Back Propagation Algorithm - Numerical Solved

Sujan Karna · Follow
3 min read · Dec 27, 2023

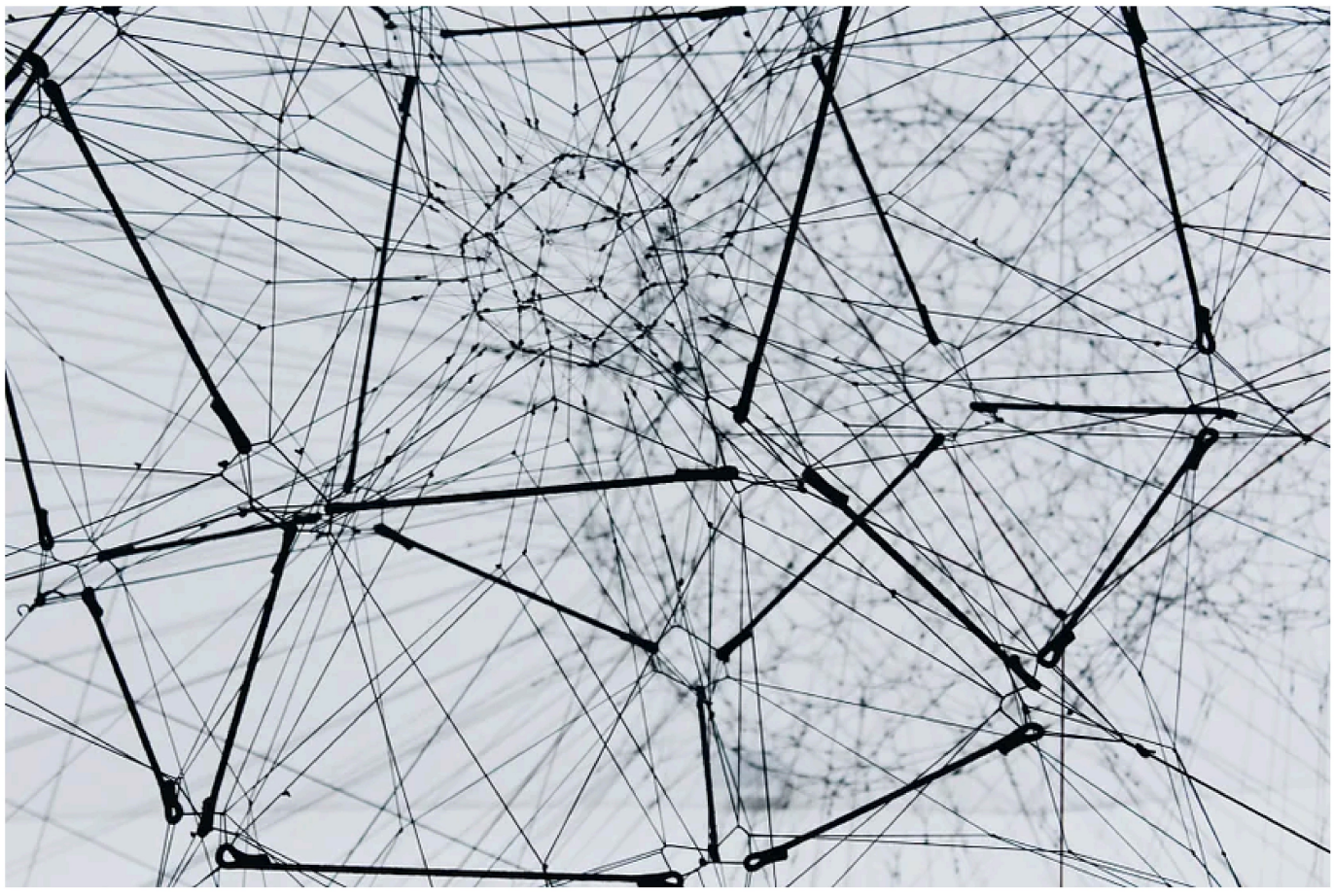
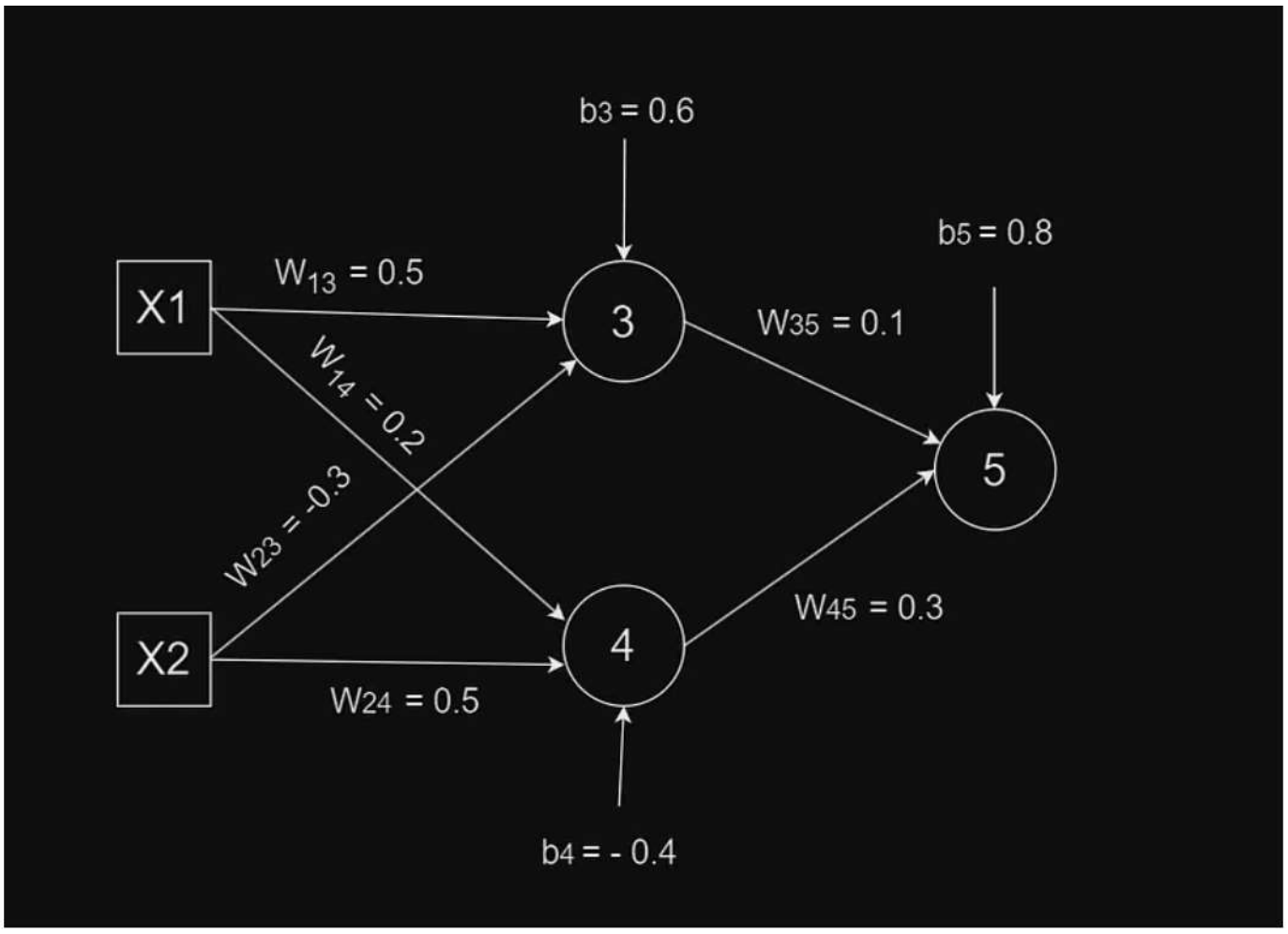


Photo by Alina Grubnyak on Unsplash

Q. Consider a *multilayer feed-forward neural network* given below. Let the *learning rate* be *0.5*. Assume initial values of weights and biases as given in the table below. Train the network for the training tuples *(1, 1, 0)* and *(0, 1, 1)*, where last number is target output. Show weight and bias updates by using *back-propagation algorithm*. Assume that *sigmoid activation function* is used in the network.

W13	W14	W23	W24	W35	W45	b3	b4	b5
0.5	0.2	-0.3	0.5	0.1	0.3	0.6	-0.4	0.8

Weight and Bias values



Neural Network

Solution:

Step 1: Initialization

a. Inputs and Outputs for 2 iterations:

X1	X2	Output(τ)
1	1	0
0	1	1

Inputs and Outputs

b. Initial weight and bias values:

W13	W14	W23	W24	W35	W45	b3	b4	b5
0.5	0.2	-0.3	0.5	0.1	0.3	0.6	-0.4	0.8

Initial weight and bias values

c. Learning rate (l) = 0.5

d. Activation function = Sigmoid function

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

Sigmoid Function

Step 2: The termination condition for our algorithm, according to the question, is when we complete 2 iterations.

Iteration 1

Input and Output values:

X1	X2	Output(τ)
1	1	0

Input and output for Iteration 1

Weight(W) and bias(b) values:

W13	W14	W23	W24	W35	W45	b3	b4	b5
0.5	0.2	-0.3	0.5	0.1	0.3	0.6	-0.4	0.8

Weight and bias values for iteration 1

a. Calculate the input(I) and output(O) for each layer

-For input layer

$O_1 = I_1 = 1$

$O_2 = I_2 = 1$

Input and outputs of Input layer

-For hidden layer

$I_3 = O_1 \times W_{13} + O_2 \times W_{23} + b_3$

$= 1 \times 0.5 + 1 \times (-0.3) + 0.6$

$= 0.5 - 0.3 + 0.6$

$= 0.8$

$$O_3 = \frac{1}{1+e^{-I_3}} = \frac{1}{1+e^{-0.8}} = 0.69$$

$I_4 = O_1 \times W_{14} + O_2 \times W_{24} + b_4$

$= 1 \times 0.2 + 1 \times 0.5 + (-0.4)$

$= 0.2 + 0.5 - 0.4$

$= 0.3$

$$O_4 = \frac{1}{1+e^{-I_4}} = \frac{1}{1+e^{-0.3}} = 0.5744$$

Inputs and outputs for hidden layer

-For output layer

$$I_5 = O_3 \times W_{35} + O_4 \times W_{45} + b_5$$
$$= 0.69 \times 0.1 + 0.5744 \times 0.3 + 0.8$$
$$= 0.0690 + 0.1723 + 0.8$$
$$= 1.0413$$

$$O_5 = \frac{1}{1+e^{-I_5}} = \frac{1}{1+e^{-1.0413}} = 0.7391$$

Input and output of output layer

b. Calculation of error at each node:

$$Err_j = O_j(1 - O_j)(T_j - O_j)$$

Error calculation formula for output layer

- ***For output layer:***

$Err_5 = O_5 (1 - O_5) (\tau - O_5)$
$= 0.7391 (1 - 0.7391) (0 - 0.7391)$
$= 0.7391 (0.2609) (- 0.7391)$
$= - 0.1425$

Error calculation for output layer

$$Err_j = O_j(1 - O_j) \sum_k Err_k w_{jk}$$

Error calculation formula for hidden layer

- ***For hidden layer:***

$Err_4 = O_4 (1 - O_4) (Err_5 \times W_{45})$
$= 0.5744 (1 - 0.5744) (- 0.1425 \times 0.3)$
$= 0.5744 (0.4256) (- 0.04275)$
$= - 0.010451$

$Err_3 = O_3 (1 - O_3) (Err_5 \times W_{35})$
$= 0.69 (1 - 0.69) (- 0.1425 \times 0.1)$
$= 0.69 (0.31) (- 0.1425)$
$= - 0.030481$

c. Update the weight

$$W_{ij}(new) = W_{ij}(old) + l \times O_i \times Err_j$$

Formula for new weight calculation

$$W_{45}(new) = W_{45}(old) + l \times O_4 \times Err_5 = 0.3 + 0.5 \times 0.5744 \times (-0.1425) = 0.2591$$
$$W_{35}(new) = W_{35}(old) + l \times O_3 \times Err_5 = 0.1 + 0.5 \times 0.69 \times (-0.1425) = 0.0508$$

$$W_{24}(new) = W_{24}(old) + l \times O_2 \times Err_4 = 0.5 + 0.5 \times 1 \times (-0.010451) = 0.494775$$
$$W_{14}(new) = W_{14}(old) + l \times O_1 \times Err_4 = 0.2 + 0.5 \times 1 \times (-0.010451) = 0.194774$$

$$W_{23}(new) = W_{23}(old) + l \times O_2 \times Err_3 = -0.3 + 0.5 \times 1 \times (-0.030481) = -0.315240$$
$$W_{13}(new) = W_{13}(old) + l \times O_1 \times Err_3 = 0.5 + 0.5 \times 1 \times (-0.030481) = 0.484759$$

Updated weight values

d. Update the bias value

$$b_j = b_{j(old)} + l \times Err_j$$

Formula for new bias calculation

$$b_3 = b_{3(old)} + l \times Err_3 = 0.6 + 0.5 \times (-0.030481) = 0.584759$$
$$b_4 = b_{4(old)} + l \times Err_4 = -0.4 + 0.5 \times (-0.010451) = -0.405225$$
$$b_5 = b_{5(old)} + l \times Err_5 = 0.8 + 0.5 \times (-0.1425) = 0.7288$$

Updated bias values

Iteration 2

Input and output values:

X1	X2	Output(τ)
0	1	1

Input and output for iteration 2

Weight(W) and bias(b) values:

W13	W14	W23	W24	W35	W45	b3	b4	b5
0.4848	0.1948	-0.315	0.4948	0.0508	0.2591	0.5848	-0.405	0.7288

New weight and bias values for iteration 2

a. Calculate the input(I) and output(O) for each layer

-For input layer

$O_1 = I_1 = 0$

$O_2 = I_2 = 1$

Input and output for input layer

-For hidden layer:

$$I_3 = O_1 \times W_{13} + O_2 \times W_{23} + b_3$$
$$= 0 \times 0.4848 + 1 \times (-0.315) + 0.5848$$
$$= 0 - 0.315 + 0.5848$$
$$= 0.2698$$

$$O_3 = \frac{1}{1+e^{-I_3}} = \frac{1}{1+e^{-0.2698}} = 0.567044$$

$$I_4 = O_1 \times W_{14} + O_2 \times W_{24} + b_4$$
$$= 0 \times 0.1948 + 1 \times 0.4948 + (-0.405)$$
$$= 0 + 0.4948 - 0.405$$
$$= 0.0898$$

$$O_4 = \frac{1}{1+e^{-I_4}} = \frac{1}{1+e^{-0.0898}} = 0.522435$$

Input and output for hidden layer

-For output layer:

$$I_5 = O_3 \times W_{35} + O_4 \times W_{45} + b_5$$
$$= 0.567044 \times 0.0508 + 0.522435 \times 0.2591 + 0.7288$$
$$= 0.892969$$

$$O_5 = \frac{1}{1+e^{-I_5}} = \frac{1}{1+e^{-0.892969}} = 0.709502$$

Input and output for output layer

b. Calculation of error at each node:

- ***For output layer:***

$Err_5 = O_5 (1 - O_5) (\tau - O_5)$

$= 0.709502 (1 - 0.709502) (1 - 0.709502)$

$= 0.709502 (0.290498) (0.290498)$

$= 0.059874$

- ***For hidden layer:***

$Err_4 = O_4 (1 - O_4) (Err_5 \times W_{45})$

$= 0.522435 (1 - 0.522435) (0.059874 \times 0.2591)$

$= 0.003871$

$Err_3 = O_3 (1 - O_3) (Err_5 \times W_{35})$

$= 0.567044 (1 - 0.567044) (0.059874 \times 0.0508)$

$= 0.567044 (0.432956) (0.003042)$

$= 0.000747$

Error calculation for hidden and output layer

c. Update the weight

$W_{45}(new) = W_{45}(old) + l \times O_4 \times Err_5 = 0.2591 + 0.5 \times 0.522435 \times 0.059874 = 0.274740$
$W_{35}(new) = W_{35}(old) + l \times O_3 \times Err_5 = 0.0508 + 0.5 \times 0.567044 \times 0.059874 = 0.067776$

$W_{24}(new) = W_{24}(old) + l \times O_2 \times Err_4 = 0.4948 + 0.5 \times 1 \times 0.003871 = 0.496735$
$W_{14}(new) = W_{14}(old) + l \times O_1 \times Err_4 = 0.1948 + 0.5 \times 0 \times 0.003871 = 0.1948$

$W_{23}(new) = W_{23}(old) + l \times O_2 \times Err_3 = -0.315 + 0.5 \times 1 \times 0.000747 = -0.314626$
$W_{13}(new) = W_{13}(old) + l \times O_1 \times Err_3 = 0.4848 + 0.5 \times 0 \times 0.000747 = 0.4848$

New weight calculation

d. Update the bias value

$b_3 = b_{3(old)} + l \times Err_3 = 0.5848 + 0.5 \times 0.000747 = 0.585173$
$b_4 = b_{4(old)} + l \times Err_4 = -0.405 + 0.5 \times 0.003871 = -0.403064$
$b_5 = b_{5(old)} + l \times Err_5 = 0.7288 + 0.5 \times 0.059874 = 0.758737$

Updated bias values

Since we completed 2 iterations, according to question, our calculation stops here.
The updated weight and bias values after second iterations are:

W13	W14	W23	W24	W35	W45	b3	b4	b5
0.4848	0.1948	-0.315	0.4967	0.0678	0.2747	0.5852	-0.403	0.7587

Backpropagation Neural Networks Machine Learning Classification

Data Ware Housing

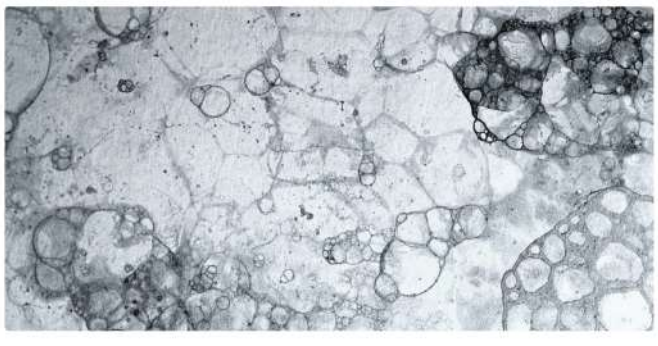
Written by Sujan Karna

Follow

31 Followers

🚀 Passionate Web Developer & Data Science Enthusiast

More from Sujan Karna



Sujan Karna

K-Means Algorithm Solved Numerical

Q. Apply K(=2)-Means algorithm over the data (185, 72), (170, 56), (168, 60), (179,68), (182,7...

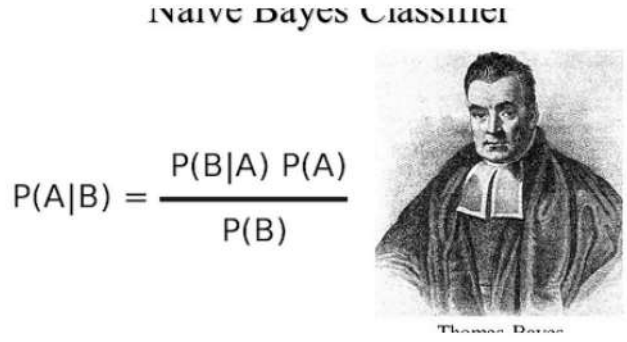
Dec 18, 2023 116 5

Sujan Karna

Density Based- DBSCAN Numerical

Q. Given the points A(3, 7), B(4, 6), C(5, 5), D(6, 4), E(7, 3), F(6, 2), G(7, 2) and H(8, 4), Fin...

Dec 20, 2023 7

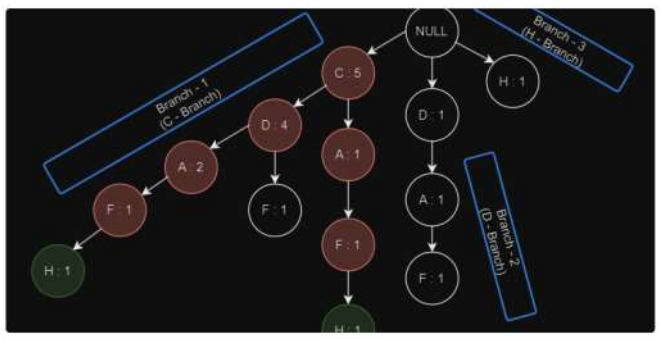


Sujan Karna

Naive Bayesian Classification - Numerical Solved

Q. Consider the following data set.

Jan 7 29 1



Sujan Karna

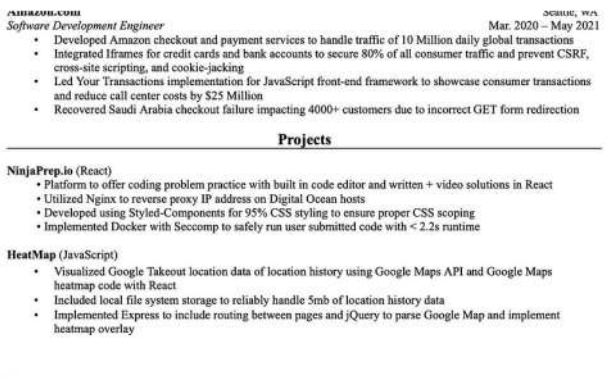
Frequent Pattern Mining-FP Growth Numerical

Q. Generate the frequent pattern from the following data set using FP growth, where...

Jan 11 18 2

See all from Sujan Karna

Recommended from Medium


Amazon.com Seattle, WA
Software Development Engineer Mar. 2020 – May 2021
- Developed Amazon checkout and payment services to handle traffic of 10 Million daily global transactions
- Integrated Iframes for credit cards and bank accounts to secure 80% of all consumer traffic and prevent CSRF, cross-site scripting, and cookie-jacking
- Led Your Transactions implementation for JavaScript front-end framework to showcase consumer transactions and reduce call center costs by $25 Million
- Recovered Saudi Arabia checkout failure impacting 4000+ customers due to incorrect GET form redirection

Projects

NinjaPrep.io (React)
- Platform to offer coding problem practice with built in code editor and written + video solutions in React
- Utilized Nginx to reverse proxy IP address on Digital Ocean hosts
- Developed using Styled-Components for 95% CSS styling to ensure proper CSS scoping
- Implemented Docker with Seccomp to safely run user submitted code with < 2.2s runtime

HeatMap (JavaScript)
- Visualized Google Takeout location data of location history using Google Maps API and Google Maps heatmap code with React
- Included local file system storage to reliably handle 5mb of location history data
- Implemented Express to include routing between pages and jQuery to parse Google Map and implement heatmap overlay

Alexander Nguyen in Level Up Coding

The resume that got a software engineer a $300,000 job at Google.

1-page. Well-formatted.

Jun 1 16.5K 257

Ansa Baby in Tech & TensorFlow

Mathematics Behind Support Vector Machines (SVM) Algorithm

A Detailed Guide to Understand the Mathematics Behind SVM

Jul 24 52 1

Lists

Predictive Modeling w/ Python
20 stories · 1428 saves

Practical Guides to Machine Learning
10 stories · 1725 saves

Natural Language Processing
1625 stories · 1191 saves

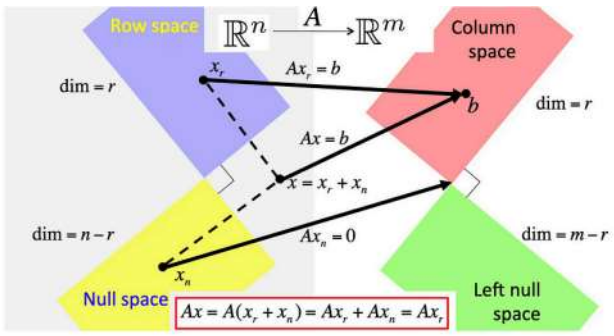
The New Chatbots: ChatGPT, Bard, and Beyond
12 stories · 436 saves

Thrisha Aib

BACKPROPAGATION ALGORITHM

Demystifying Backpropagation: The Backbone of Neural Network Training

Apr 17 1

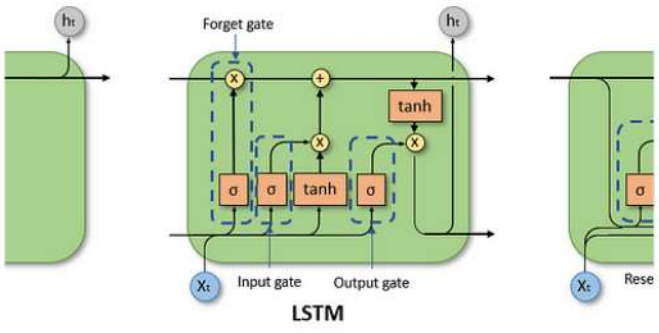


Joseph Robinson, Ph.D. in Towards AI

The Fundamental Mathematics of Machine Learning

A Deep Dive into Vector Norms, Linear Algebra, Calculus

Jul 26 447 5

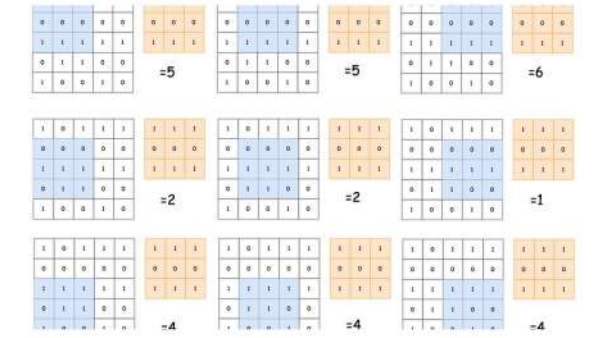


Jonte Dancker in Towards Data Science

A Brief Introduction to Recurrent Neural Networks

An introduction to RNN, LSTM, and GRU and their implementation

Dec 26, 2022 624 7



Ransaka Ravihara

CNN Basics Explained: Beginner's Guide

A simple introduction to convolution neural network

Feb 13 1

See more recommendations