2021/1/22 이유진 (youjin lee)

Paper Review

Learning how to explain neural networks: PatternNet and PatternAttribution

Pieter-Jan Kindermans, Kristof T. Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, Sven Dähne

2017

Abstract

DNNs를 더 잘 이해하기 위해 발명된 DeConvNet(DCN), Guided BackProp(GBP), LRP는 수만개의 파라미터를 포함한 다층 네트워크에 적용이 가능하지만 그보다 단순한 선형 모델에서 적절하게 작동하지 않는다. 우리는 신경망을 위한 설명법들이 단순성의 한계를 극복하여 선형모델에서도 신뢰성 있게 작동해야 한다고 주장한다. 선형 모델 분석에 근거하여 우리는 이론적으로 선형 모델에 적용할 수 있고 DNNs에 대한 개선된 설명을 생산할 수 있는 두 개의 설명법(PatternNet, PatternAttribution)을 산출하는 일반화를 제안한다.

Introduction

Deep learning은 여러 분야에 걸쳐 두각을 보여왔고 최근 신경망 분류기들은 데이터에 있는 다른 모든 관련 없는 분산 구성 요소를 필터링하고 제거하여 입력 데이터 포인트에 포함된 관련 신호(예: 고양이의 존재)를 매우 잘 탐지하게 되었다. 관련 신호와 방해 요소와의 분리는 수만개의 파라미터와 비선형 활성화 함수를 지닌 많은 레이어를 통해 입력을 거쳐 얻어지고, 이 모델들은 신호의 매우 응축된 결과를 산출한다(예: 이미지에 고양이가 있을 확률). 분류기 결정을 설명하는 방법들은 분류기를 통해 응축된 출력 신호를 다시 전파하여 입력에서 해당 관련 신호가 어떻게 인코딩 되었는지 보여줄 수 있다는 가정 하에 작동한다. 간단히 말해 만약 분류기가 고양이를 감지했다면 시각화는 네트워크 관점에서 입력 이미지의 고양이-관련 영역을 가리켜야 한다. 이런 관점에서 DCN, GBP, LRP, Deep Taylor Decomposition(DTD), Integrated Gradients, SmoothGrad가 발명되었다.

우리는 다른 관점으로 접근했다. 우선 가장 간단한 인공 신경망을 세팅하고 그 맥락에서의 설명 방법을 분석한다: 전적으로 선형 모델과 선형 생성 모델로부터 비롯된 데이터. 이 단순한 설정은 (1) 신호와 방해 요소가 입력 데이터내에서 어떻게 인코딩 되는지를 완전히 제어 가능하며 (2) 결과 설명이 알려진 신호 구성 요소와 어떻게 관련이 있는지 분석적으로 추적할 수 있게 해준다.

이 분석은 비선형 모델까지 이어지는 현재 설명법의 단점을 강조해준다. 우리는 이 연구 결과에 근거하여 그 결함들을 완화시켜줄 PatternNet과 PatternAttribution을 제안한다. 또한 이 방법을 실제 관련 네트워크와 데이터셋에 적용하고 정성적으로 개선된 신호 시각화와 속성을 생산함을 보여주었다(Fig. 2 & Fig. 4b). 게다가 해당 이론 모형이 실증적으로 유지되는지 실험적으로 검증했다(Fig. 3).

Notation and Scope

분석적 측면에서 우리는 네트워크의 선형 뉴런들은 선택적으로 ReLU, max-pooling, softmax만을 따른다. 또한 모든 뉴런들은 각각 weight vector를 가진다고 본다. 나머지 제약조건들은 이전의 DCN, GBP, LRP 등의 논문과 유사하다. 명확성을 높이기 위해 bias들은 모두 일정한 뉴런으로 간주한다.

Understanding Linear Models

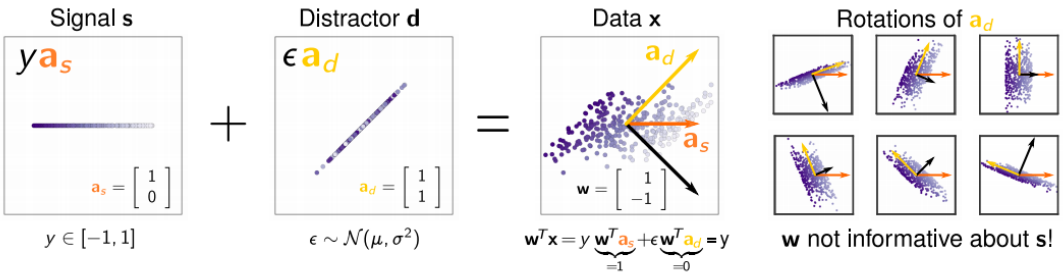


Figure 2: For linear models, i.e., a simple neural network, the weight vector does **not** explain the signal it detects Haufe et al. (2014). The data $\boldsymbol{x} = y\boldsymbol{a}_s + \epsilon\boldsymbol{a}_d$ is color-coded w.r.t. the output $y = \boldsymbol{w}^T\boldsymbol{x}$. Only the signal $s = y\boldsymbol{a}_s$ contributes to y. The weight vector $\boldsymbol{w}$ does not agree with the signal direction, since its primary objective is canceling the distractor. Therefore, rotations of the basis vector $\boldsymbol{a}_d$ of the distractor with constant signal $\boldsymbol{s}$ lead to rotations of the weight vector (**right**).

DNNs을 살펴보기 전에 선형 모델을 먼저 분석해본다(Fig. 2). 데이터 x가 주어졌을 때 예제는 다음과 같다.

$$x = s + d \qquad s = a_s y,\ with\ a_s = (1,0)^T,\quad y \in [-1,1]$$
$$d = a_d \epsilon,\ with\ a_d = (1,1)^T,\quad \epsilon \sim \mathcal{N}(\mu, \sigma^2)$$

우리는 x로부터 y를 추출하기 위해 선형 회귀 모델을 학습하고, 구조적으로 s는 우리 데이터의 신호(y에 대한 정보를 포함하는 x의 부분 영역)를 의미한다. 방해자 d는 탐색 작업을 더욱 어렵게 만들며 y를 최적으로 추출하기 위해서 d를 걸러낼 수 있어야만 한다. 이것이 가중치 벡터를 필터라고도 부르는 이유이며 예제에서는 $w = (1,1)^T$가 이 작업을 수행한다. 이 예제를 통해 우리는 최적화 가중치 벡터 w가 일반적으로 신호 방향과 나란하지는 않지만 방해자의 영향을 줄이려 함을 관찰할 수 있다. 또한 이는 w가 방해자 $w^T d = 0$에 직교할 때 최적으로 해결될 수 있음으로 오른쪽 그림과 같이 방해자 a_d의 방향이 바뀌었을 때 w도 역시 바뀌어야 한다. 반면 신호 a_s 방향의 변화는 $w^T a_s = 1$인 w의 부호와 크기의 변화로 보상될 수 있기에 방향은 여전히 유지된다.

신호와 방해자를 가지고 있는 상황일 때, 선형 모델의 가중치 벡터 방향은 방해자에 의해 결정됨을 의미한다. 또한 가중치 벡터만으로 입력의 어떤 부분이 출력 y를 생성했는지 알 수 없음을 암시한다. 이는 방향 a_s이고 데이터로부터 학습해야 한다. 여기서 우리는 위의 선형 문제가 convex하므로 모델을 최적화하여 얻은 w가 위에서 정의한 분석 솔루션에 수렴된다는 점에 유의해야한다.

이번에는 방해자가 없는 대신 방향이 없는 Gaussian noise를 가지고 있다고 가정한다. Bias의 변화로 보상할 수 있기에 noise의 평균이 0인 경우만 고려하면 되는데 해당 noise는 상관 관계나 구조를 포함하지 않기 때문에 유일한 방법은 다른 측정기준을 평균화하는 것이다. 이는 w를 조절하여 효과적으로 해결하기는 어렵지만 Gaussian noise를 더하는 것은 w를 축소시키고 L2 regularization의 효과를 가진다.

위의 추론을 고려할 때 우리는 어떤 조건하에 DNNs가 작동하는지 의문을 가져야만 한다. 특히 DCN, GBP는 기울기를 사용하여 crisp한 시각화를 생성하기에 더욱 그렇다. 그러므로 우리는 우리의 이론을 DNNs에 적용하여 다음과 같은 정량적이고 정성적인 실험을 수행한다.

- Fig.3에서 우리는 가중치 벡터이나 학습된 방향이 VGG16 내의 모든 단일 뉴런의 입력에 대한 정보 내용을 얼마나 잘 포착했는지 평가했다. 이 실험은 가중치 벡터로 정의된 방향보다 학습된 방향이 더 많은 정보를 가지고 있음을 보여주었고, 이는 우리가 대부분 방해자 영역에서 작동하고 있음을 의미한다.
- 본 실험은 Fig. 4a의 이미지 품질 저하 실험을 통해 확인되었다
- 또한 Fig.1, Fig. 4b, Fig.5에서 시각화의 질적 검사에 의해 확증되었다.

Overview of Explanation Approach and Their Behavior

이번 절에서 개별적인 분류기 결정에 대한 설명 방법을 살펴보고 이전 절의 선형 모델 분석과 어떻게 연결되어 있는지 논의한다. Fig. 1에서는 function, signal, attribution visualization으로 나눌 수 있는 다양한 유형의 설명 방법에 대한 개요를 제공한다. 이 세 그룹은 모두 네트워크에 대해 서로 다른 정보를 제시하고 보완한다.

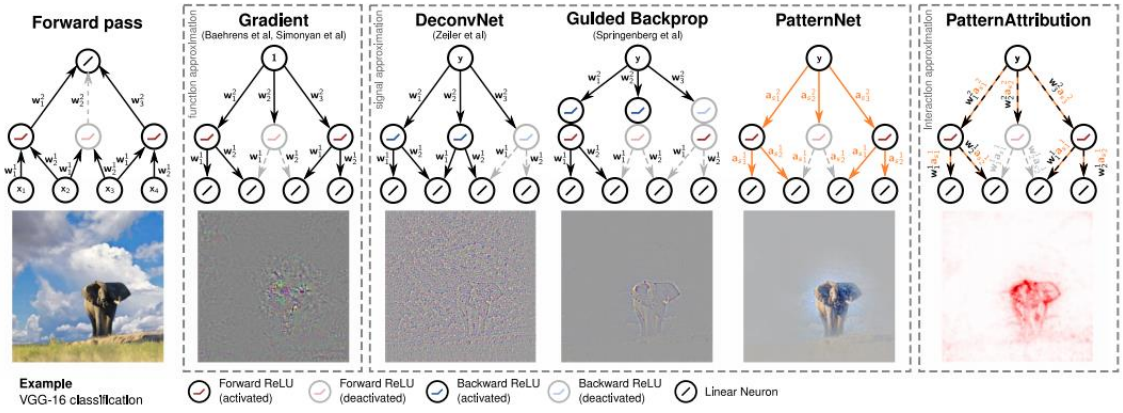


Figure 1: Illustration of explanation approaches. Function and signal approximators visualize the explanation using the original color channels. The attribution is visualized as a heat map of pixel-wise contributions to the output

Functions – gradients, saliency map

입력 공간에서 함수를 설명하는 것은 모델이 x로부터 y를 추출하는데 사용하는 연산을 설명하는 것과 같다. DNNs는 매우 비선형하기에 추정만 가능하다. Saliency map은 입력 공간에서 특정 방향에 따라 움직이는 것이 모델 기울기에 의해 방향이 주어지는 y에 얼마나 영향을 주는지(민감도 분석) 추정한다.

Signal – DeConvNet, Guided BackProp, PatternNet

신경망에 의해 감지된 신호는 네트워크 활성화를 야기하는 데이터의 구성 요소이다. 이 방법의 목표는 입력 픽셀 공간에서 다시 매핑하여 어떤 입력 패턴이 원래 feature maps에서 주어진 활성화를 야기했는지 보여주는 것이다.

선형 모델에서 신호는 $s = a_s y$이고, 패턴 a_s는 신호 방향을 포함한다. 이 신호들을 시각화하기 위한 방법으로는 DCN과 GBP가 있고, 이들은 saliency map과 같은 알고리즘을 사용했지만 rectifiers를 각각 다르게 다루었다. DCN은 rectifiers를 forward pass에서 제외시키는 대신 각각 deconvolution한 후에 추가적인 ReLU를 취한 반면 GBP는 추가적인 ReLU를 취했을 뿐만 아니라 forward pass에도 포함시켰다. 네트워크의 선형 구성 요소에 대한 back-projections은 각 뉴런의 신호 방향이라 추정되는 것의 중첩에 해당된다. 이런 이유로 이 projections은 입력 공간의 재구성이 아닌 상위 레이어 뉴런을 활성화시킨 feature들의 근사치로 봐야한다.

신경망이 가장 단순할 경우 – 선형 모델 – 이 시각화는 기울기로 축소된다. 이는 필터 w 그리고 패턴 a_s나 신호 s도 보여주지 않는다. 따라서 DCN, GBP는 선형 모델에 감지된 신호를 생성하는 것을 보장할 수 없다. 이 방법들은 억지로 시각화를 생성하기 때문에 우리는 나중에 필터 w의 방향과 신호 s의 방향이 일치하는지 확인하고 다름을 증명한 후 새로운 방법인 PatternNet을 제안할 것이다.

Attribution – LRP, Deep Taylor Decomposition, PatternAttribution

우리는 레이어를 거쳐 출력에 신호 dimension이 얼마나 기여했는지 볼 수 있는데, 이를 attribution이라 한다. 선형 모델

에서 최적의 attribution은 가중치 벡터 w와 신호의 원소별 곱셈으로 얻을 수 있다. 픽셀 단위 기여도의 분해(relevance라 불림)하는 방법 LRP, 위 방법을 확장시킨 것으로 입력에서 기여한 측면에서의 뉴런의 활성화를 분해하는 방법인 DTD가 있다. 이는 1차 테일러 확장을 사용했고, 선택된 출력 뉴런 i의 관련성은 forward pass로부터 출력으로 초기화된다. 레이어 l에서 뉴런 i의 관련성은 다음과 같이 재분배된다.

$$r_i^{output} = y, \qquad r_{j\neq i}^{output} = 0, \qquad r_i^{l-1} = \frac{w \odot (x - x_0)}{w^T x} r_i^l$$

여기서 우리는 forward pass로부터 비활성 ReLU 유닛이 backward pass에서 재분배를 중지하기 때문에 $w^T x > 0$라고 가정할 수 있다. 이는 ReLU가 기울기의 전파를 어떻게 멈추게 하는지와 동일하다. DTD 적용의 어려운 점은 많은 선택의 여지가 있는 root point x_0의 선택이다. x_0의 선택은 방해자 $x_0 = d$를 추정하는 것에 해당하며, 그에 따라 신호 $\hat{s} = x - x_0$를 인식하는 것은 중요하다. PatternAttribution은 DTD의 확장으로 데이터로부터 root point를 어떻게 정할 것인지를 학습한다.

Learning to Estimate the Signal

Function을 시각화하는 것은 단순한 반면 signal, attribution의 시각화는 더 어렵다. 이는 신호가 무엇이고 방해자가 무엇인지에 대한 추정을 필요로 하기 때문이다. 이번 절에서 우리는 뉴런-별 신호 추정기에 대한 품질 measure을 제안한다. 이를 통해 기존 방법들을 평가하고 이 기준을 최적화하는 신호 추정기을 얻는다. 또한 이렇게 얻은 추정기는 PatternNet과 PatternAttribution에 사용될 것이다.

Quality Criterion for Signal Estimators

입력 데이터 $x = s + d$는 신호와 방해자로 구성되어 있고 신호는 출력에 기여하나 방해자는 그렇지 않다. 필터 w가 y를 추출하기에 충분히 잘 학습되어 있다고 가정하면, 우리는 아래의 식을 가진다.

$$w^T x = y, \quad w^T s = y, \quad w^T d = 0$$

이러한 조건만으로 신호를 추정하는 것은 불량 설정 문제임을 유의해야한다. 우리는 $w^T u \neq 0$이 아닌(수직이 아닌) 랜덤 벡터 u를 가지는 $\hat{s} = u(w^T u)^{-1}$형태인 선형 추정기로 한정할 수 있다. 이 경우 신호는 $w^T \hat{s} = y$를 만족하는 $\hat{s} = u(w^T u)^{-1} y$로 추정된다. 이는 DCN에 대한 무한한 back-projections뿐 아니라 DTD의 무한히 많은 수의 규칙들이 존재함을 의미한다. 해당 문제를 완화하기 위해 우리는 $\hat{d} = x - S(x)$와 $y = w^T x$를 사용하여 분명한 분산과 공분산들로 작성될 신호 추정기 $S(x) = \hat{s}$에 대해 다음과 같은 품질 measure ρ를 도입했다.

$$\rho(S) = 1 - \max_v corr\left(w^T x, v^T\big(x - S(x)\big)\right) = 1 - \max_v \frac{v^T cov[\hat{d}, y]}{\sqrt{\sigma_{v^T\hat{d}}^2 \sigma_y^2}} \qquad (1)$$

이 기준은 선형 투영을 사용하여 나머지 $x - \hat{s}$로부터 얼마나 많은 정보가 재구성될 수 있는지 측정하여 추가적인 제약 조건을 도입한 것이다. 가장 좋은 신호 추정기는 나머지에서 정보 대부분을 제거하므로 큰 $\rho(S)$ 값을 산출한다. 상관 관계는 스케일과 관계없이 변하지 않기 때문에 우리는 $v^T \hat{d}$를 분산 $\sigma_{v^T\hat{d}}^2 = \sigma_y^2$로 제한한다. 고정된 $S(x)$에 대한 최적값 v를 찾는 것은 $\hat{d}$로부터 y까지의 least-squares regression이다.

Existing Signal Estimators

[S_x – the identity estimator] 신호 추정에 대한 가장 기본적인 접근 방법으로 전체 데이터가 신호라고 가정하고 방해자가 없다고 가정하는 것이다. $S(x) = x$. 선형 모델에서 이는 attribution로써 $r = w \odot x$에 해당한다. 즉, 전체 네트워크에서 가정된 신호가 단순히 원래 이미지임을 의미하고, 또한 데이터에 방해자가 있을 경우 모두 attribution에 속함을 의미한다. $r = w \odot x = w \odot s + w \odot d$.

[S_w – the filter based estimator] DCN과 GBP에 의해 만들어진 암묵적인 가정에 따르면 인지된 신호는 가중치 벡터의 방향에 따라 달라진다는 것이다. 이 가중치 벡터는 유효한 신호 추정기가 되기 위해 정규화 되어야 한다. DTD에서 이는 $w^2 - rule$에 해당되며 다음과 같은 신호 추정기를 생성한다. $S_w(x) = \frac{w}{w^T w} w^T x$.

PatternNet and PatternAttribution

우리는 이전에 제안한 기준에 최적화하여 데이터로부터 신호 추정기 S를 학습하는 것을 제안했다. 신호 추정기 S는 모든 가능한 $v: \forall v, cov[y, \hat{d}]v = 0$에 대해 상관 관계가 0일 때 식(1)에 대하여 최적화된다. 공분산의 선형성 때문에 $\hat{d} = x - S(x)$이므로 위의 조건은 아래와 같다.

$$cov[y, \hat{d}] = 0 \;\Rightarrow\; cov[x, y] = cov[S(x), y] \qquad (2)$$

공분산이 요약 통계임을 인지하는 것은 중요하고 결과적으로 문제는 다양한 방식으로 해결할 수 있다. 여기서 우리는 이 문제에 대해 가능한 두 가지 방법을 제시할 것이며, 추정기가 최적화할 때 출력 y와 공존하지 않기 때문에 bias 뉴런의 기여는 0으로 간주된다.

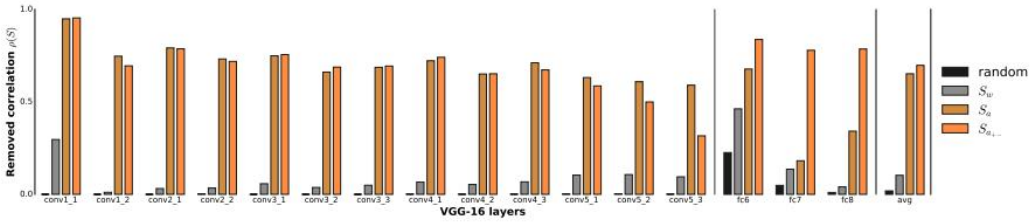
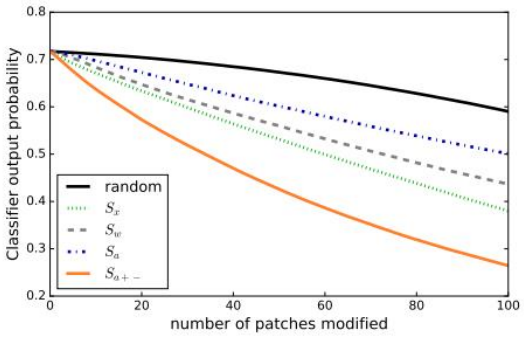
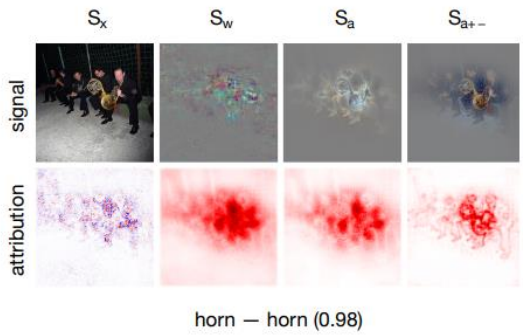


Figure 3: Evaluating $\rho(S)$ for VGG-16 on ImageNet. Higher values are better. The gradient ($S_{\boldsymbol{w}}$), linear estimator ($S_{\boldsymbol{a}}$) and nonlinear estimator ($S_{\boldsymbol{a}_{+-}}$) are compared. An estimator using random directions is the baseline. The network has 5 blocks with 2/3 convolutional layers and 1 max-pooling layer each, followed by 3 dense layers.



(a) Image degradation experiment on all 50.000 images in the ImageNet validation set. The effect on the classifier output is measured. A steeper decrease is better.



(b) Top: signal. Bottom: attribution. For the trivial estimator S_x the original input is the signal. This is not informative w.r.t. how the network operates.

< S_a – The linear estimator >

선형 뉴런은 입력 x로부터 선형 신호 s만 추출할 수 있다. 그러므로 우리는 신호 추정기를 산출하는 s와 y사이의 선형 의존도를 추정할 수 있다.

$$S_a(x) = aw^T x. \qquad (3)$$

식(3)을 식(2)에 대입하여 최적화하면 $cov[x, y] = cov[aw^T x, y] = acov[y, y] \Rightarrow a = \frac{cov[x,y]}{\sigma_y^2}$ 이다. 이 솔루션은 다른

derivation임에도 불구하고 neuro-imaging에서 일반적으로 사용되는 방법과 동일하다는 점에 주목한다. 이 방법으로는 앞서 보인 선형 예제에 적용 가능하고 방해자가 신호와 직교하는 경우에만 필터 기반 접근법과 동일하다. 우리는 선형 추정기가 컨볼루션 레이어에 잘 작동함을 알아냈지만 밀도가 높은 레이어에서 ReLU와 함께 이 신호 추정기를 사용하면 여전히 방해 구성 요소와 상당한 상관 관계를 가지게 된다(Fig. 3).

< $S_{a_{+-}}$ – The two-component estimator >

선형 신호 추정기를 넘어서기 위해서 학습에 rectifier가 어떻게 영향을 미치는지 이해하는 것은 중요하다. ReLU의 특징때문에 가중치는 $y > 0$인 방해 구성 요소 뉴런으로 걸러낼 필요가 있다. 이를 통해 신경망이 지역적인 필터 적용을 허용하므로 글로벌한 방해 구성 요소는 추정할 수 없다. 우리는 양과 음 영역 사이를 구분할 필요가 있다.

$$x = \begin{cases} s_+ + d_+, & if\ y > 0 \\ s_- + d_-, & otherwise \end{cases}$$

음의 영역 내의 신호와 방해자가 다음 ReLU에 의해 사라지더라도 신호를 근사화하기 위해 여전히 이렇게 구별할 필요가 있다. 그렇지 않으면 뉴런이 사용되었는지에 대한 정보가 방해자 내에서 남아있을 것이고, 따라서 우리는 two-component estimator를 제안한다.

$$S_{a_{+-}}(x) = \begin{cases} a_+ w^T x, & if\ w^T x > 0 \\ a_- w^T x, & otherwise \end{cases} \quad (4)$$

다음, a_+와 a_-의 패턴에 대한 표현을 나타낸다. 우리는 각각 양과 음의 영역에서 x에 대한 기댓값을 $\mathbb{E}_+|x|$, $\mathbb{E}_-|x|$라고 표현한다. π_+을 $w^T x > 0$인 입력 x일 예상 확률이라고 하면 데이터/신호와 출력의 공분산은 아래와 같다.

$$cov[x, y] = \pi_+(\mathbb{E}_+[xy] - \mathbb{E}_+[x]\mathbb{E}[y]) + (1 - \pi_+)(\mathbb{E}_-[xy] - \mathbb{E}_-[x]\mathbb{E}[y]) \quad (5)$$

$$cov[s, y] = \pi_+(\mathbb{E}_+[sy] - \mathbb{E}_+[s]\mathbb{E}[y]) + (1 - \pi_+)(\mathbb{E}_-[sy] - \mathbb{E}_-[s]\mathbb{E}[y]) \quad (6)$$

공분산이 모두 같다고 가정하면 식(2)를 사용하여 양과 음의 영역 각각을 신호 추정기에 최적화하기 위해 다룰 수 있다.

$$\mathbb{E}_+[xy] - \mathbb{E}_+[x]\mathbb{E}[y] = \mathbb{E}_+[sy] - \mathbb{E}_+[s]\mathbb{E}[y]$$

이를 식(4)에 대입하여 a_+에 대해 해결하면 필요한 파라미터가 산출된다.

$$a_+ = \frac{\mathbb{E}_+[xy] - \mathbb{E}_+[x]\mathbb{E}[y]}{w^T\mathbb{E}_+[xy] - w^T\mathbb{E}_+[x]\mathbb{E}[y]} \quad (7)$$

$S_{a_{+-}}$를 위한 솔루션은 입출력 간의 관계가 선형일 때 선형 추정기로 감소하므로 앞의 선형 예제를 제대로 해결한다.

< PatternNet and PatternAttribution >

본 논문의 분석을 바탕으로 우리는 PatternNet과 PatternAttribution을 제안한다. PatternNet은 입력 공간에 추정된 신호의 층별 back-projection을 산출한다. 신호 추정기는 각 레이어에서 뉴런별 비선형 신호 추정기 $S_{a_{+-}}$의 중첩으로 근사된다. 이는 backward pass하는 동안 네트워크의 가중치가 유익한 방향으로 대체되는 기울기 계산과 같다. PatternAttribution은 attribution $w \odot a_+$을 노출시키고 LRP를 개선한 것이다. 또한 DTD의 root point 추정기로도 볼 수 있다. 여기서 설명은 분류 점수에 대한 추정된 신호의 뉴런별 기여로 구성된다. 방해자를 무시함으로써 PatternAttribution은 noise를 줄이고 heatmaps 더 명확히 생성할 수 있다.

Experiments and Discussion

설명 품질을 평가하기 위해 우리는 이미지 분류 과제에 초점을 맞췄다. 그럼에도 불구하고 우리는 이미지에만 국한되지 않고 Theano, Lasagne에도 위 방법을 활용하여 구현했으며, 본 연구에서는 널리 알려진 ImageNet 데

이터셋에 VGG-16 모델을 구현하여 분석을 제한하였다.

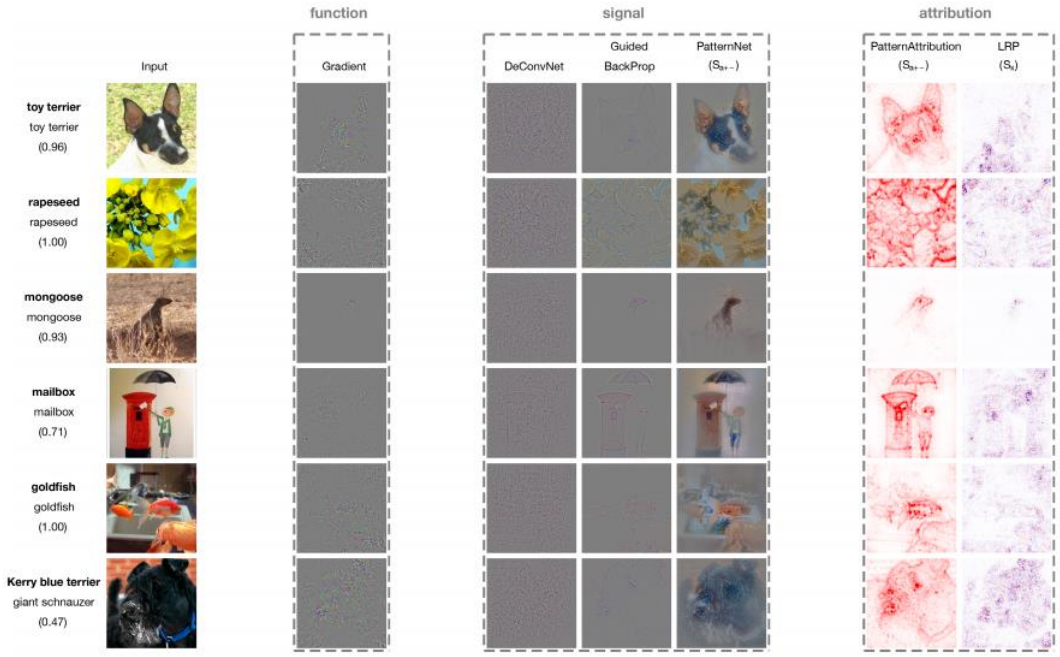


Figure 5: Visualization of random images from ImageNet (validation set). In the leftmost shows column the ground truth, the predicted label and the classifier's confidence. Methods should only be compared within their group. PatternNet, Guided Backprop, DeConvNet and the Gradient (saliency map) are back-projections to input space with the original color channels. They are normalized using $x_{norm} = \frac{x}{2\max|x|} + \frac{1}{2}$ to maximize contrast. LRP and PatternAttribution are heat maps showing pixel-wise contributions. Best viewed in electronic format (zoomed in). The supplementary contains more samples.

[Measuring the quality of signal estimators]

Fig.3을 보면 높은 값이 더 나은 상관 관계 측정값 $\rho(x)$로부터 결과를 제시한다. 우리는 기준 신호 추정기로 무작위로 방향을 정했으며 분명히 이는 상관관계를 거의 제거하지 못했다. 필터 기반 추정기는 첫번째 레이어의 정보 일부를 제거하는데 성공했으며 이는 해당 레이어의 패턴과 필터가 유사함을 의미한다. 그러나 상위 레이어에서 기울기는 훨씬 적은 정보를 제거했고 전반적으로 무작위로 방향을 정한 추정기보다 좋은 성과를 보이지 못했다. 즉, 이는 가중치가 신경망에서 감지된 자극과 일치하지 않음을 의미하므로 DCN, GBP가 만든 신호를 통한 가정은 유효하지 않음을 의미한다.

[Image degradation]

첫 실험은 개별적인 뉴런들의 신호 추정기의 품질을 직접 측정하는 것이었고, 다음 실험은 전체 네트워크를 고려하여 간접적으로 품질을 측정하는 것이었다. 우리는 attribution에 할당된 순서를 근거로 점점 더 많은 patch를 손상시키며 처음 선택된 클래스에 대한 예측 결과(softmax 레이어를 거쳐 나온 값)가 어떻게 변화하는지 측정했다. 결과는 Fig.4a를 통해 patch가 무작위로 정렬되었을 때 최악임을 알 수 있다. 선형 최적화된 추정기 S_a는 상당히 낮은 성능을 지녔고, 그 다음으로 필터 기반 추정기 S_w, 그 다음 trivial 신호 추정기 S_x가 미약하게 더 나은 성능을 보였다. 그러나 $S_{a_{+-}}$는 원래 예측에 대한 confidence가 가장 큰 폭으로 빠르게 감소함을 알 수 있고, 이에 대한 성능은 시각화에 의해 뒷받침된다.

[Qualitative evaluation]

Fig. 5에서 ImageNet에서 무작위로 선택된 이미지 6장에 대한 시각화를 보여준다. PatternNet은 DCN, GBP와는 대조적으로 추가적인 rectifier를 포함하지 않고도 원래와 가까운 신호를 복구할 수 있다. 우리는 이를 패턴의 최적화가 입력 공간에서 중요한 방향을 포착할 수 있게 해주기 때문이라 주장한다. 이는 기존에 존재하는 방법들과 대조되며 전반적으로 해당 방법이 다른 방법에 비해 가장 명확한 시각화를 보여준다.

Conclusion

기계학습 측면에서 비선형 방법을 이해하고 설명하는 것은 중요한 과제이다. 비선형 모델을 시각화하는 알고리즘이 등장했지만 이론적인 기여는 부족하다. 우리는 모델 gradient의 방향이 반드시 데이터 내 신호에 대한 추정이 아님을 입증했다. 대신 이는 방해 잡음 기여와 신호 방향 간 상관관계를 반영한다(Fig. 2). 이는 기존 신경망에 대한 일반적인 설명 방법이 단순한 선형 모델에서는 정확한 설명을 제공하지 않는다는 것을 의미한다. 그래서 우리는 뉴런별 설명을 위해 객관적인 함수를 제안했고 이는 비선형 모델로도 확장될 수 있다. 또한 데이터 분포를 고려하여 신호 시각화 방법인 PatternNet과 분해 방법 PatternAttribution으로 최적화될 수 있다.