Fundamentals

CSE 493G1, Section 1

Slides by Tanush Yadav

Agenda

- Introductions
- Course Logistics
- Soft Prerequisites
- Google Colab
- NumPy Foundations
- Mathematical Foundations

Introductions

About Me

- Tanush (he/him)
    - Sophomore in CS
    - 2nd time TAing, did 390Z last quarter
    - Hobbies: running, hiking, listening to music

    - Email: tanush@cs.washington.edu
    - OH: Thurs 11:30 – 1:30, Location TBD

Icebreakers

- Name
- Major + Year
- What’s something memorable you did over winter break?
- Why’d you sign up for this class?
- What courses are you taking this quarter?

Course Logistics

Assignments

[0] Colab Setup

[1] KNN, SVM, Softmax (1/18)

[2] Multi-layer NNs, Image Features, Optimizers (1/30)

[3] Normalization Layers, Dropout, CNNs (2/13)

[4] Pytorch, Visualizations, RNNs (2/22)

[5] Transformers, GANs, Self-Supervised (3/5)

Assignments with friendly warnings

[0] Colab Setup resolve weird environment bugs here

[1] KNN, SVM, Softmax (1/18) probably your first time with this codebase, NumPy, and the DL math

[2] Multi-layer NNs, Image Features, Optimizers (1/30)

[3] Normalization Layers, Dropout, CNNs (2/13) extremely tedious derivatives

[4] Pytorch, Visualizations, RNNs (2/22)

[5] Transformers, GANs, Self-Supervised (3/5) these models take a while to train

Quizzes

- 5 quizzes, roughly aligning to the 5 assignments
- Lowest quiz score dropped
- Next week's section is dedicated to quiz prep!

Projects

- Proposal due 2/6, Milestone due 2/29, Report due around finals week
- Figure out your groups and get started! These take a LONG time

- Will discuss in section on February 2nd
- Come to office hours to get feedback on ideas in the meantime
- Read the projects page on website for more details

- (tentative) opt-in W credit, details will be posted on Ed and course web later

Soft Prerequisites

MATH 126 (Multivariable Calculus)

CSE 312 (Probability / Statistics)

CSE 332 (Data Structures)

We assume you know basic Python.

- CSE 312 tutorial (check your 312 Ed)
- Stanford CS231n tutorial

You'll have to pick up on NumPy.

- Simple library for doing scientific computing in Python
- Optimizes matrix operations under-the-hood

We assume you know multivariable calculus.

- Math 126
- Derivatives (and partial derivatives)
- Chain rule

You'll have to pick up on some DL-specific stuff.

- Derivatives of matrices (Jacobians)
- Odd notation not used in mathematics

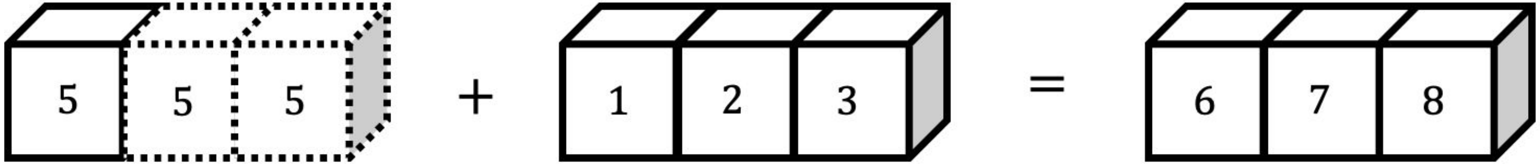
NumPy Foundations

Live Coding!

- Colab demo
- NumPy demo

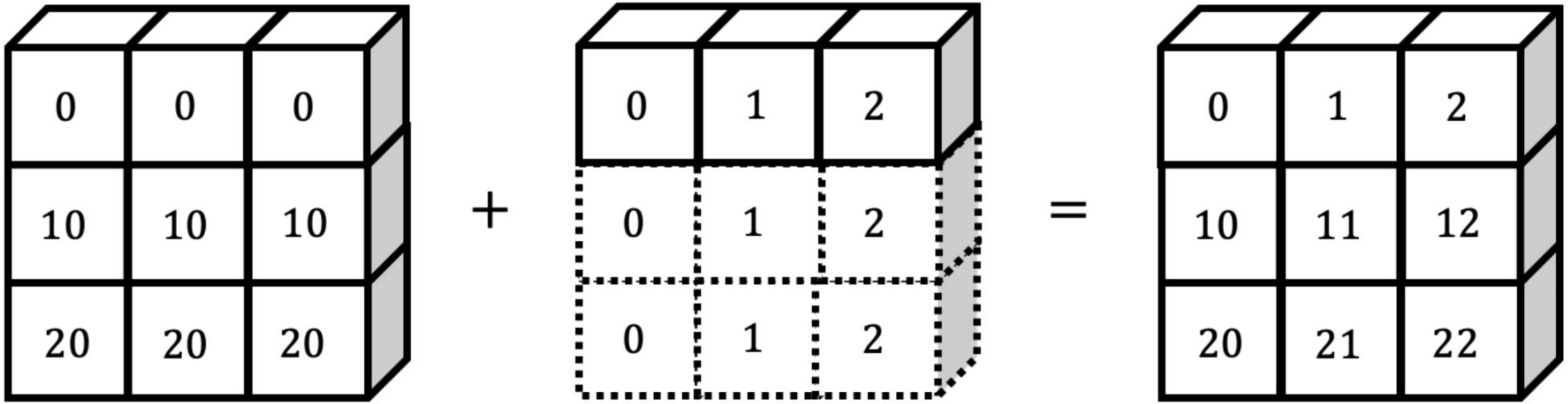
Broadcasting

$$5 + \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 6 \\ 7 \\ 8 \end{bmatrix}$$



Broadcasting

$$\begin{bmatrix} 0 & 0 & 0 \\ 10 & 10 & 10 \\ 20 & 20 & 20 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 2 \end{bmatrix}$$



Broadcasting

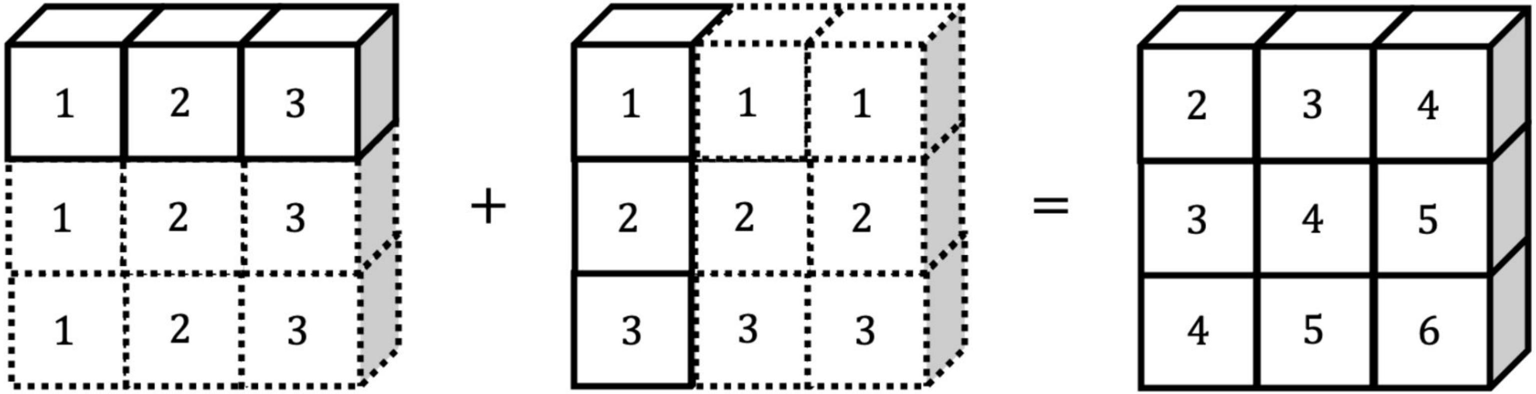
“In the context of deep learning, we also use some less conventional notation. We allow the addition of a matrix and a vector, yielding another matrix: **C = A + b**, where $\mathbf{C}_{x,y} = \mathbf{A}_{x,y} + \mathbf{b}_{y}$.

In other words, **the vector b is added to each row of the matrix**. This shorthand eliminates the need to define a matrix with b copied into each row before doing the addition. This implicit copying of b to many locations is called broadcasting.”

- *Deep Learning*, Goodfellow, pg 32

Broadcasting

$$\begin{bmatrix} 1 & 2 & 3 \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$



Broadcasting

1. If the two arrays differ in their number of dimensions, the shape of the one with fewer dimensions is padded with ones on its leading (left) side.
2. If the shape of the two arrays does not match in any dimension, the array with shape equal to 1 in that dimension is stretched to match the other shape.
3. If in any dimension the sizes disagree and neither is equal to 1, an error is raised.

From Jake VanderPlas' *Python Data Science Handbook* (link)

Mathematical Foundations

Chain Rule

Chain Rule for One Independent Variable

Suppose that $x = g(t)$ and $y = h(t)$ are differentiable functions of t and $z = f(x, y)$ is a differentiable function of x and y. Then $z = f(x(t), y(t))$ is a differentiable function of t and

$$\frac{dz}{dt} = \frac{\partial z}{\partial x} \cdot \frac{dx}{dt} + \frac{\partial z}{\partial y} \cdot \frac{dy}{dt}, \tag{14.5.1}$$

where the ordinary derivatives are evaluated at t and the partial derivatives are evaluated at (x, y).

Section 14.5, *Calculus*, Strang and Herman.

Chain Rule

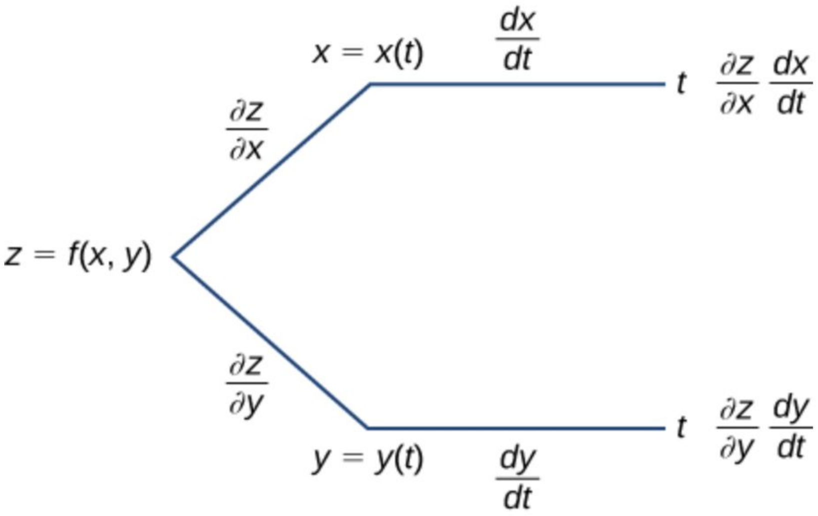


Figure 14.5.1: Tree diagram for the case $\frac{dz}{dt} = \frac{\partial z}{\partial x} \cdot \frac{dx}{dt} + \frac{\partial z}{\partial y} \cdot \frac{dy}{dt}$.

Section 14.5, *Calculus*, Strang and Herman.

Chain Rule

Chain Rule for Two Independent Variables

Suppose $x = g(u, v)$ and $y = h(u, v)$ are differentiable functions of u and v, and $z = f(x, y)$ is a differentiable function of x and y. Then, $z = f(g(u, v), h(u, v))$ is a differentiable function of u and v, and

$$\frac{\partial z}{\partial u} = \frac{\partial z}{\partial x}\frac{\partial x}{\partial u} + \frac{\partial z}{\partial y}\frac{\partial y}{\partial u} \tag{14.5.2}$$

and

$$\frac{\partial z}{\partial v} = \frac{\partial z}{\partial x}\frac{\partial x}{\partial v} + \frac{\partial z}{\partial y}\frac{\partial y}{\partial v}. \tag{14.5.3}$$

Section 14.5, *Calculus*, Strang and Herman.

Chain Rule

Generalized Chain Rule

Let $w = f(x_1, x_2, \dots, x_m)$ be a differentiable function of m independent variables, and for each $i \in 1, \dots, m$, let $x_i = x_i(t_1, t_2, \dots, t_n)$ be a differentiable function of n independent variables. Then

$$\frac{\partial w}{\partial t_j} = \frac{\partial w}{\partial x_1}\frac{\partial x_1}{\partial t_j} + \frac{\partial w}{\partial x_2}\frac{\partial x_2}{\partial t_j} + \cdots + \frac{\partial w}{\partial x_m}\frac{\partial x_m}{\partial t_j}$$

for any $j \in 1, 2, \dots, n$.

Section 14.5, *Calculus*, Strang and Herman.

Questions?

Thanks for coming to section!