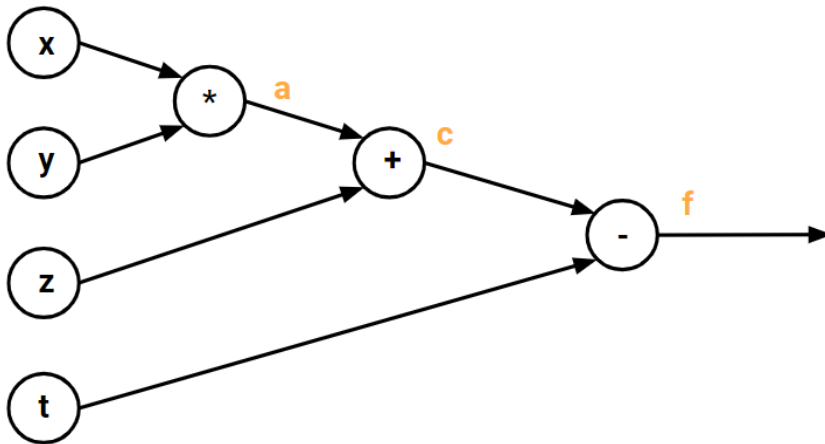


Computational Graph

Bài tập

1. Computational Graph là đồ thị có hướng hay vô hướng?
 - A. Đồ thị có hướng
 - B. Đồ thị vô hướng
2. Thuật toán được sử dụng trong Computational Graph là?
 - A. Thuật toán forward pass
 - B. Thuật toán backward pass
 - C. Gradient Descent
 - D. A và B
 - E. A, B và C
3. Cho đồ thị $f(x, y, z, t)$ như hình, biết $a = x * y$, $c = a + z$, $f = c - t$



Áp dụng chain rule (quy tắc dây chuyền). Đạo hàm của $f'(x) = ?$

- A. $\frac{df}{dx} = \frac{df}{dc} \cdot \frac{dc}{da} \cdot \frac{da}{dy}$
 - B. $\frac{df}{dx} = \frac{df}{dx} \cdot \frac{df}{dy} \cdot \frac{df}{dz} \cdot \frac{df}{dt}$
 - C. $\frac{df}{dx} = \frac{df}{dc} \cdot \frac{dc}{da} \cdot \frac{da}{dx}$
 - D. $\frac{df}{dx} = \frac{df}{dc} \cdot \frac{dc}{da} \cdot x$
4. Computational graph được chia thành mấy loại?
 - A. 1
 - B. 2
 - C. 3
 - D. 4

5. Cho biết kết quả của đoạn code sau:

```
x = torch.tensor([3.], requires_grad=True)
y = torch.tensor([10.], requires_grad=True)
z = torch.tensor([5.], requires_grad=True)
t = x * y
f = t + z

f.backward()

print(x.grad, y.grad, z.grad)
```

- A. tensor([10.]) tensor([3.]) tensor([1.])
- B. tensor([3.]) tensor([10.]) tensor([5.])
- C. tensor([1.]) tensor([3.]) tensor([10.])
- D. tensor([10.]) tensor([3.]) tensor([5.])

Đáp án

1. Computational Graph là đồ thị có hướng hay vô hướng?

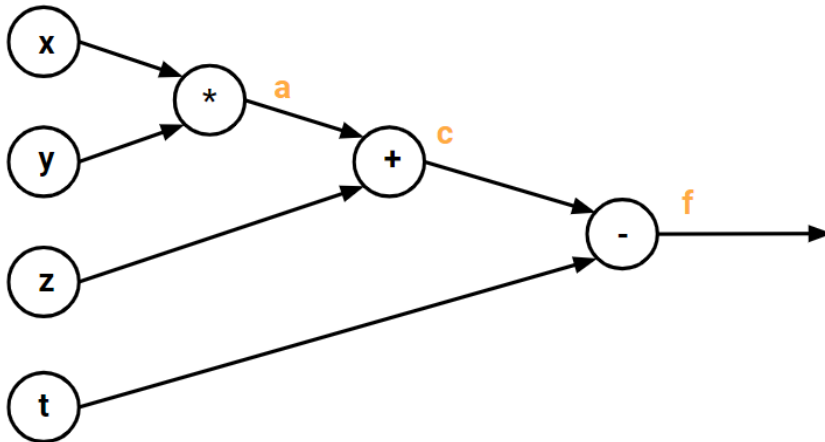
- A. Đồ thị có hướng
- B. Đồ thị vô hướng

2. Thuật toán được sử dụng trong Computational Graph là?

- A. Thuật toán forward pass
- B. Thuật toán backward pass
- C. Gradient Descent
- D. A và B
- E. A, B và C

- **Giải thích:** Computational graph sử dụng thuật toán forward pass để tính các phép toán theo hướng của đồ thị (theo chiều mũi tên của các cạnh), và thuật toán backward pass để tính đạo hàm từng node (theo chiều ngược lại của mũi tên của cạnh). Còn thuật toán gradient descent chỉ là thuật toán được dùng để cập nhật trọng số weight và bias trong bài toán Machine Learning.

3. Cho đồ thị $f(x, y, z, t)$ như hình, biết $a = x * y$, $c = a + z$, $f = c - t$



Áp dụng chain rule (quy tắc dây chuyền). Đạo hàm của $f'(x) = ?$

- A. $\frac{df}{dx} = \frac{df}{dc} \cdot \frac{dc}{da} \cdot \frac{da}{dy}$
- B. $\frac{df}{dx} = \frac{df}{dx} \cdot \frac{df}{dy} \cdot \frac{df}{dz} \cdot \frac{df}{dt}$
- C. $\frac{df}{dx} = \frac{df}{dc} \cdot \frac{dc}{da} \cdot \frac{da}{dx}$
- D. $\frac{df}{dx} = \frac{df}{dc} \cdot \frac{dc}{da} \cdot x$

- **Giải thích:** Đây là lý thuyết.
Các bạn có thể hiểu rõ bằng link sau:
https://nttuan8.com/bai-2-logistic-regression/#Chain_rule

4. Computational graph được chia thành mấy loại?

- A. 1
- B. 2**
- C. 3
- D. 4

- **Giải thích:** static computational graph (tĩnh) và dynamic computational graph (động)
 Các bạn có thể tham khảo ở link sau:
<https://www.geeksforgeeks.org/computational-graphs-in-deep-learning/>

5. Cho biết kết quả của đoạn code sau:

```
x = torch.tensor([3.], requires_grad=True)
y = torch.tensor([10.], requires_grad=True)
z = torch.tensor([5.], requires_grad=True)
t = x * y
f = t + z

f.backward()

print(x.grad, y.grad, z.grad)
```

- A. tensor([10.]) tensor([3.]) tensor([1.])**
- B. tensor([3.]) tensor([10.]) tensor([5.])
- C. tensor([1.]) tensor([3.]) tensor([10.])
- D. tensor([10.]) tensor([3.]) tensor([5.])

- **Giải thích:**

```
x = torch.tensor([3.], requires_grad=True) # khởi tạo x = 3
y = torch.tensor([10.], requires_grad=True) # khởi tạo y = 10
z = torch.tensor([5.], requires_grad=True) # khởi tạo z = 5
t = x * y
f = t + z

f.backward() # tính đạo hàm của hàm f với các tensor có thuộc tính
requires_grad=True

print(x.grad, y.grad, z.grad)
```

- Áp dụng chain rule:

$$\frac{df}{dx} = \frac{df}{dt} \cdot \frac{dt}{dx} = 1 \cdot y = y = 10$$

$$\frac{df}{dy} = \frac{df}{dt} \cdot \frac{dt}{dy} = 1 \cdot x = x = 3$$

$$\frac{df}{dt} = \frac{df}{dt} = 1$$