https://github.com/multicore-it/n

Lot ELJIN Littes: Ilgithub.com/multicum.com/

1. 튜닝 I

https://github.com/multicore-lt/r

인공신경망투병^T 개유

Activation Linear Output Input Function Function W * X + bSigmoid Х Υ [심층 인공신경망] [Tuning Point] 입력 값 전처리 **(1)** Input Cost (2) 비용함수 **Function** Activation (3) 알고리즘 1,://github.com/multicore-it/r/ **Function (4**) 가중치 초기화 Initializer Optimizer (Back Propagation) **(5)** Optimizer 알고리즘 **(6)** 노드 개수 Node (7) 래이어 개수 Laver Node

Layer

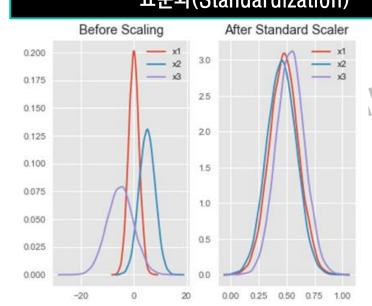
1 -

Multicore-it/rl

上の位移場の無場でのe-itlnh https://github.son無場工

전처리 입력데이터

표준화(Standardization)



Scaling | Pic credits: Google

- 표준정규분포를 구해서
- 데이터에 일반적으로 분포를 가진 ______ 기법을 적용

https://github.com/multicore-it/r/

인공신경망·튜닝^{Iticore-itlr}. https://github.eore-itlr

입력데이터 전처리

정규화(Normalization)

• 벡터의 크기(혹은 길이)를 측정하는 방법(혹은 함수)

노름 (Norm)

$$||\mathbf{x}||_{\mathsf{p}} := \left(\sum_{i=1}^n |\mathbf{x}_i|^p\right)^{1/p}$$

·n: 원소의 개수 ·x: 두 벡터 사이의 거리

· L1 Norm: p가 1 · L2 Norm: p가 2 (default)

- 정규화는 벡터의 각 원 소 를 노 름 으 로 나누어준 것
- 원소의 절대 값이 1보다 작은 값으로 모두 7.com/multicore-it/r/

정규화 (Normalization) • 크기를 1로 규격화하기 위해 벡터를 노름으로 나눈 것

Normalization(x) = x / ||x||



입력데이터 전처리

```
인공신경망·튜닝T 입력도

Impy as np
tplotlib.pyplot as nlt
import numby as no
import matplotlib.pyplot as plt
def normalize(x):
                            ① 벡터의 노름 구하기
   norm = np.linalg.norm(x)
                                                                                                     정규화(Normalization)
   if norm == 0:
       return x
                                 벡터의 정규화
   return x / norm
x = np.random.normal(3, 2.5, size=(2, 100))
                                         ③ 가우시안 분포 데이터 생성
x_norm = normalize(x) # 데이터 점규화
                                                                            random
                                                                                                                    normalized
plt.figure(figsize=(10.10))
plt.subplot(221)
                                                          15.0
plt.scatter(x[0], x[1], color="orange")
plt.title("random")
                                                          12.5
plt.subplot(222)
                                                                                                   0.20
plt.scatter(x_norm[0], x_norm[1], color="blue")
                                                          10.0
plt.title("normalized")
                                                                                                    0.15
                                                           7.5
                                                                                                   0.10 -
                                                           5.0
                                                                                                    0.05
                                                           2.5
                                                           0.0
                                                                                                   0.00
                                                          -2.5
                                                                                                   -0.05
```

-2.5

2.5

5.0

7.5

10.0

-0.05

0.00

0.05

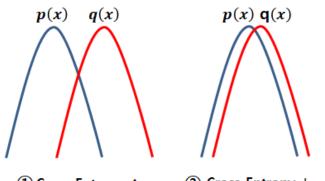
0.10

上 6 也 易 导 用 出 的 H no - https://github.sore-it/n 비용함수

교차 엔트로피

교차 엔트로피 (Cross Entropy)

- 이산 확률 분포 $H(p, q) = -\sum_{x \in X} p(x) \log q(x)$
- 연속 확률 분포 $H(p, q) = -\int_x p(x) \log q(x)$



- 1 Cross Entropy ↑
- (2) Cross Entropy ↓

- 정보를 최적으로 인코딩 필요한 비트 수
- 엔트로피: 두 확률 분포가 틀릴 양을 고려해서 정 보 s://github.com/multicore-it/r/ 최적으로 표현되기 위해 필요한



인공신경망·튜닝^{lticore-itl'} 화서차 https://github.sor

활성화 알고리즘

오차 역전파 과정

예측 (정방향)

각 래이어 별, 노드 별 출력 계산



노드 별 출력과 실제 값의 차이 계산



역방향으로 각 래이어, 별 노드 별 오차 기여도 분석



경사하강법을 통해 오차가 줄어드는 방향으로 노드 별 가중치 조정

(Error 계산해서 오차가 줄어드는 방향으로 네트워크를 거꾸로 올라가면서 가중치를 조정하는 과정



인공신경망·튜닝^{lti}l Hticore-it/11

활성화 알고리즘

그래디언트 소실과 폭주

- 가중치가 너무 작게 설정되거나 활성화 함수의 출력 구간이 입력 값보다 너무 작게 기울기 소실 현상이 발생
- 설정되어 있거나 활성화 함수의 출력 구간이 너무 크게 구성되어 있으면 기울기 폭주 현상이 발생

기울기 소실 (Vanishing Gradient)

입력 층으로 갈수록 기울기가 급격히 감소하는 현상

기울기 폭주 (Exploding Gradient)

입력 층으로 갈수록 기울기가 급격히 폭증하는 현상

https://gilling





인공신경망·튜닝^{Iticore-itli} https://github.eor

활성화 알고리즘

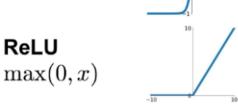
활성화 함수

Sigmoid

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

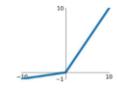
tanh

tanh(x)



Leaky ReLU

 $\max(0.1x,x)$



Maxout

 $\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU



전 (Error Backpropagation): 계산해서 오차가 줄어드는 방향으로 올라가면서 네트워크를 거꾸로 가중치를 조정하는 과정

