1주차: Deep Sparse Rectifier Neural Networks

논문 링크: https://proceedings.mlr.press/v15/glorot11a/glorot11a.pdf

활성화 함수 ReLU에 관련된 논문

Abstract

rectifying neurons가 기존의 생물학적 뉴런과 tanh 활성화 함수보다 좋은 성능을 보임.

비선형성, 미분 불가능성 성질

비지도학습 사전 학습의 유무에 따른 성능 차이 감소

1. **Introduction**

computational neuroscience 모델과 ml neural network 모델을 ReLU 활성 함수 max(0,x)로 연결

rectifier(=hinge) activation function

vertex → 원시 모형 → 복잡한 시각적 모형 인식

Deep network

2006 Deep Belief Network(DBN): feature extraction, dimensionality reduction에 사용

2007 비지도 학습으로 각 층 초기화하여 일반화

tanh와 시그모이드 함수 대안으로서 비선형 특성을 가진 ReLU 함수 제안

L1 규제로 sparsity 만들고 가중치 발산 문제 해결

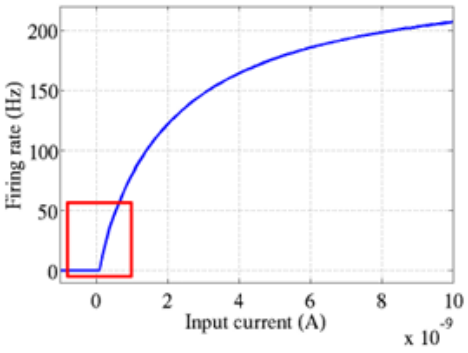
unsupervised pretraining 없이 좋은 성능을 보임.

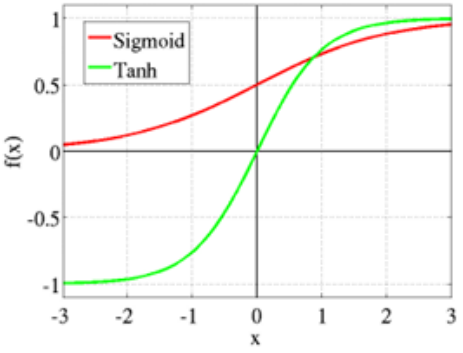
2. **Background**

2.1 Neuroscience observations

antisymmetric / symmetric

뉴런들은 1~4% 활성화 vs L1 페널티 없는 neural net





차이점: non-linear 한 구간 존재 여부

Tanh는 0, sigmoid 함수는 0.5에서 antisymmetry vs 뉴런은 변곡점 없음.

2.2 Advantages of Sparsity

기존 활성 함수에서는 non-zero activation 수행 → ReLU 함수로 Sparsity 구현 가능

Sparsity 중요성

- Information disentangling: 변수를 설명하는 데이터 분리 목표. Dense 할 때는 어떠한 변화에 거의 모든 벡터의 변화가 생기지만 Sparse 할 때 작은 변화는 특정 부분에 영향을 준다.
- Efficient variable-size representation: active 뉴런 수를 다양화해 효율적으로 차원과 정밀도 제공
- Linear separability: sparse 한 상태에서는 정보가 고차원 공간에 표현됨.
- Distributed but sparse: 0이 아닌 특성 수에 따라 뛰어난 성능 및 효율성

3. **Deep Rectifier Networks** ⭐

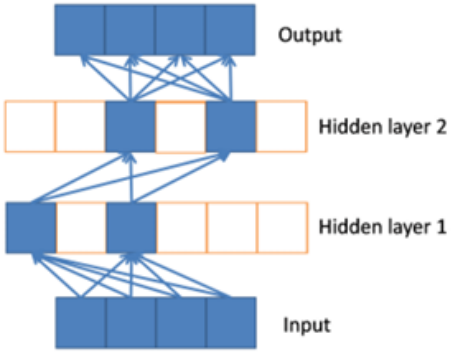
3.1 Rectifier Neurons

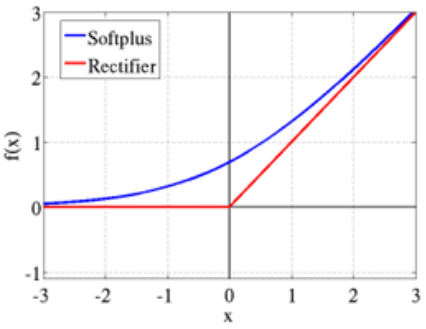
rectifier (x) = max (0,x)

one-sided ⇒ not symmetry and not antisymmetry

❓ 2개의 rectifier unit을 사용하면 symmetry antisymmetry 성질 이용 가능 y=|x| 이런 그래프? 어떤 상황에 사용되지?

- advantages
  - 쉽게 sparsity 얻어 수학적 계산 이득





  - 비선형적 특성은 처음 경로 선택에 있어 일부 집합만 활성화될 때 발생. 초기 레이어 이후에는 선형적 모델 적용
    - chatgpt 설명)

      *ReLU 함수는 입력이 **음수일 때 비선형적**입니다. 입력값이 음수일 때 출력이 항상 0이 되므로, 이 구간에서 모델이 특정 뉴런을 '꺼버리는' 효과가 나타납니다. 이로 인해 ReLU는 일부 입력 값에 대해 비선형적으로 동작하게 됩니다.*
  - 시그모이드 함수나 tanh 함수처럼 기울기 소실 문제 발생하지 않음. 계산 편리
- Potential Problems
  - 0이 기울기 back propagation 방해 ⇒ softplus(x) = log(1+e^x) 로 보완 (smoothing)
  - unbounded 한 문제 ⇒ L1 페널티로 추가적 sparsity 유발
  - parameterization 문제 ⇒ 기울기와 bias 에 가중치 부여해 해결

3.2 Unsupervised Pre-training

stacked denoising auto encoder 에 rectifier 활성 함수 사용하기 어려움.

임계값 아래 값들과 unbounded 값들 처리 어려움.

Linear reconstruction function: $f(x, theta) = Wdecmax(Wenc * x + benc, 0) + bdec$

a. softplus activation function: 실제 값과 softplus 값에 변형된 값을 넣어 값이 차이 제곱

$$L(x, \theta) = ||x - \log(1 + \exp(f(\tilde{x}, \theta)))||^2$$

b. unbounded 문제 해결 위해 0과 1 사이 값이 나오도록 (미분한) sigmoid 함수 활용

$$\begin{aligned} L(x, \theta) &= -x \log(\sigma(f(\tilde{x}, \theta))) \\ &\quad -(1 - x) \log(1 - \sigma(f(\tilde{x}, \theta))) \end{aligned}$$

c. linear activation function 활용 후 제곱

d. rectifier activation function 활용 후 제곱

4. Experimental Study

4.1 Image Recognition

MNIST, CIFAR10, NISTP, NORB 이미지 분류 관련 데이터셋으로 실험

Neuron	MNIST	CIFAR10	NISTP	NORB
With **unsupervised pre-training**				
Rectifier	**1.20%**	**49.96%**	**32.86%**	**16.46%**
Tanh	**1.16%**	**50.79%**	35.89%	17.66%
Softplus	**1.17%**	**49.52%**	**33.27%**	19.19%
Without **unsupervised pre-training**				
Rectifier	**1.43%**	**50.86%**	**32.64%**	**16.40%**
Tanh	1.57%	52.62%	36.46%	19.29%
Softplus	1.77%	53.20%	35.48%	17.68%

실험 조건:

with unsupervised pre-training: 3개 은닉층. 각 층에 1000개 유닛. tanh 함수에서는 cross entropy 활용

0으로 변형될 확률 1/4. 학습률 일정.

without unsupervised pre-training: 앞과 같은 학습률 적용. NLL 사용 (-logP(correct class|input)). 미니 배치 10

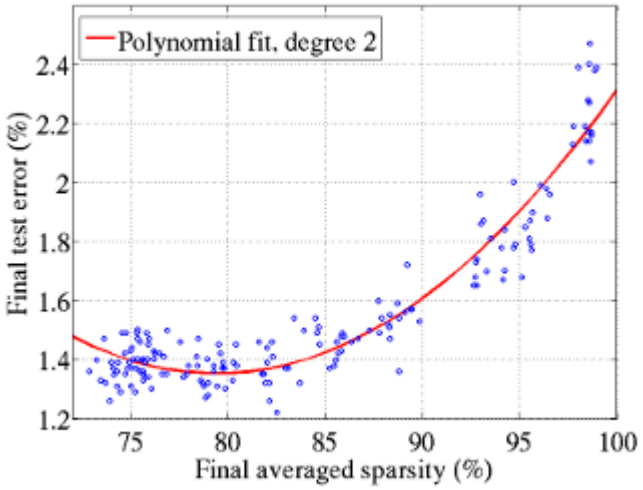
결과:

임계점 0이 있음에도 local minima 찾을 수 있도록 훈련됨. 트레이드 오프가 없어 일관적으로 ReLU 모델이 우수함 증명

❓ softplus는 rectifier함수를 보완한 함수인데 성능이 왜 나쁜 것인지...

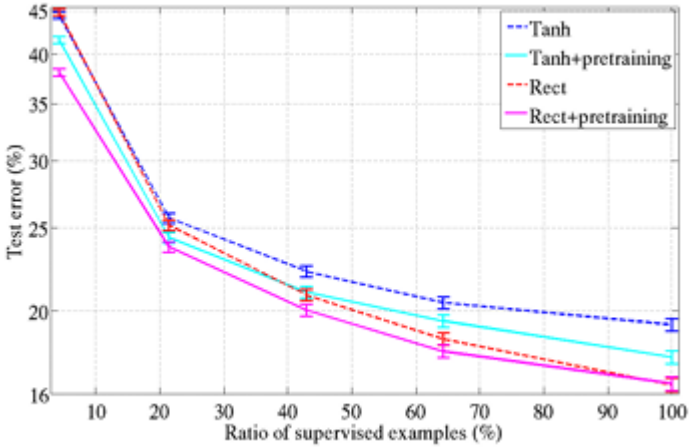
unsupervised pre-training을 했을 때 오차에 있어 다른 함수들과 큰 차이가 나지 않음

Unsupervised pre-training을 하지 않았을 때 다른 함수들에 비해 명백하게 큰 차이로 오차가 감소됨을 확인할 수 있다.



sparsity 70~85%까지 비슷한 오차를 보임.

Semi-supervised 실험 조건: 라벨링된 training set 비율 조정



⇒ Semi-supervised 데이터에 대해서는 pre-training이 확실한 성능 향상이 보임

4.2 Sentiment Analysis

positive nagative 분류 혹은 별점 분류

data: 식당 리뷰 5 star RMSE 로 평가

데이터 가공: bag of words에서 이진 벡터로 변환 ⇒ 매우 sparse 한 데이터 생성됨. 은닉층 1개 혹은 3개. 10 fold cross validation. L1 페널티 적용

결과:

Network	RMSE	Sparsity
No hidden layer	0.885 ± 0.006	99.4% ± 0.0
Rectifier (1-layer)	0.807 ± 0.004	28.9% ± 0.2
Rectifier (3-layers)	**0.746** ± 0.004	53.9% ± 0.7
Tanh (3-layers)	0.774 ± 0.008	00.0% ± 0.0

은닉층 많을수록 RMSE 작아짐(좋은 결과)

❓ layer가 늘어났을 때 Sparsity 가 어떻게 증가한 건지 궁금

ReLU 활성화 함수는 Sparse 데이터에 성능이 좋아 텍스트 데이터에서 좋은 성능을 보임

5. Conclusion

Rectifier unit은 unsupervised pre-training이 있을 때와 없을 때의 차이가 적게 만들어 줌

50~80%의 sparsity에서 모델이 잘 생성됨

(Sparsity: 활성화되지 않은 비율 정도로 생각. 뇌의 sparsity는 95~99%)

텍스트 분석, 감성 분석에도 효과적

gradient 0 존재, 파라미터라이징 문제는 L1 페널티 적용해 보완