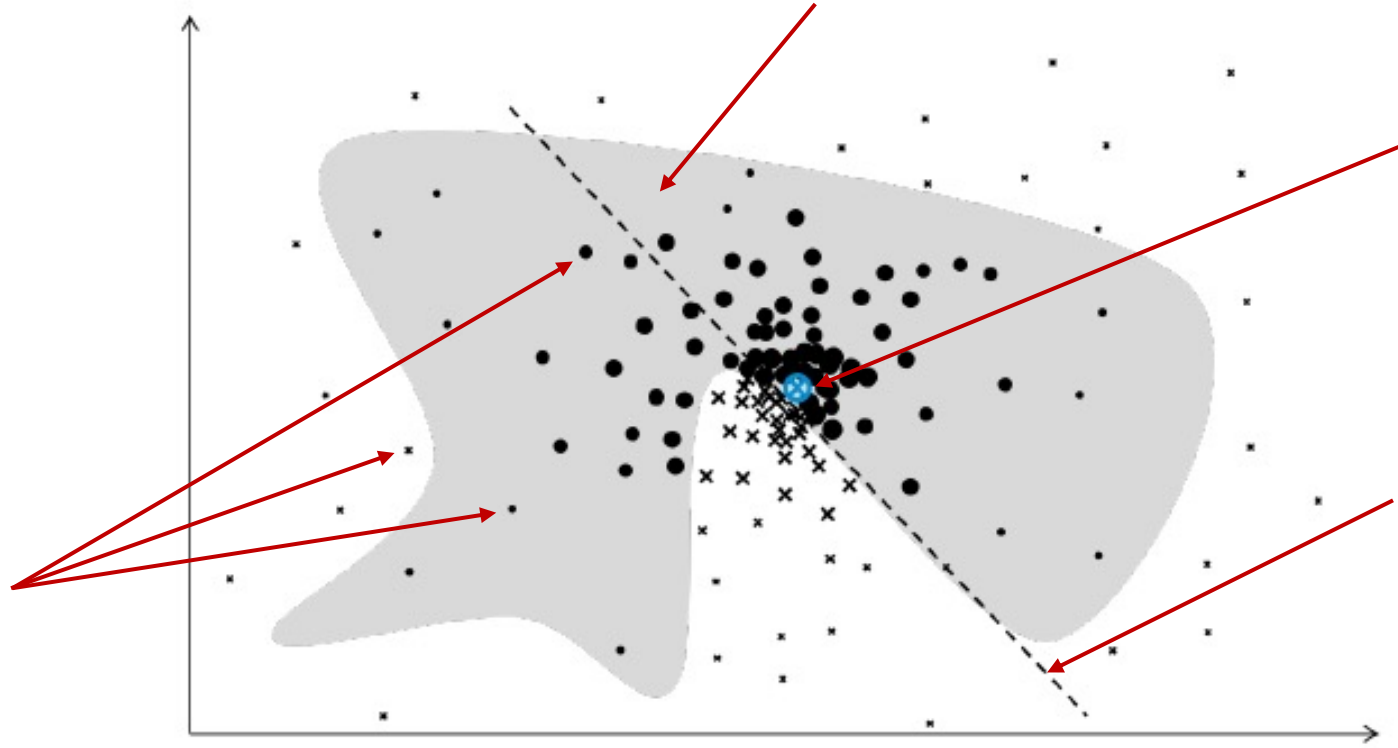


Figure 2: Explaining individual predictions of competing classifiers trying to determine if a document is about "Christianity" or "Atheism". The bar chart represents the importance given to the most relevant words, also highlighted in the text. Color indicates which class the word contributes to (green for "Christianity", magenta for "Atheism").

Ribeiro, M. T., Sing, S. & Guestrin, C. "`Why should I trust you?´ Explaining the predictions of any classifier," KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135-1144, 2016.

LIME


trained model
sample being explained
local explanation model
perturbed samples
(sizes show the weights which are inversely proportional to their distances)

LIME

$\mathcal{I} = \{1, \ldots, n\}$, sample indices

$\mathcal{F} = \{1, \ldots, p\}$, feature indices

f, trained model

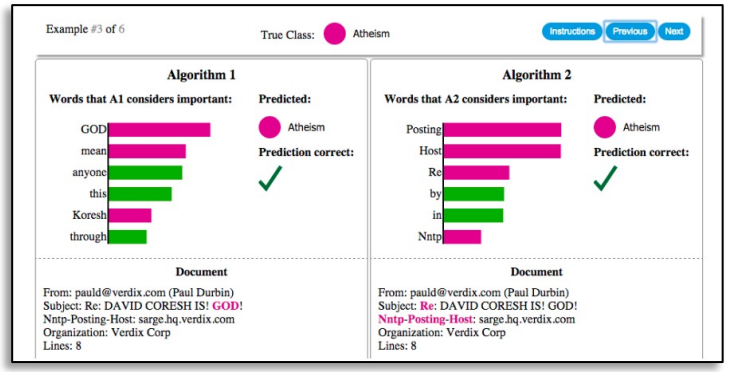
g, local model

interpretable representation
(simplified input)

$$\boldsymbol{x} \in \mathbb{R}^p \longrightarrow \boldsymbol{x}' \in \{0,1\}^{p'}$$

$$\boldsymbol{x} = h_{\boldsymbol{x}}(\boldsymbol{x}')$$

"a word exists (1) or not (0)"



Local models

$$\boldsymbol{z}' \approx \boldsymbol{x}' \quad \text{and} \quad g(\boldsymbol{z}') \approx f(h_{\boldsymbol{x}}(\boldsymbol{z}')) := f_{\boldsymbol{x}}(\boldsymbol{z}')$$

Optimization Problem

f, trained model

g, local model

$$\min_{g \in \mathcal{G}} \ \ell(f, g, \pi_{\boldsymbol{x}'}) + \Omega(g)$$

$g(\boldsymbol{z}') = \boldsymbol{w}_g^\intercal \boldsymbol{z}'$

LIME uses K-LASSO*

loss function

distance

model complexity

class of *potentially* interpretable models

$$\ell(f, g, \pi_{\boldsymbol{x}'}) = \sum_{\boldsymbol{z}' \in \mathbf{Z}} \pi_{\boldsymbol{x}'}(\boldsymbol{z}') \Big(f_{\boldsymbol{x}}(\boldsymbol{z}') - g(\boldsymbol{z}') \Big)^2$$

set of perturbed samples

$$\pi_{\boldsymbol{x}'}(\boldsymbol{z}') = \exp\left(-\frac{D(\boldsymbol{x}', \boldsymbol{z}')^2}{\sigma^2} \right)$$

distance function

kernel width $\in [0, 1]$

* First select K features with Lasso using regularization path, then learn the weights with least squares.

Local to Global

Algorithm 2 Submodular pick (SP) algorithm

for $\boldsymbol{x}_i, i \in \mathcal{I}$ **do**
  $\mathcal{W}_i \leftarrow \textbf{explain}(\boldsymbol{x}_i)$ ▷ Using Algorithm 1
end for
for $j \in \mathcal{F}$ **do**
  $I_j \leftarrow \sqrt{\sum_{i \in \mathcal{I}} |\mathcal{W}_{ij}|}$ ▷ Compute feature importances
end for
$V \leftarrow \{\}$
while $|V| < B$ **do** ▷ Greedy optimization of $\max_{V, |V| \leq B} c(V, \mathcal{W}, I)$
  $V \leftarrow V \cup \text{argmax}_i \, c(V \cup \{i\}, \mathcal{W}, I)$
end while
return V

LIME with K-Lasso

$$g(\boldsymbol{z}') = \boldsymbol{w}_g^\intercal \boldsymbol{z}'$$

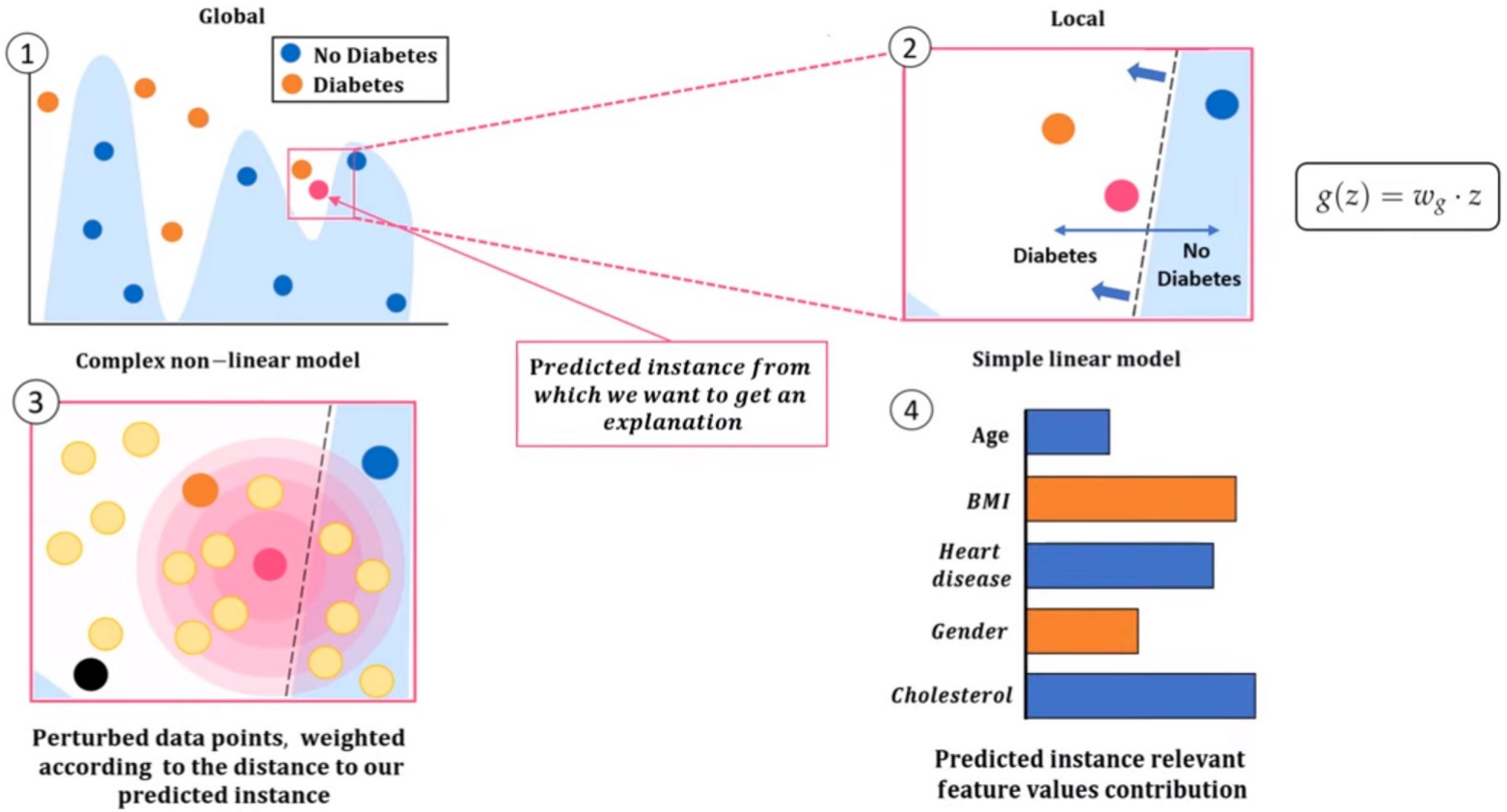
global importance

budget

coverage

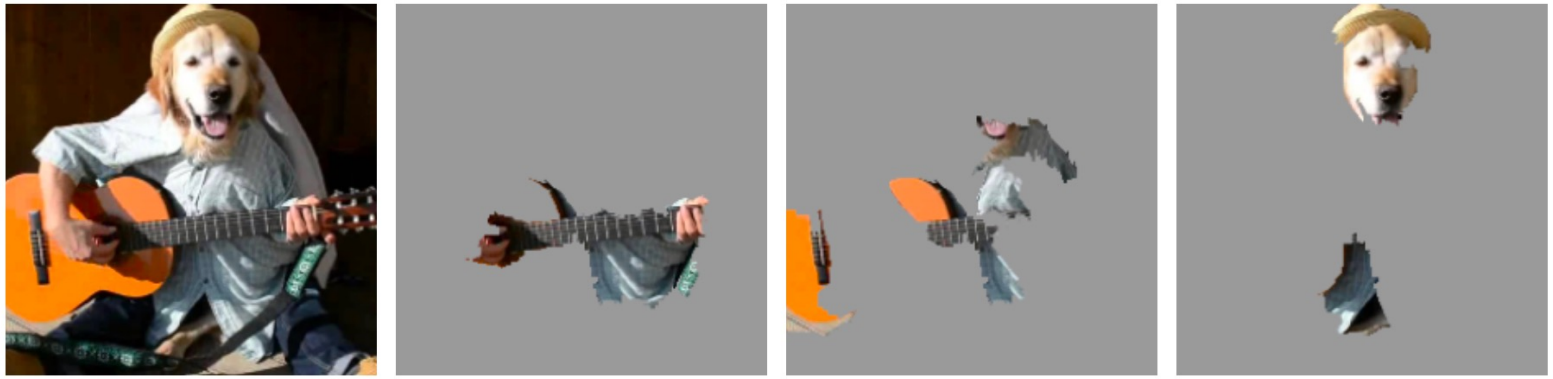
Ribeiro, M. T., Sing, S. & Guestrin, C. "`Why should I trust you?´ Explaining the predictions of any classifier," KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135-1144, 2016.

Interpretation



(source will be added soon)

Interpretation

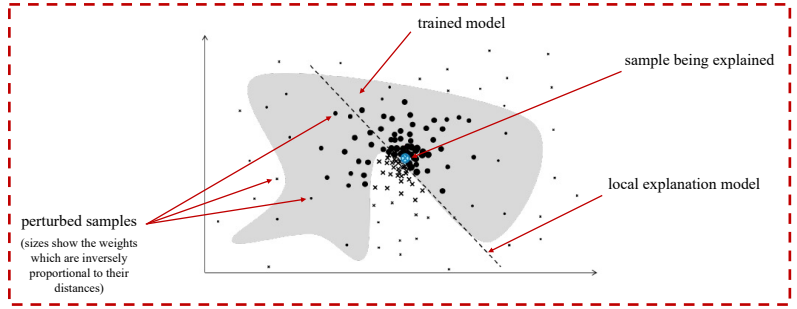


(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Figure 4: Explaining an image classification prediction made by Google's Inception neural network. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$), "Acoustic guitar" ($p = 0.24$) and "Labrador" ($p = 0.21$)

Ribeiro, M. T., Sing, S. & Guestrin, C. "`Why should I trust you?´ Explaining the predictions of any classifier," KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135-1144, 2016.

Interpretation



XGBoost model predicted an acceptance probability of 77.0%

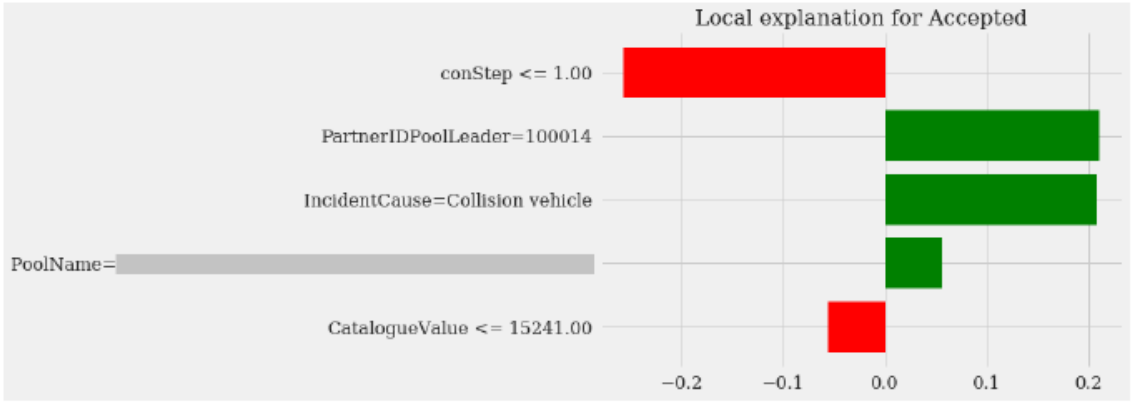


Figure 23: FP - LIME by Default (5 Features)

[Local Pred. = 0.58, Intercept = 0.42, R^2 = 0.28, RMSE = 0.26, Time = 4.67 s]

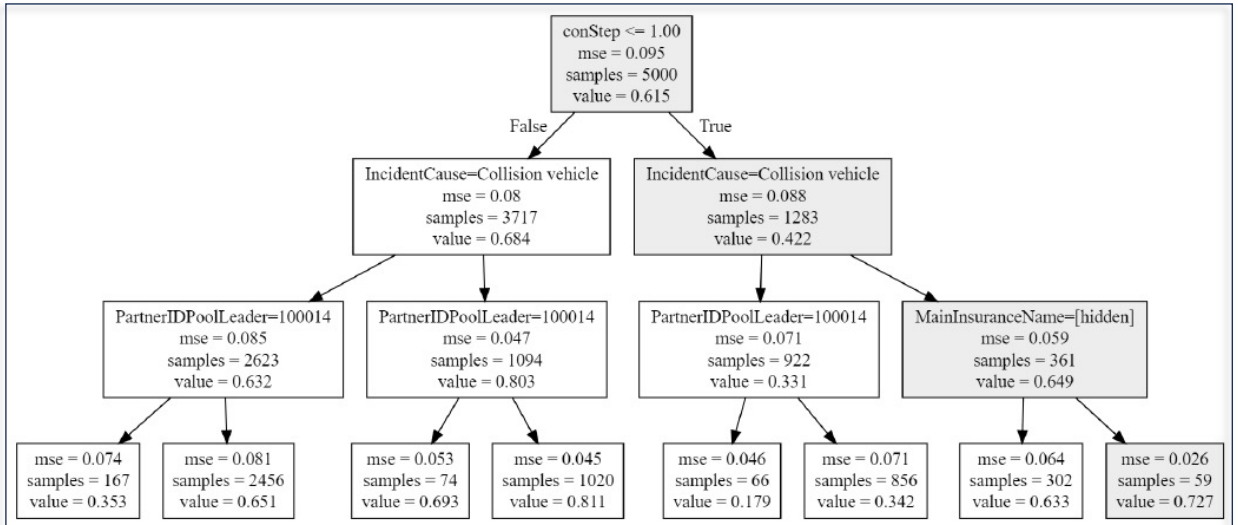
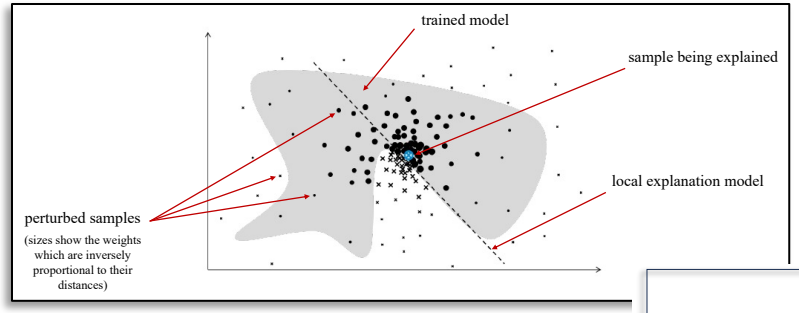


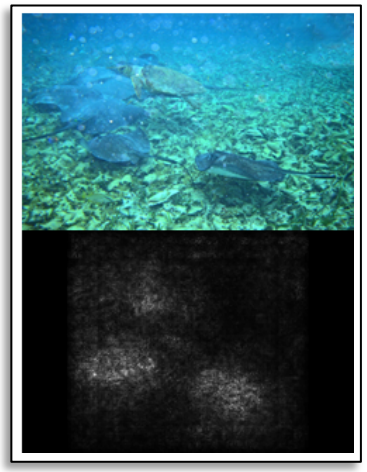
Figure 24: FP - LIME Decision Tree (3 Layers)

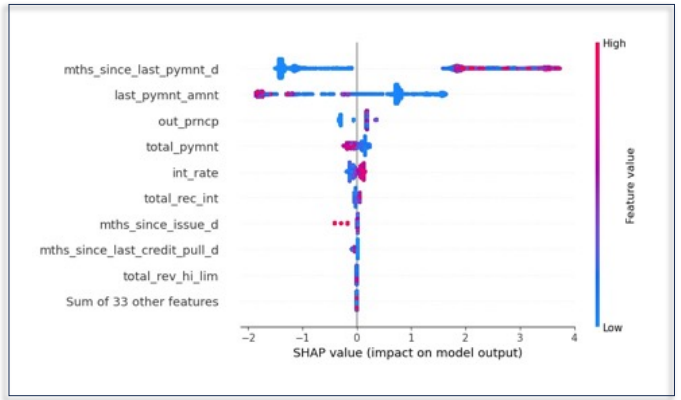
[Local Pred. = 0.73, R^2 = 0.28, RMSE = 0.26, Time = 2.15 s]

Big Picture

- LIME
- SHAP
- Gradient Methods
  - Vanilla Gradients
  - SmoothGrad
  - Other Methods







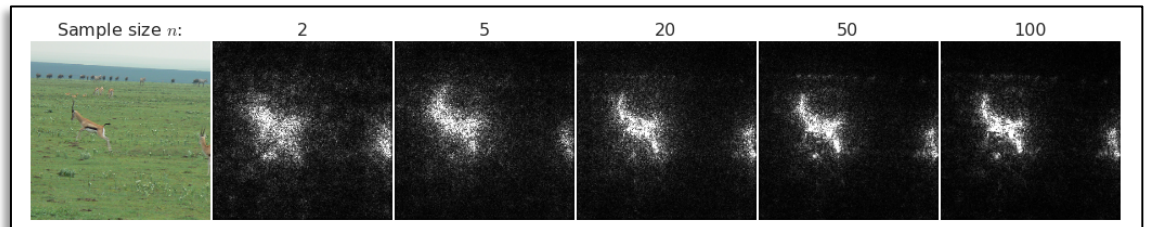


Figure 4. Effect of sample size on the estimated gradient for inception. 10% noise was applied to each image.

Additive Feature Attribution

"SHAP assigns each feature an importance value for a particular prediction. Its novel components include: (1) the identification of a new class of additive feature importance measures, and (2) theoretical results showing there is a unique solution in this class with a set of desirable properties. "

A Unified Approach to Interpreting Model Predictions

Scott M. Lundberg
Paul G. Allen School of Computer Science
University of Washington
Seattle, WA 98105
slund1@cs.washington.edu

Su-In Lee
Paul G. Allen School of Computer Science
Department of Genome Sciences
University of Washington
Seattle, WA 98105
suinlee@cs.washington.edu

(link)

$$g(\boldsymbol{z'}) = \phi_0 + \sum_{j=1}^{p'} \phi_j z'_j$$

presence of feature $\in \{0,1\}^{p'}$

attribution of each feature $\in \mathbb{R}$

Three Requirements

Local Accuracy

$$f(\boldsymbol{x}) = g(\boldsymbol{x}') = \phi_0 + \sum_{j=1}^{p'} \phi_j x'_j \text{ with } \phi_0 = f_{\boldsymbol{x}}(\mathbf{0})$$

Missingness

$$x'_j = 0 \implies \phi_j = 0 \text{ for all } j = 1, \ldots, p'$$

Consistency

For any two models f and $\bar{f}$:

$$\bar{f}_{\boldsymbol{x}}(\boldsymbol{z}') - \bar{f}_{\boldsymbol{x}}(\boldsymbol{z}'_{-j}) \geq f_{\boldsymbol{x}}(\boldsymbol{z}') - f_{\boldsymbol{x}}(\boldsymbol{z}'_{-j}) \text{ for all } \boldsymbol{z}' \in \mathbf{Z} \implies \phi_j(\bar{f}, \boldsymbol{x}) \geq \phi_j(f, \boldsymbol{x})$$

$\boldsymbol{z}'_{-j}$ means $z'_j = 0$

$\phi_j(f, \boldsymbol{x})$ emphasizes the dependence of the attributions on f and $\boldsymbol{x}$

Unique Explanation

Local Accuracy

Missingness

Consistency

+

$$g(\boldsymbol{z}') = \phi_0 + \sum_{j=1}^{p'} \phi_j z'_j$$

$$\phi_j(f, \boldsymbol{x}) = \sum_{\boldsymbol{z}' \subseteq \boldsymbol{x}'} \frac{|\boldsymbol{z}'|!(p' - |\boldsymbol{z}'| - 1)!}{(p')!} \left(f_{\boldsymbol{x}}(\boldsymbol{z}') - f_{\boldsymbol{x}}(\boldsymbol{z}'_{-j}) \right)$$

$\mathcal{O}(2^p)$ subsets !

the number of non-zero entries in $\boldsymbol{z}'$

all vectors $\boldsymbol{z}'$ where the non-zero entries are a subset of the nonzero entries in $\boldsymbol{x}'$

This follows from a classic result in game theory* on distributing the total gain from a cooperative game.

* Shapley, L. S. "A value for n-person games," Contributions to the Theory of Games 2.28, 307–317, 1953.

SHAP (SHapley Additive exPlanation) Values

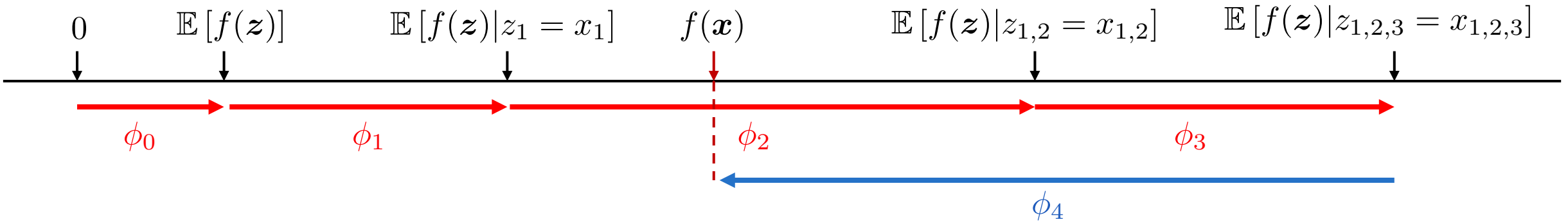
$$\phi_j(f, \boldsymbol{x}) = \sum_{\boldsymbol{z}' \subseteq \boldsymbol{x}'} \frac{|\boldsymbol{z}'|!(p' - |\boldsymbol{z}'| - 1)!}{(p')!} \left(f_{\boldsymbol{x}}(\boldsymbol{z}') - f_{\boldsymbol{x}}(\boldsymbol{z}'_{-j}) \right)$$

$\mathcal{O}(2^p)$ subsets !

$$f_{\boldsymbol{x}}(\boldsymbol{z}') = \mathbb{E}\left[f(\boldsymbol{z}) | \boldsymbol{z}_{\mathcal{S}}\right]$$

the indices of non-zero entries in $\boldsymbol{z}'$

$$h_{\boldsymbol{x}}(\boldsymbol{z}') = \boldsymbol{z}_S$$



The order does matter!

$$f_{\boldsymbol{x}}(\boldsymbol{z}') \approx f([\boldsymbol{z}, \bar{\boldsymbol{z}}_{\mathcal{F}\setminus\mathcal{S}}]) \text{ with } \bar{\boldsymbol{z}}_{\mathcal{F}\setminus\mathcal{S}} := \mathbb{E}\left[\boldsymbol{z}_{\mathcal{F}\setminus\mathcal{S}}\right]$$

the set of indices not in $\mathcal{S}$

Lundberg, S. M. & Lee, S-I. "A unified approach to interpreting model predictions," NIPS '17: Proceedings of the 31st International Conference on Neural Information Processing Systems, 4768–4777, 2017.

LIME Revisited: Kernel SHAP

$$\pi_{\boldsymbol{x}'}(\boldsymbol{z}') = \exp\left(-\frac{D(\boldsymbol{x}', \boldsymbol{z}')^2}{\sigma^2}\right)$$

Local Accuracy ✓

Missingness ✓

Consistency ✓

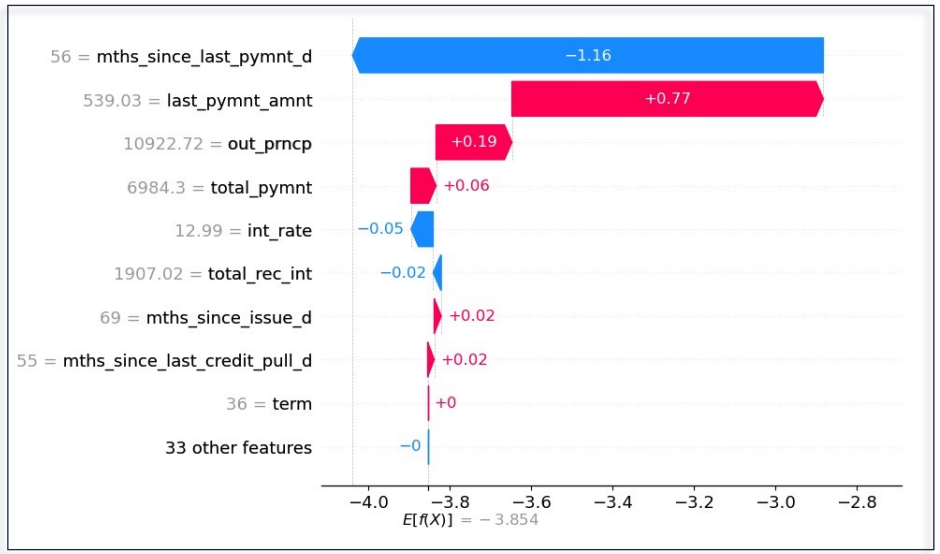
$$\min_{g \in \mathcal{G}} \ \ell(f, g, \pi_{\boldsymbol{x}'}) + \Omega(g)$$

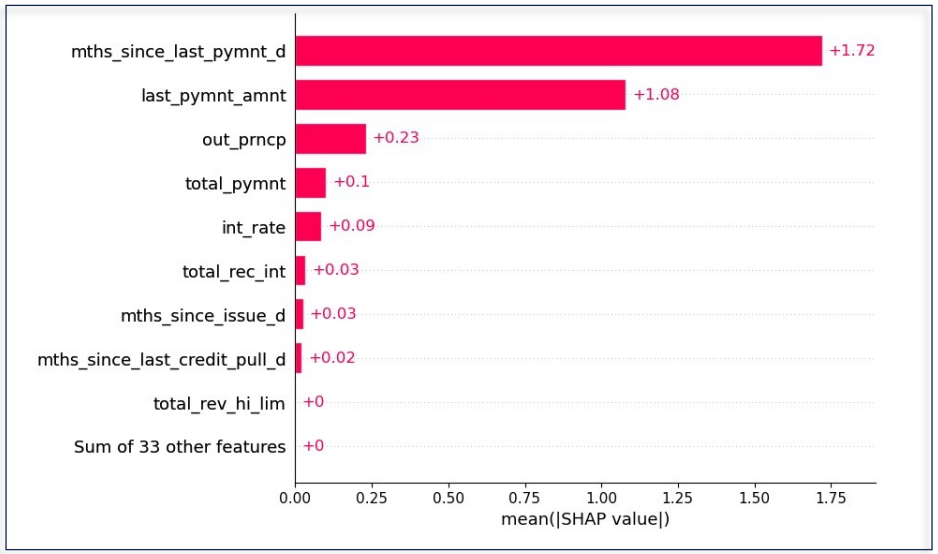
$$\ell(f, g, \pi_{\boldsymbol{x}'}) = \sum_{\boldsymbol{z}' \in \mathbf{Z}} \pi_{\boldsymbol{x}'}(\boldsymbol{z}') \Big(f_{\boldsymbol{x}}(\boldsymbol{z}') - g(\boldsymbol{z}') \Big)^2$$

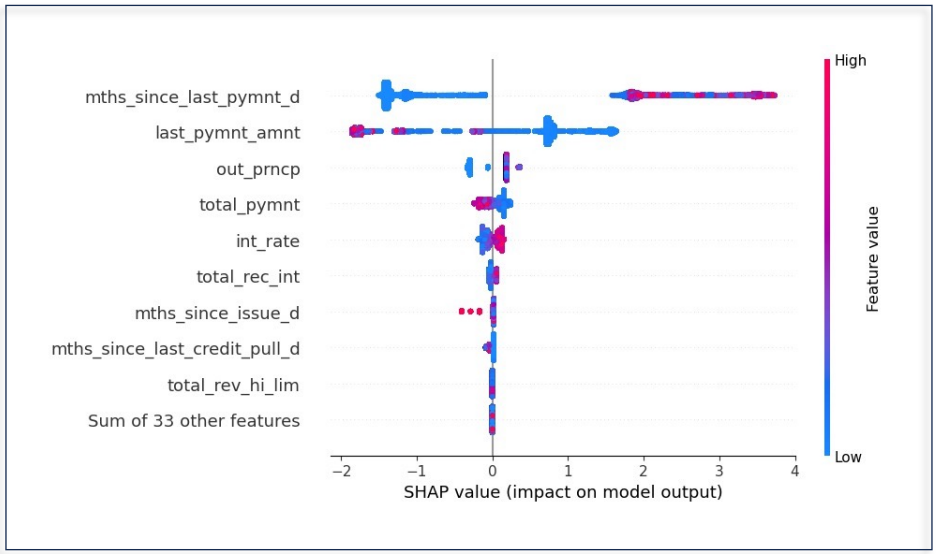
$$\Omega(g) = 0$$

$$\pi_{\boldsymbol{x}'}(\boldsymbol{z}') = \frac{p' - 1}{\binom{p'}{|\boldsymbol{z}'|} |\boldsymbol{z}'|(p' - |\boldsymbol{z}'|)}$$

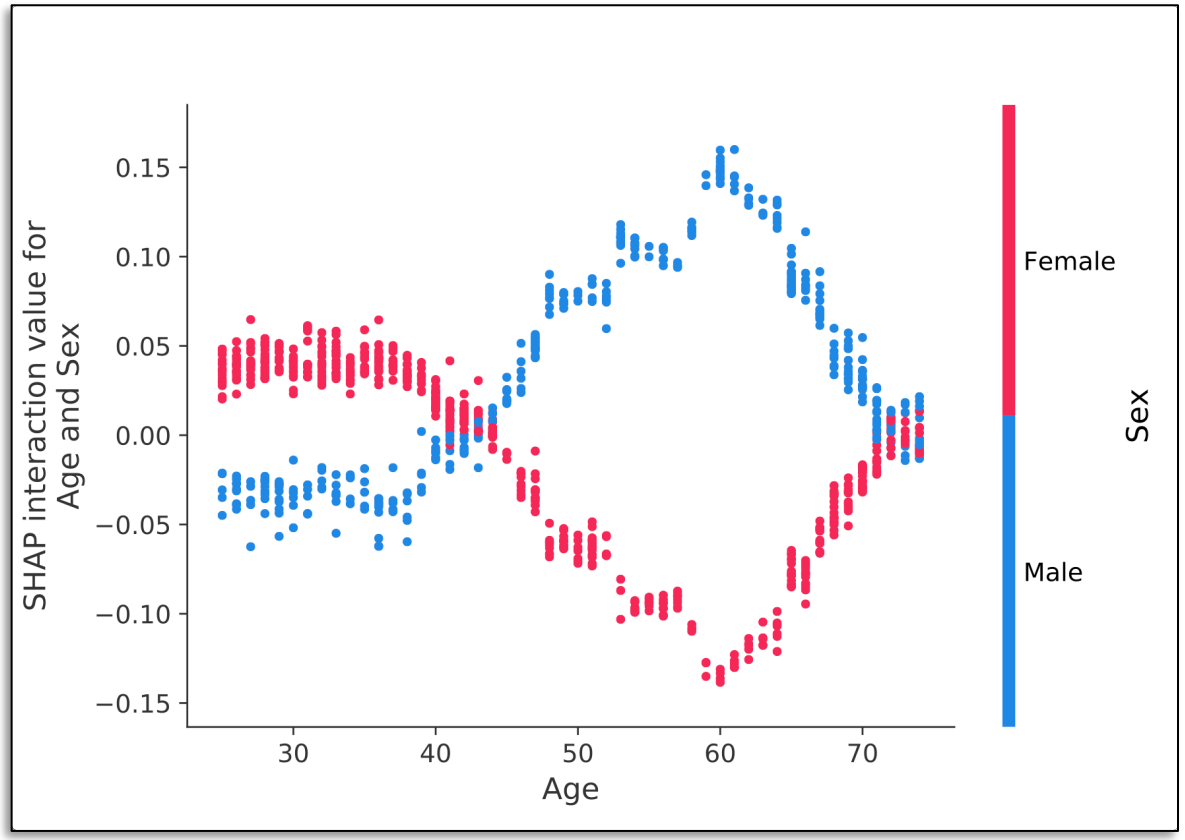
Interpretation



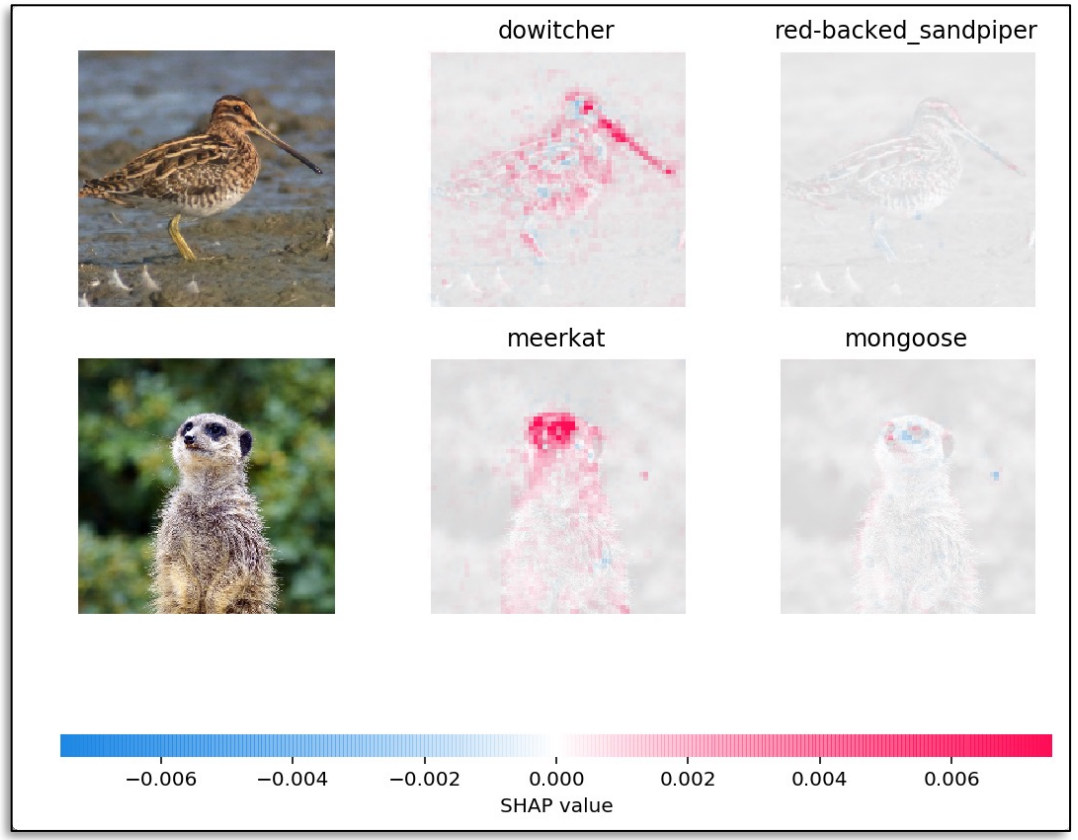




Interpretation



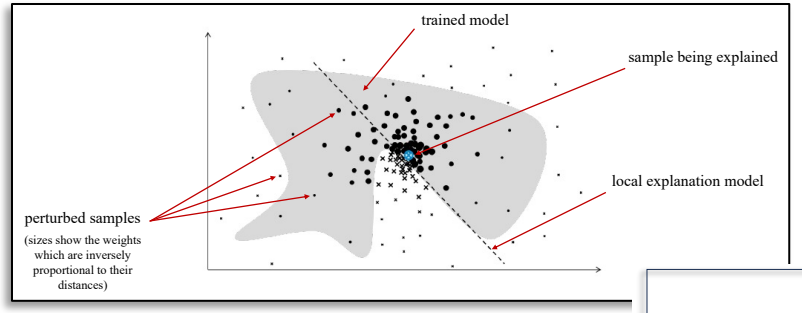
(link)

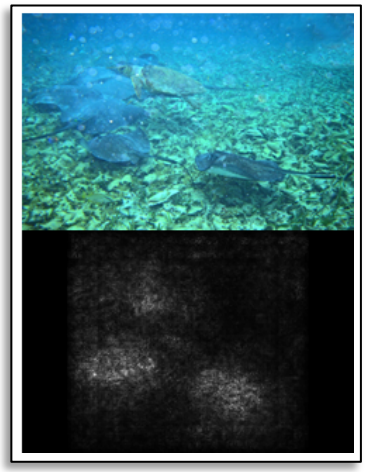


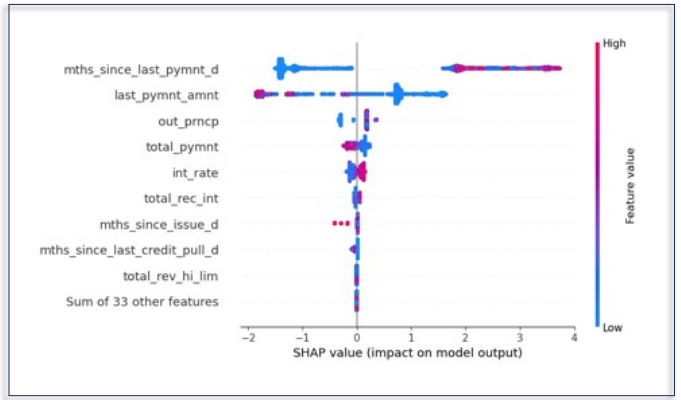
(link)

Big Picture

- LIME
- SHAP
- Gradient Methods
  - Vanilla Gradients
  - SmoothGrad
  - Other Methods







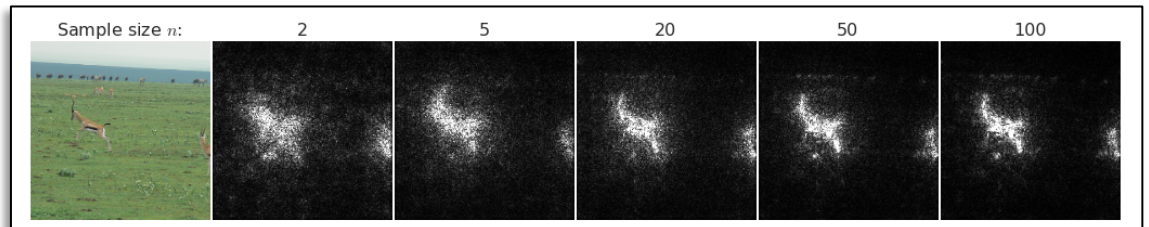


Figure 4. Effect of sample size on the estimated gradient for inception. 10% noise was applied to each image.

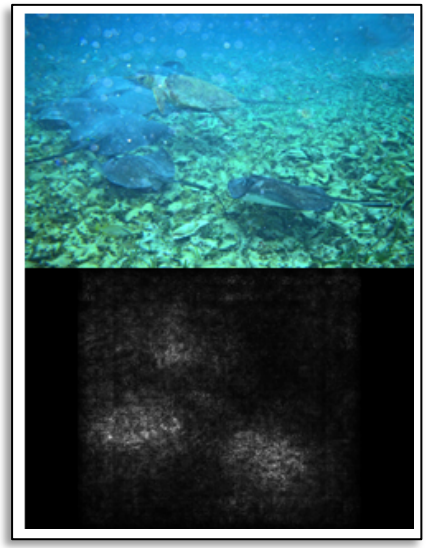
Vanilla Gradient

"We consider two visualisation techniques, based on computing the gradient of the class score with respect to the input image […] The second technique computes a class saliency map, specific to a given image and class. "

Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps

Karen Simonyan **Andrea Vedaldi** **Andrew Zisserman**
Visual Geometry Group, University of Oxford
{karen,vedaldi,az}@robots.ox.ac.uk

(link)



$$S_c(\boldsymbol{x}) \approx \boldsymbol{w}^\intercal \boldsymbol{x} + b$$

score of belonging to class *c*

sample image

$$w_j = \frac{\partial S_c(\boldsymbol{x})}{\partial x_j}$$

SmoothGrad

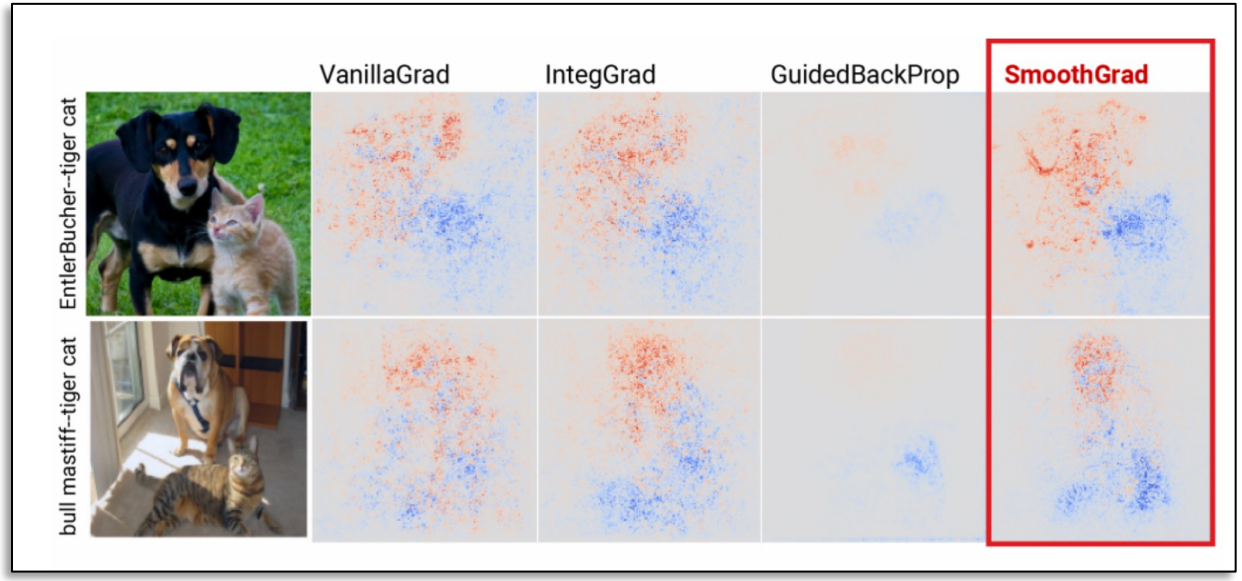
"This paper describes a very simple technique, SMOOTH-GRAD, that in practice tends to reduce visual noise, and also can be combined with other sensitivity map algorithms. The core idea is to take an image of interest, sample similar images by adding noise to the image, then take the average of the resulting sensitivity maps for each sampled image."

"A note on terminology: although the terms 'sensitivity map', 'saliency map', and 'pixel attribution map' have been used in different contexts, in this paper, we will refer to these methods collectively as 'sensitivity maps.'"

SmoothGrad: removing noise by adding noise

Daniel Smilkov [1] **Nikhil Thorat** [1] **Been Kim** [1] **Fernanda Viégas** [1] **Martin Wattenberg** [1]

(link)



SmoothGrad

$$M_c(\boldsymbol{x}) = \nabla S_c(\boldsymbol{x}) = \left[\frac{\partial S_c(\boldsymbol{x})}{\partial x_j}\right]_{j \in \mathcal{F}}$$

$$\hat{M}_c(\boldsymbol{x}) = \frac{1}{m}\sum_{k=1}^{m} M_c(\boldsymbol{x} + \boldsymbol{\varepsilon}_k) \qquad \boldsymbol{\varepsilon}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}^2),\ k = 1, \ldots, m$$

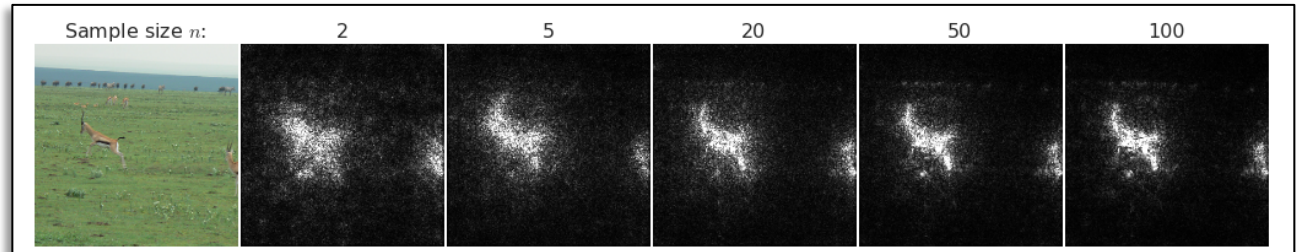


Figure 4. Effect of sample size on the estimated gradient for inception. 10% noise was applied to each image.

Smilkov, D., Thorat, N., Kim, B., Viégas, F. & Wattenberg, M. "SmoothGrad: removing noise by adding noise," arXiv:1706.03825, 2017.

Interpretation

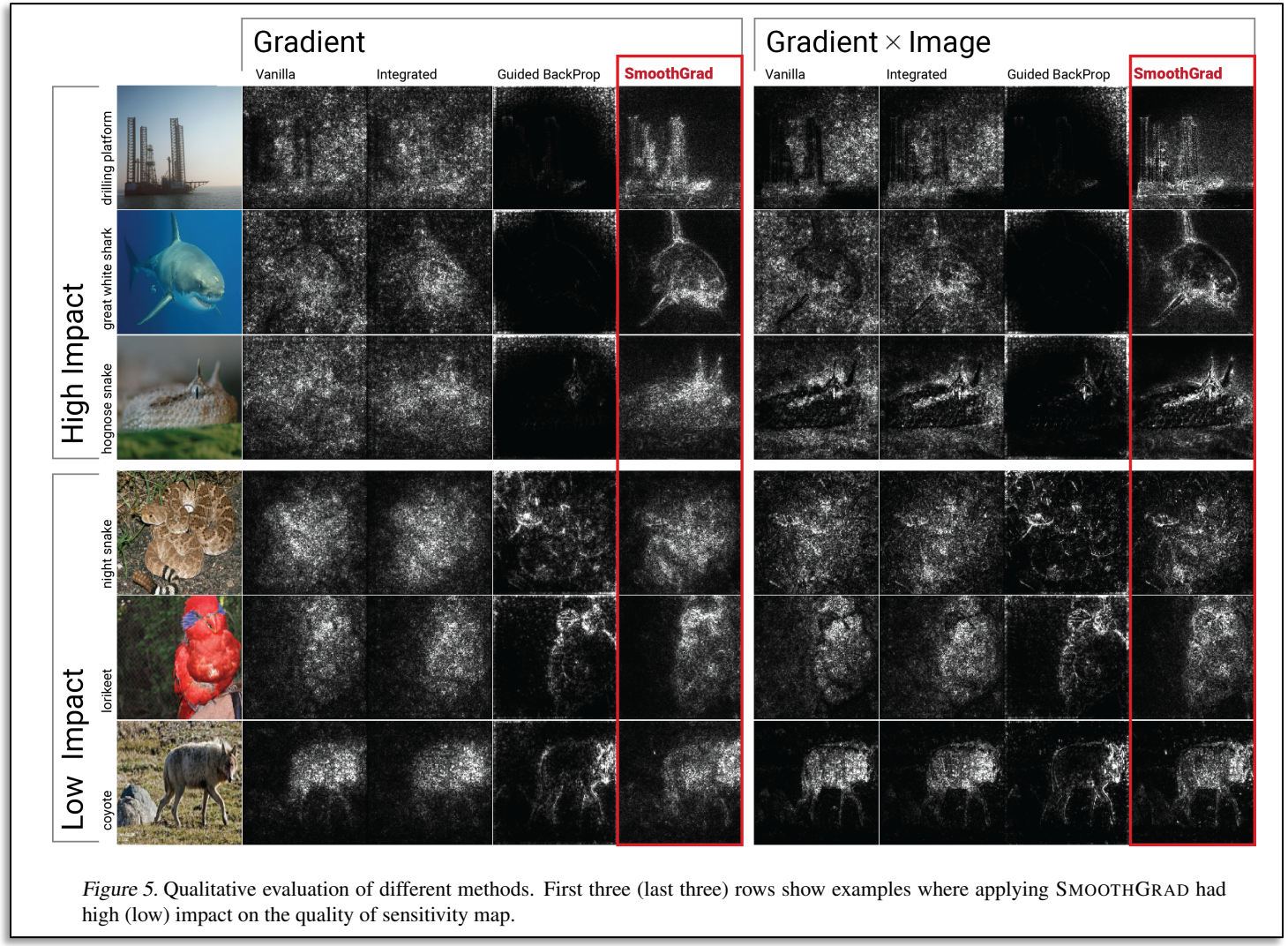


Figure 5. Qualitative evaluation of different methods. First three (last three) rows show examples where applying SMOOTHGRAD had high (low) impact on the quality of sensitivity map.

Smilkov, D., Thorat, N., Kim, B., Viégas, F. & Wattenberg, M. "SmoothGrad: removing noise by adding noise," arXiv:1706.03825, 2017.

Other Methods

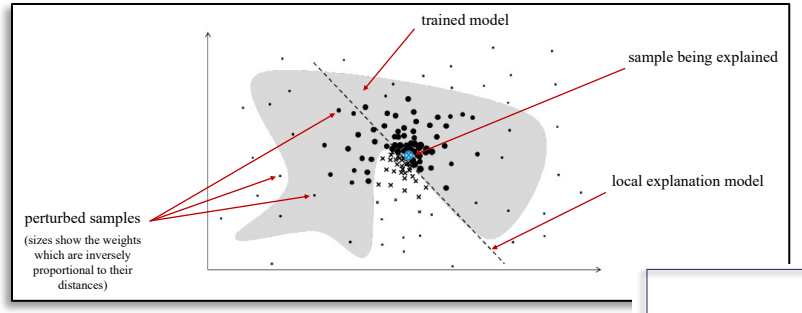
- (Guided) Integrated Gradients
- GradCAM
- FullGrad
- XRAI
- …

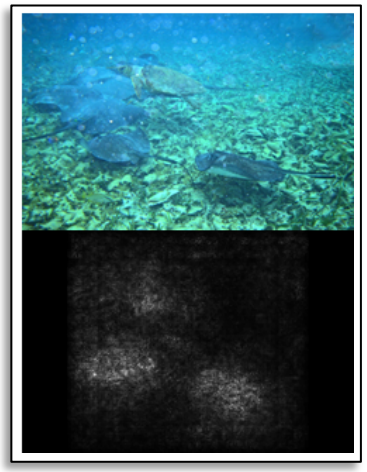
(link)

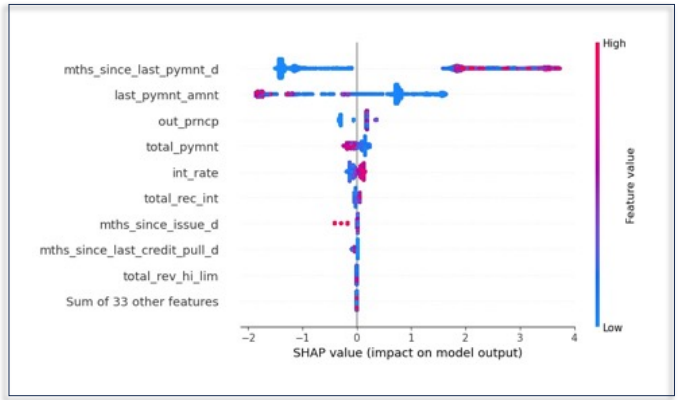
(link)

Big Picture

- LIME
- SHAP
- Gradient Methods
  - Vanilla Gradients
  - SmoothGrad
  - Other Methods







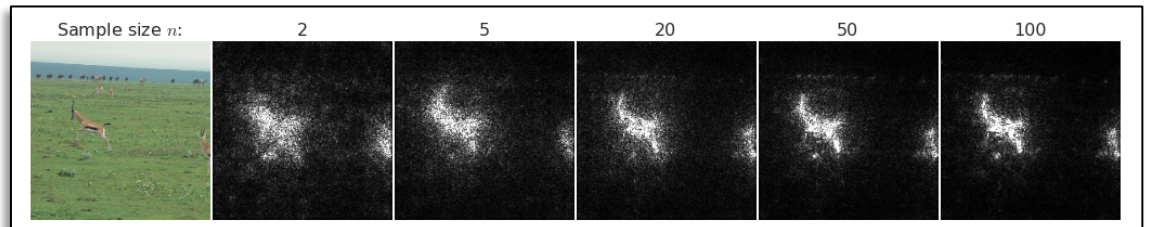


Figure 4. Effect of sample size on the estimated gradient for inception. 10% noise was applied to each image.

Reading Material