[Guidotti et al. 2017]

Explaining Deep Learning Methods

- 1. Explaining Models
  - 1.1. Activation Maximization
  - 1.2. Data Generation
  - 1.3. Model Simplification/ Global Surrogates
- 2. Explaining Outcome
  - 2.1. Perturbation
  - 2.2. Gradient based
  - 2.3. Backpropagation (Decomposition)

Part 3: Interpretable Deep Learning

- **Explaining Models (EM)**
- Explaining Outcome (EO)

* Most of the slides comes in this section comes from

- ICASSP 2017 Tutorial and CVPR'18 Tutorial by W. Samek, G. Montavon and K.R. Müller [ICASSP 2017 Tutorial] [CVPR'18 Tutorial]
- G. Montavon, et al. "Methods for interpreting and understanding deep neural networks," *Digit. Signal Process.*, vol. 73, pp. 1–15, 2018.
- R. Guidotti et al., "A Survey of Methods for Explaining Black Box Models," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–42, Aug. 2018.

Class Prototypes (CP)

- *"How does a goose typically look like according to the neural network?"*

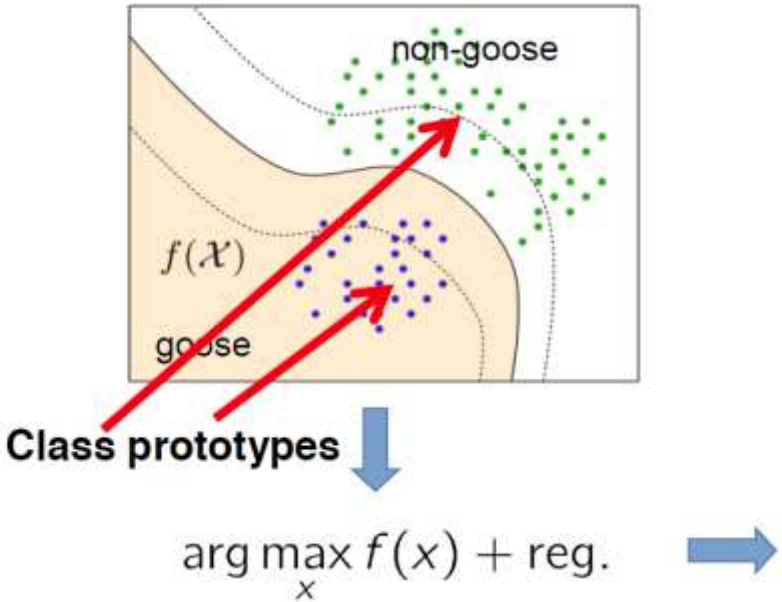
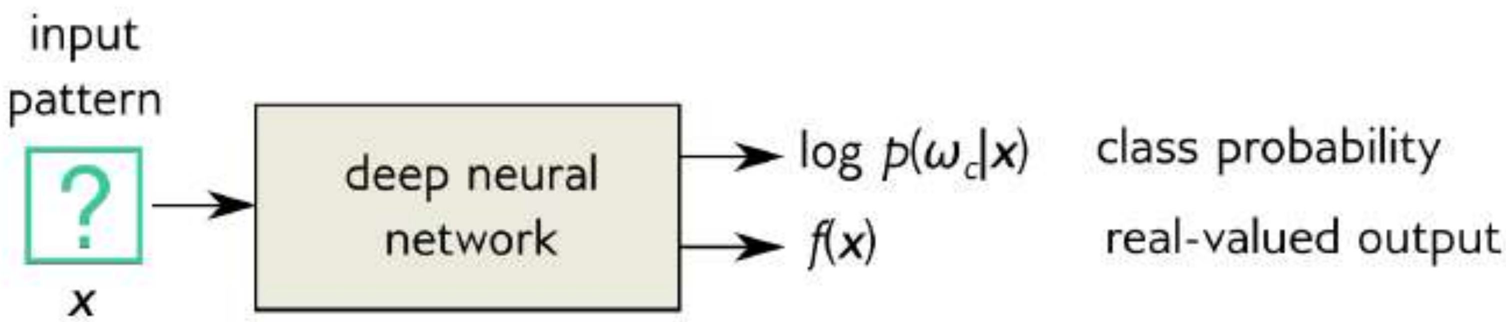


Image from **Symonian'13**

[CVPR'18 Tutorial]

Activation Maximization (AM)

Interpreting concepts predicted by a deep neural net via activation maximization



- Example :
  - Creating class prototype: $argmax_{\boldsymbol{x} \in \chi} \log p(w_c|\boldsymbol{x})$
  - Synthesizing extreme case: $argmax_{\boldsymbol{x} \in \chi} f(\boldsymbol{x})$

Activation Maximization

- [Erhan et al. 2010] – Find image that maximize neuron activity in of interest in Deep Belief Network
- [Le et al. 2012] – Visualize class model in Autoencoder
- [Simonyan et al. 2014] – Saliency map of CNN
- [Nguyen et al. 2016]
- …

Saliency Map via AM

Saliency map of goose and ostrich from **Simonyan et al. 2013**

Problem: Saliency map obtained by AM

1) often not resembling true data,

2) can be uninterpretable to humans

Improving Activation Maximization

- **Idea:** Force the features learned to match the data more closely.
- Now the optimization problem become

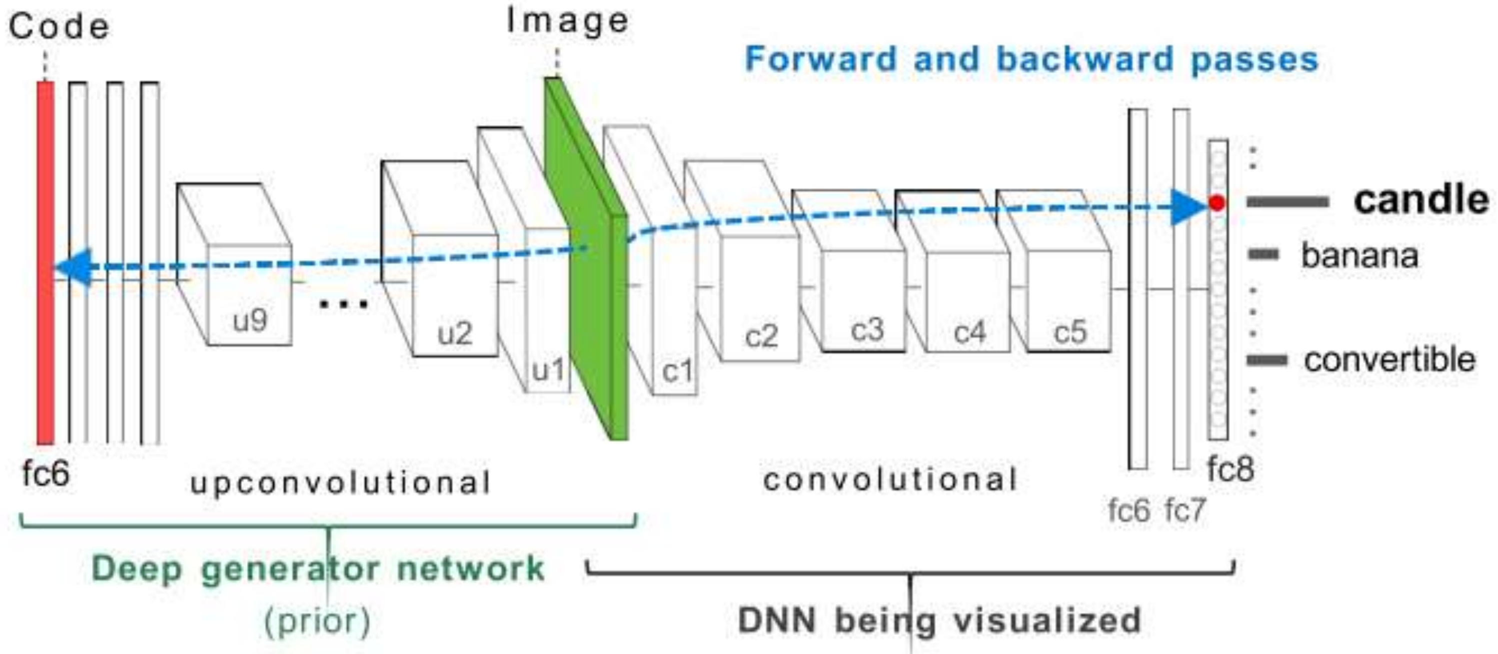
Finding the input pattern that maximizes **class proba bility**. p(**w**|**x**)

Find the **most likely input pattern** for a given class. p(**x**|**w**)

Data Generation

Problem: Activation maximization problem as finding a code $\boldsymbol{y}^l$ such that:

$$\widehat{\boldsymbol{y}^l} = \underset{\boldsymbol{y}^l}{\arg\max}\, \Phi_h\left(G_l(\boldsymbol{y}^l)\right) - \lambda\|\boldsymbol{y}^l\|$$



Deep generator network proposed by Nguyen et al. 2016

Model Simplification/ Global Surrogates

- Model Simplification – AKA Model Compression
  - Applied more for embedded programing then to interpretation
- Global Surrogates – Simple models often fails for DNN cases.

[Watanabe et al 2018]

Modular Representation

- Trained network
- Trained network
- Community structure
- Modular representation
  - bundled connections are defined that summarize multiple connections between pairs of detected communities

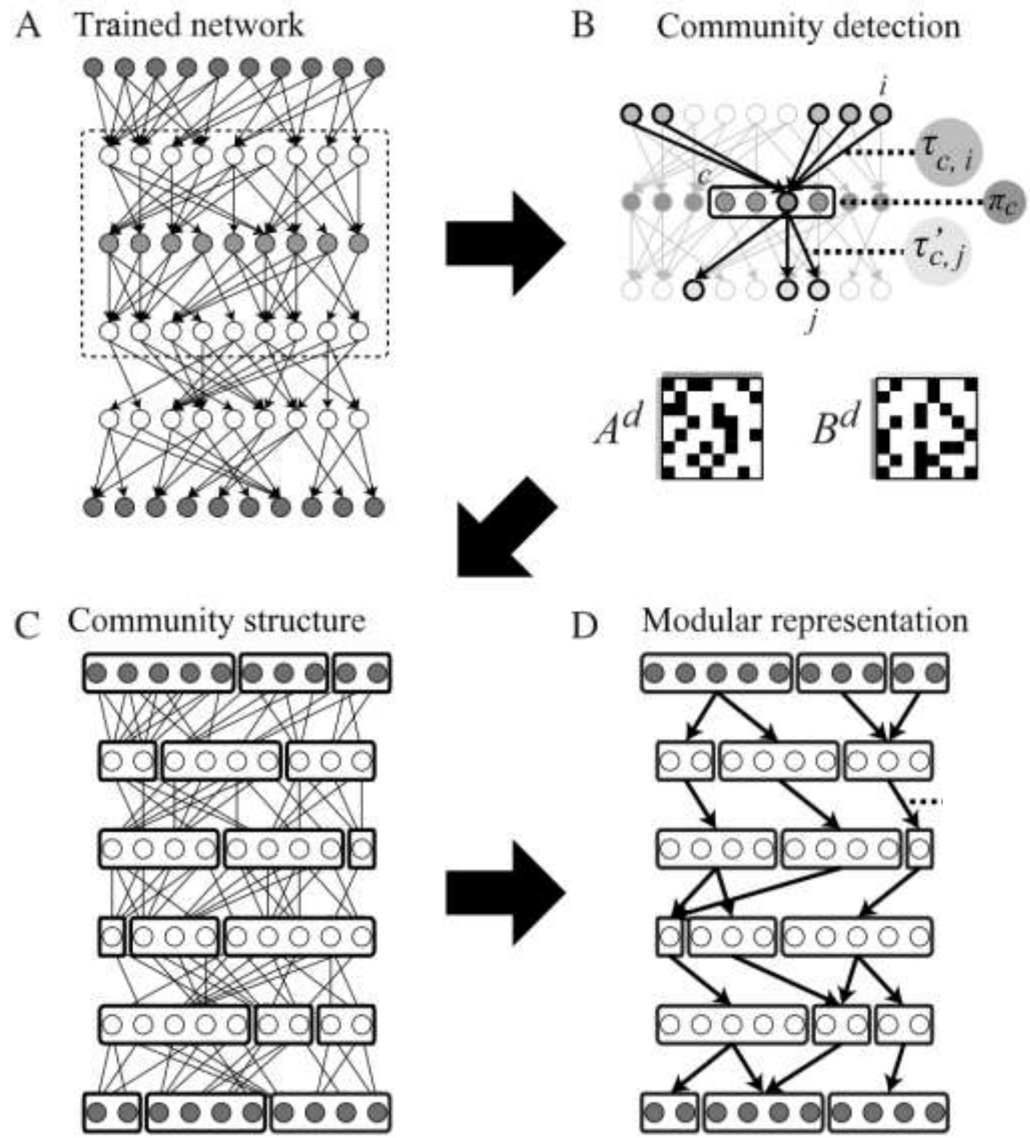


Fig 1. of Watanabe et al. 2018

Part 3: Interpretable Deep Learning

- Explaining Models (EM)
- **Explaining Outcome (EO)**

* Most of the slides comes in this section comes from

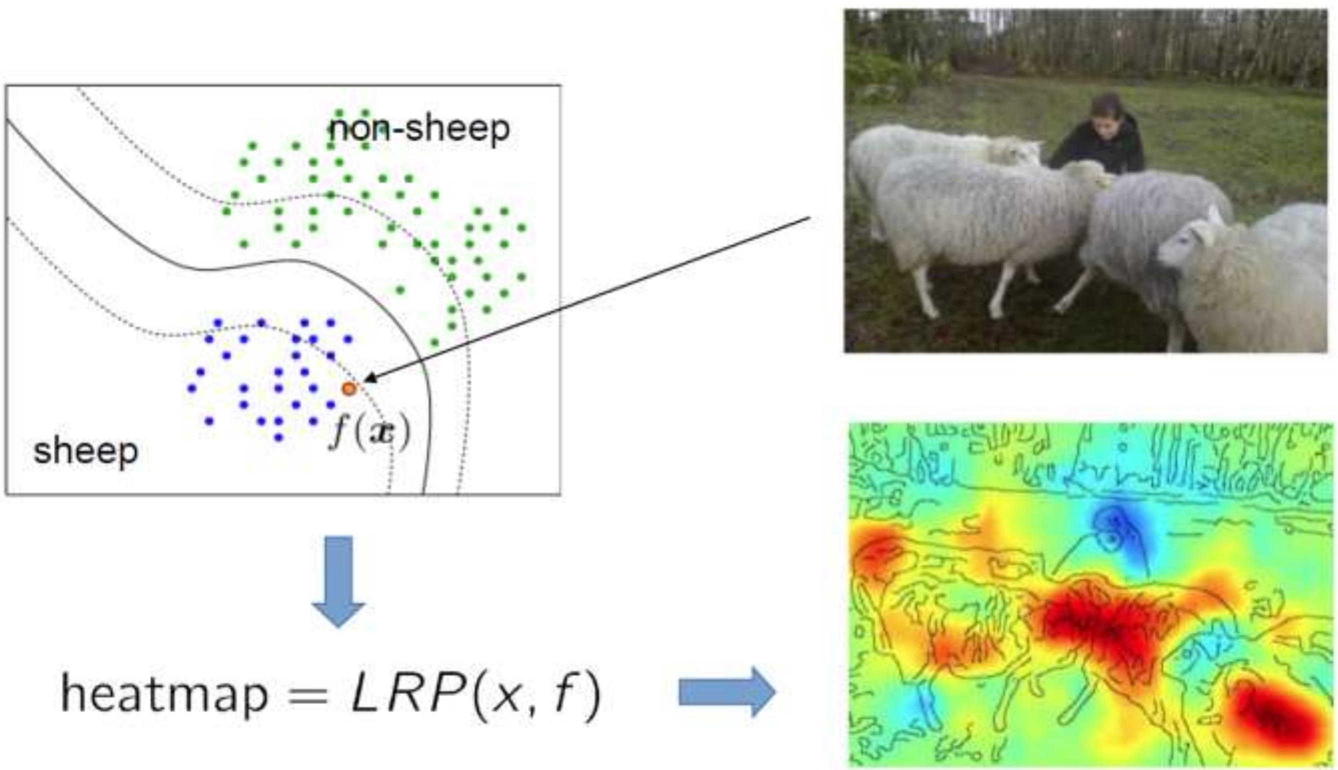
- ICASSP 2017 Tutorial and CVPR'18 Tutorial by W. Samek, G. Montavon and K.R. Müller [ICASSP 2017 Tutorial] [CVPR'18 Tutorial]
- G. Montavon, et al. "Methods for interpreting and understanding deep neural networks," *Digit. Signal Process.*, vol. 73, pp. 1–15, 2018.
- R. Guidotti et al., "A Survey of Methods for Explaining Black Box Models," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–42, Aug. 2018.

Explaining Outcome

- **Goal:** Determine the relevance of each (set of) input feature for a given decision on an instance, by assigning to these variables **a scores to each (set of) feature.**
- Important for **Personalized Healthcare**
- Most DNN explained via a **Saliency Mask**
  - Feature importance that is presented in a visual form to show subset of the original input which is mainly responsible for the prediction.

Explaining Individual Outcome

- *EX> "Why is a given image classified as a sheep?"*



Images from **Lapuschkin'16**

Saliency Map Examples

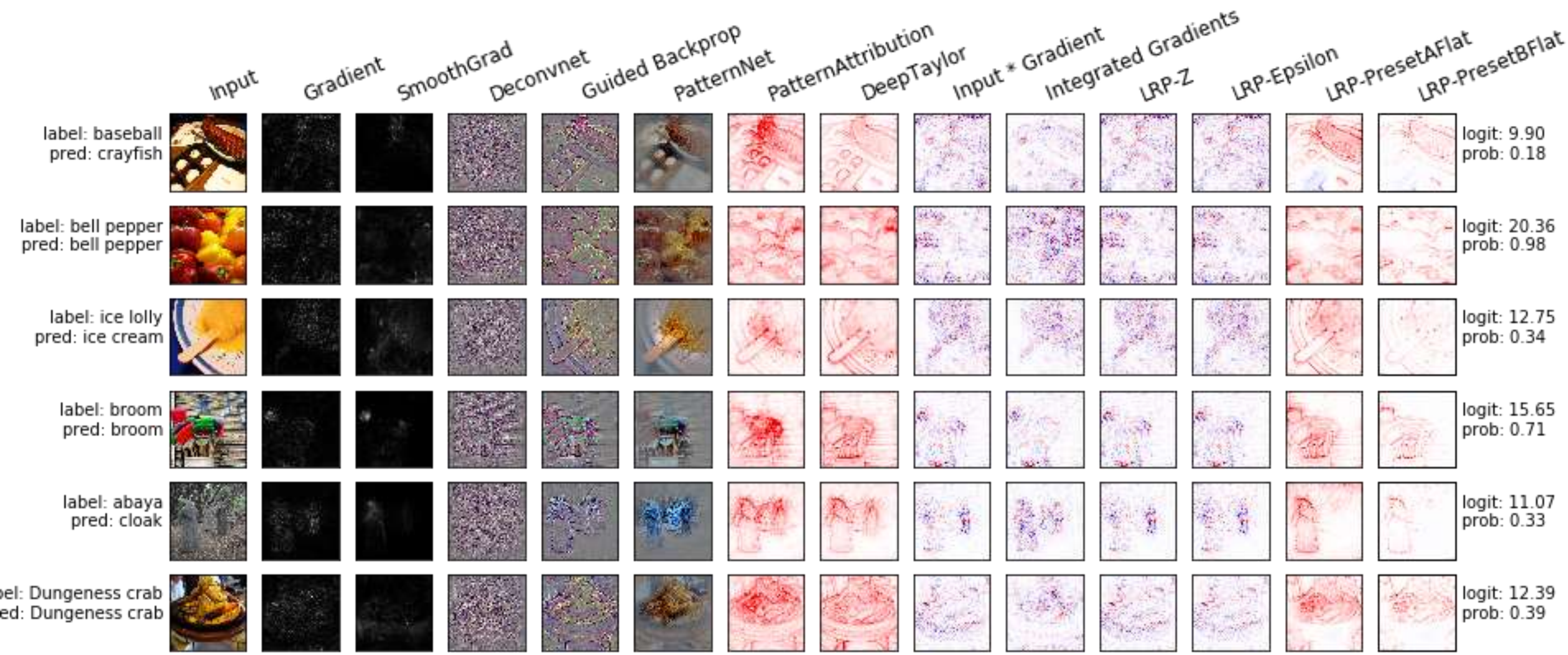


Figure from https://github.com/albermax/innvestigate

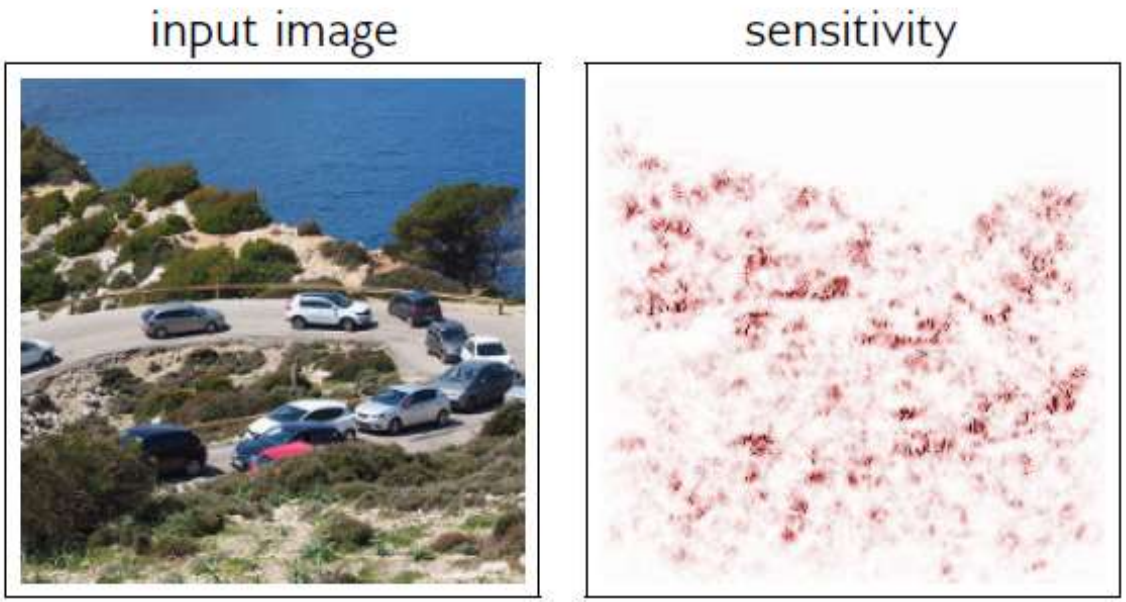
Explaining by Sensitivity Analysis

Given prediction function $f\,(x_1, x_2, \dots, x_d)$ on d dimensional input data $\boldsymbol{x} = (x_1, x_2, \dots, x_d\,)$,

Sensitivity analysis is the measure of local variation of the prediction function f along each input dimension

$$R_i = \left(\frac{\partial f}{\partial x_i}\Big|_{x=x}\right)^2$$

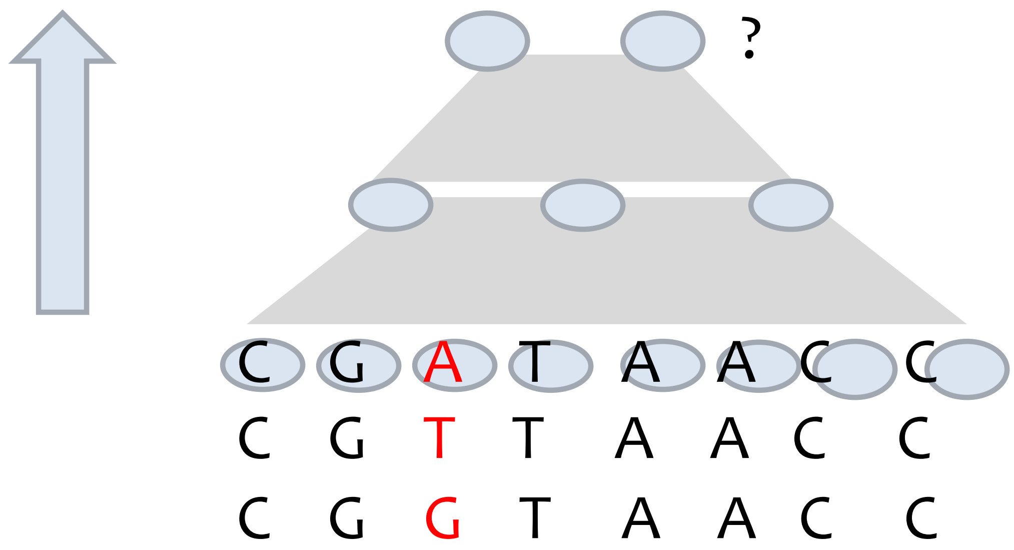
Sensitivity Analysis



- Easy to implement
  - Requires access to the **gradient** of the decision function
  - May not explain the prediction well

Perturbation Approaches

- Make perturbation to input and observe the difference in the output
- ☹ Every time you make a perturbation output needs to be recomputed



[Fong and Vedaldi ICCV'17]

Meaningful Perturbation

The aim of saliency is to identify which regions of an image x are used by the black box to produce the output value f(x) by "deleting" different regions R of x

"deletions":

Class Activation Mapping (CAM)

- linear combination of a late layer's activations and class-specific weights

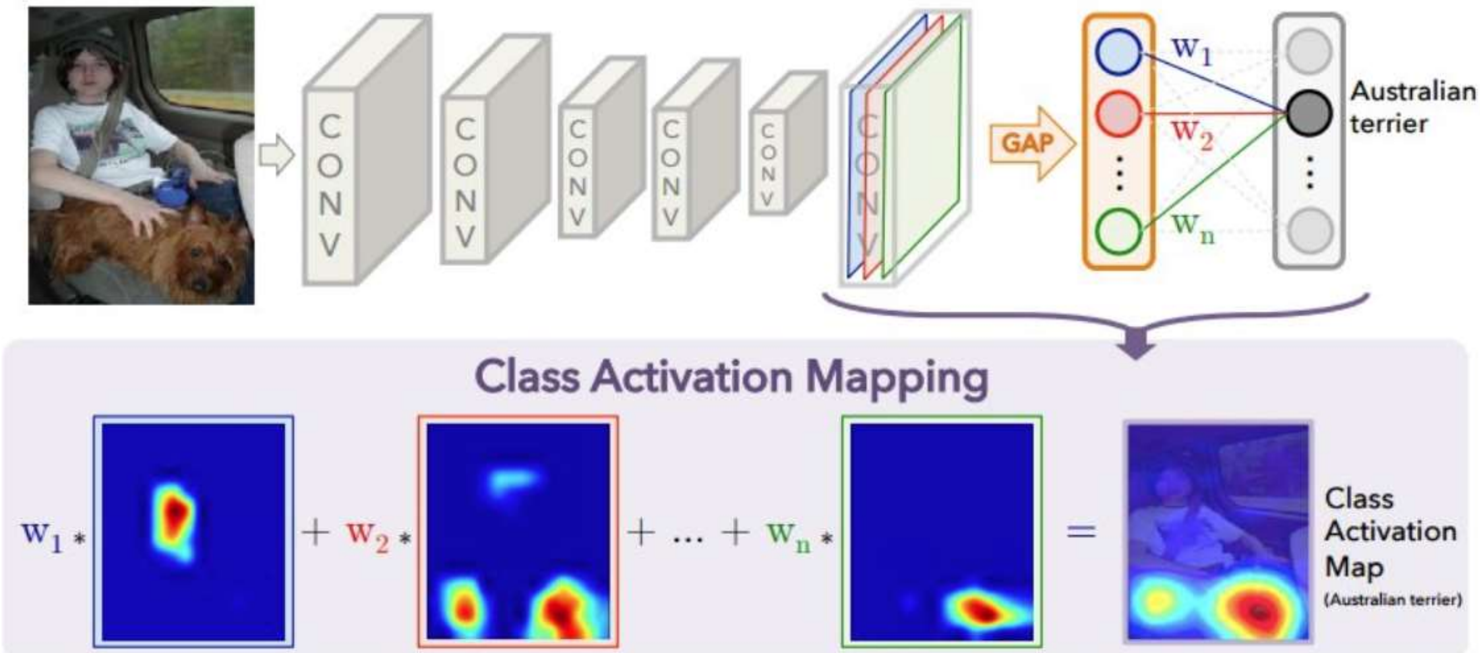


Figure from http://cnnlocalization.csail.mit.edu/

Gradient-Weighted CAM (Grad-CAM)

- Linear combination of a late layer's activations and class-specific gradients

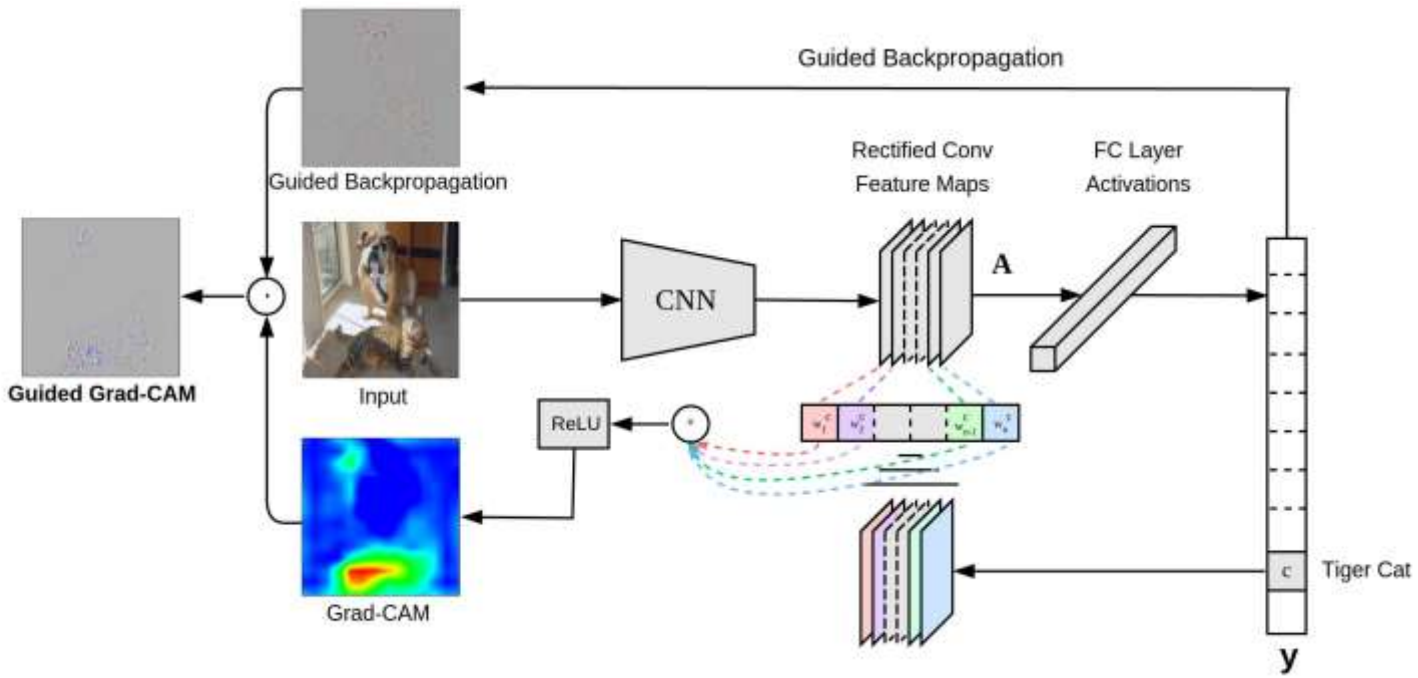
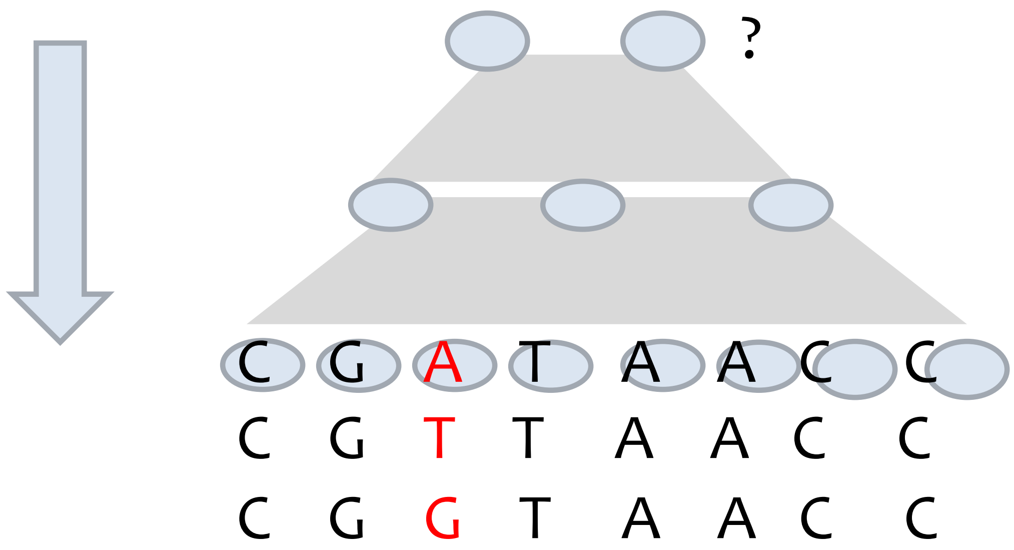


Figure from Selvaraju et al.

Backpropagation methods

- Sensitivity analysis
- Layer-wise relevance propagation (Deep Tylor)
- DeepLIFT



Explaining by Decomposing

Decomposition methods decompose prediction value f(x) to **relevance scores** R_i such that

$$\sum_i R_i = f(x_1, \ldots, x_d)$$

Decomposition **explains the function value** itself.

Sensitivity Analysis in Decomposition View

- Decomposition: $\sum_i R_i = f(x_1, \dots, x_d)$

- Sensitivity Analysis:

$$R_i = \left(\frac{\partial f}{\partial x_i}\Big|_{x=x}\right)^2$$

$$\sum_i R_i = \|\nabla_x f\|^2$$

  - Sensitivity analysis **explains a variation** of the function.

Decomposition on Shallow Nets

- Taylor decomposition of function $f(x_1, \ldots, x_d)$

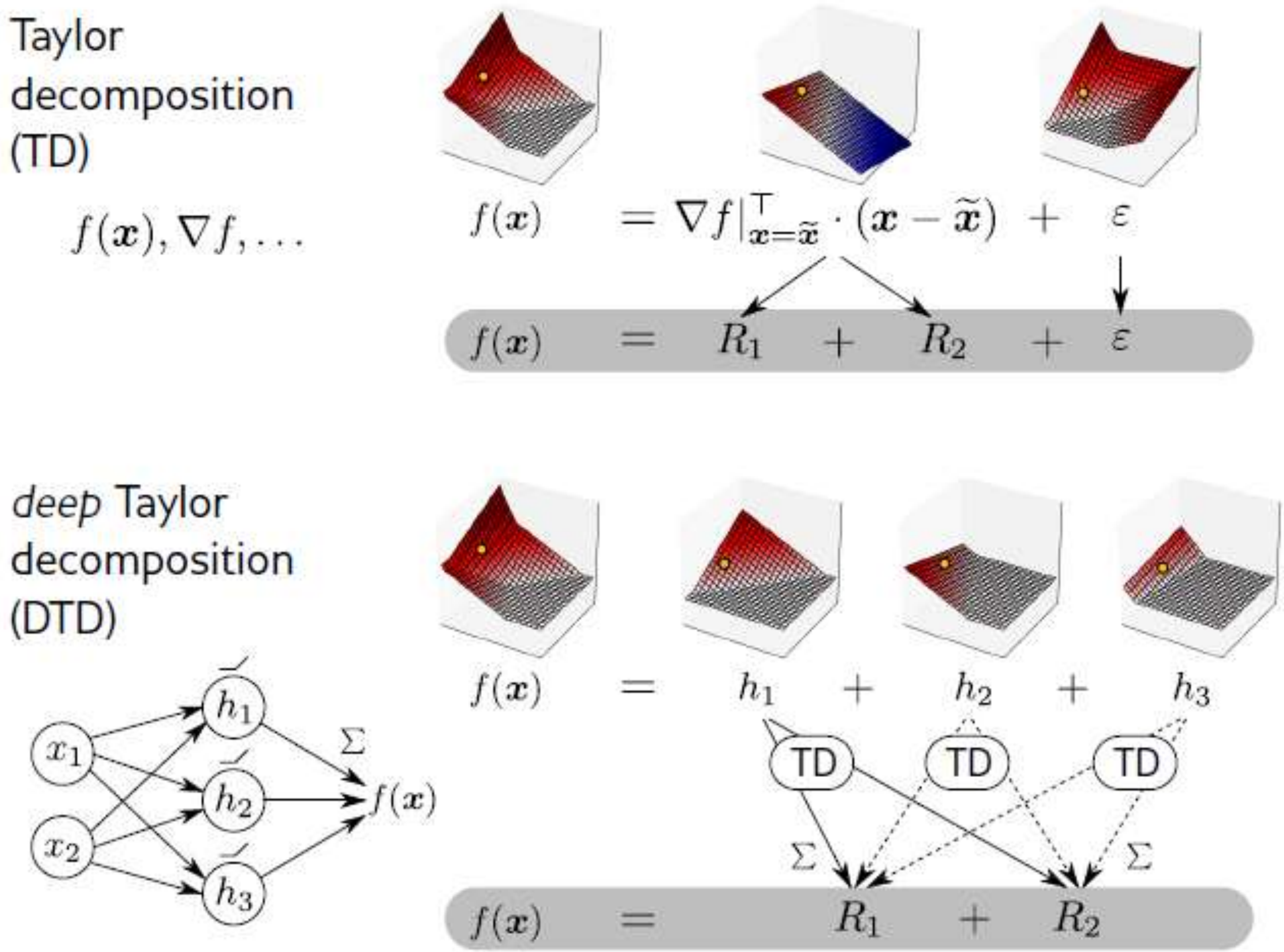
$$f(\mathbf{x}) = \underbrace{f(\widetilde{\mathbf{x}})}_{0} + \sum_{i=1}^{d} \underbrace{\frac{\partial f}{\partial x_i}\Big|_{\mathbf{x}=\widetilde{\mathbf{x}}} (x_i - \widetilde{x}_i)}_{R_i} + \underbrace{O(\mathbf{x}\mathbf{x}^\top)}_{0}$$

- Can it be applied on Deep Learning?
  - Doesn't work well on DNN
  - Also subjected to gradient noise

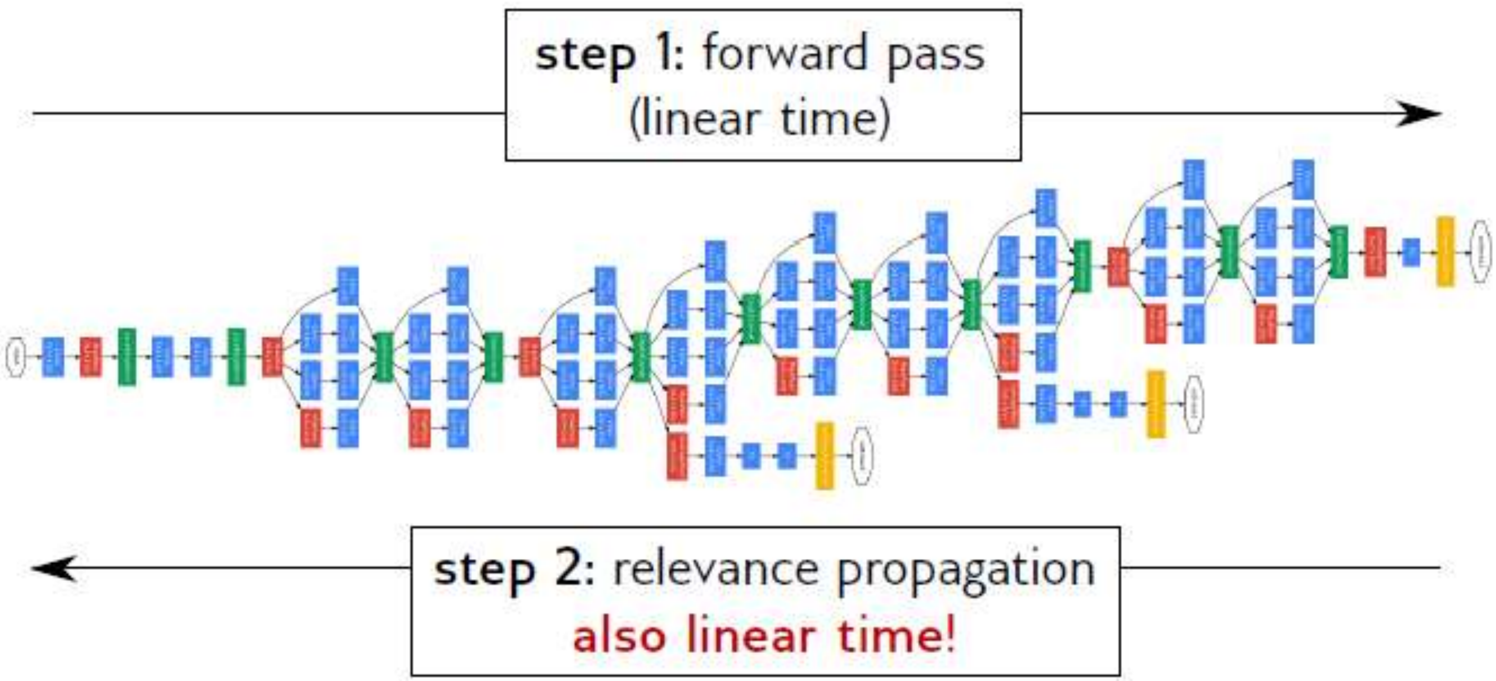
[ICASSP 2017 Tutorial]

[Montavon et al. 2017]

Deep Taylor Decomposition



Layer-Wise Relevance Propagation (LRP)



Propagation rule:

$$R_i = \sum_j q_{ij} R_j \qquad \sum_i q_{ij} = 1$$

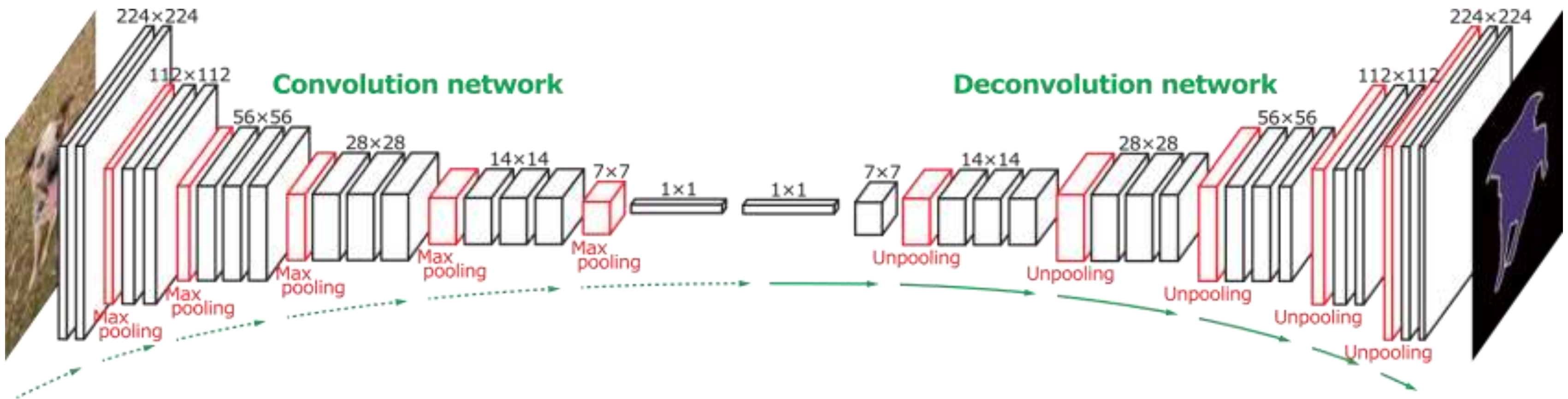
DeepLIFT

- DeepLIFT explains the difference in output from some ‘reference’ output in terms of the difference of the input from some ‘reference’ input.
- The ‘reference’ input represents some default or ‘neutral’ input that is chosen according to what is appropriate for the problem at hand
- **Activation difference** propagated down to input
- Capable to propagate relevance down even when the gradient is zero. (solves saturation problem)

[Zeiler and Fergus ECCV'14]
[Noh et al. ICCV'15]

DeConvNet

- Outputs **probability map** that indicate probability of each pixel belonging to one of the classes



- Convolution Network extract features
- Deconvolution Network generate probability map (same size as the input)

Figure from [Noh et al. ICCV'15]

Summary – What We Have Discussed

- Interpretable ML
- Agonistics methods
- Model-specific methods
- Interpretability in deep learning

Discussion – Current Limitations

- What we have not discussed
  - Interpretable recurrent neural nets
  - Interpretable reinforcement learning
  - Interpretable unsupervised learning models

Reference

- G. Montavon, W. Samek, and K. Müller, "Methods for interpreting and understanding deep neural networks," *Digit. Signal Process.*, vol. 73, pp. 1–15, 2018.
- W. Samek, G. Montavon & K.-R. Müller "Tutorial on Methods for Interpreting and Understanding Deep Neural Networks." ICASSP 2017 Tutorial.
- David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert MÃžller. How to explain individual classification decisions. volume 11, pages 1803–1831, 2010.
- Wojciech Samek, Alexander Binder, Gregoire Montavon, Sebastian Lapuschkin, and Klaus Robert Muller. 2016. Evaluating the Visualization of What a Deep Neural Network Has Learned. *IEEE Transactions on Neural Networks and Learning Systems*: 1–13.
- Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus Robert Müller. 2017. Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recognition* 65, August 2016: 211–222.
- Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. volume 10, page e0130140, 2015.

Reference cont.

- Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In *Workshop at International Conference on Learning Representations*, 1–8.
- Korattikara A, Rathod V, Murphy K, Welling M. Bayesian Dark Knowledge. arXiv preprint arXiv:150604416. 2015;.
- Zachary C Lipton. 2016. The Mythos of Model Interpretability. *ICML Workshop on Human Interpretability in Machine Learning*.
- D. Erhan, Y. Bengio, A. Courville, and P. Vincent. Visualizing higher-layer features of a deep network. Technical Report 1341, University of Montreal, Jun 2009.
- M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. CoRR, abs/1311.2901v3, 2013.
- Q. Le, M. Ranzato, R. Monga, M. Devin, K. Chen, G. Corrado, J. Dean, and A. Ng. Building high-level features using large scale unsupervised learning. In Proc. ICML, 2012.
- Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. 2016. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *29th Conference on Neural Information Processing Systems (NIPS 2016)*, 1–29.
- Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning Important Features Through Propagating Activation Differences. In *CVPR*.

Thank you!

Omics Data + Clinical Data

Interpretable **Integrative**

Machine Learning

Better Healthcare

Prior Bio-clinical Knowledge

https://leesael.github.io/