

Automatic Differentiation Reverse Mode

Zachary Boor

Uses

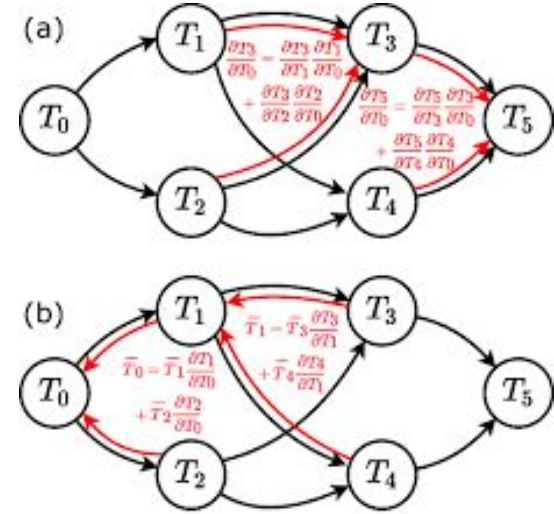
- Deep machine learning
- Neural Networks



Mathematical Side of Reverse AD

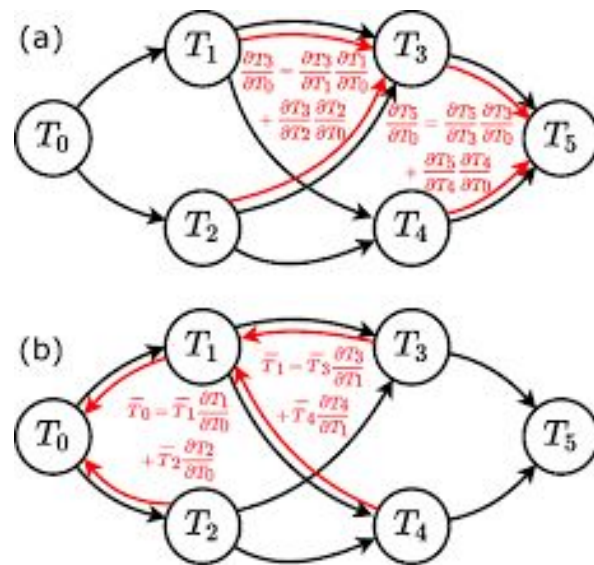
Forward Pass

- Define all Input Variables
- Define all Intermediate Variables
- Define the Output Variables



Backward Pass

- Adjointso
 - Fancy term for gradients
- Output to inputs



Coding Side of Reverse AD

Forward Pass

- Define Function
- Code each operation with operator overloading
 - Operator overloading overrides operations to include AD
- Store data necessary for derivative calculation



Backward Pass

- Variable to hold result of operators
- Closure Function
- Work with the Adjoint
- Partial Derivatives



$$a * b + c * a / e * d + a * e - d / b$$

a, b, c, d, e = Variable(25.0, 1.0), Variable(4.0, 0.0),
Variable(-5.0, 0.0), Variable(10.0, 0.0), Variable (3.0, 0.0)



a = primal: 25.0, adjoint: 1.0, b = primal: 4.0, adjoint: 0.0, c = primal: -5.0, adjoint: 0.0, d = primal: 10.0, adjoint: 0.0, e = primal: 3.0, adjoint: 0.0

d = primal: 10.0, adjoint: -41.916666666666664

a.adjoint = -8.666666666666664, b.adjoint = 25.625, c.adjoint = 83.33333333333333, d.adjoint = -41.916666666666664, e.adjoint = 163.88888888888889



Sources

<https://huggingface.co/blog/andmholm/what-is-automatic-differentiation>

<https://mostafa-samir.github.io/auto-diff-pt2/>

