An Introduction to Interpretable Machine Learning

Sael Lee
BK Associate Professor
SNU Computer Science and Engineering
Seoul National University, Korea
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

Lee Sael - SNU
Class Prototypes (CP)

“How does a goose typically look like according to the neural network?”

Image from Symonian’13
Activation Maximization (AM)

Interpreting concepts predicted by a deep neural net via activation maximization

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

\[
\begin{align*}
\text{Finding the input pattern that maximizes class probability. } \ p(w|x) \\
\text{Find the most likely input pattern for a given class. } \ p(x|w)
\end{align*}
\]
Data Generation

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

$$\hat{y}^l = \arg \max_{\mathbf{y}^l} \Phi_h \left( G_l(\mathbf{y}^l) \right) - \lambda \Vert \mathbf{y}^l \Vert$$

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.
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
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?”

heatmap = $LRP(x, f)$
Saliency Map Examples

Figure from https://github.com/albermax/innvestigate
Explaining by Sensitivity Analysis

Given prediction function \( f(x_1, x_2, \ldots, x_d) \) on \( d \) dimensional input data \( x = (x_1, x_2, \ldots, x_d) \),

**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

Lee Sael - SNU
Perturbation Approaches

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

- flute: 0.9973
- flute: 0.0007
- Learned Mask
- blur
- constant
- noise
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: \( \sum_i R_i = f(x_1, \ldots, x_d) \)

- 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 $f(x_1, \ldots, x_d)$

$$f(x) = f(\tilde{x}) + \sum_{i=1}^{d} \left. \frac{\partial f}{\partial x_i} \right|_{x=\tilde{x}} (x_i - \tilde{x}_i) + O(xx^\top)$$

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

Taylor decomposition (TD)

\[ f(x), \nabla f, \ldots \]

\[ f(x) = \nabla f \bigg|_{x=\bar{x}} \cdot (x - \bar{x}) + \varepsilon \]

\[ f(x) = R_1 + R_2 + \varepsilon \]

Deep Taylor decomposition (DTD)

\[ f(x) = h_1 + h_2 + h_3 \]

\[ f(x) = \sum_{i=1}^{3} R_i \]

Montavon et al. 2017

Lee Sael - SNU
Layer-Wise Relevance Propagation (LRP)

Propagation rule:

\[ R_i = \sum_j q_{ij} R_j \quad \sum_i q_{ij} = 1 \]
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)
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

Reference cont.

Thank you!

Omics Data + Clinical Data

Interpretable  Integrative
Machine Learning

Prior Bio-clinical Knowledge

Better Healthcare

https://leesael.github.io/