Numerical Linear Algebra Preliminaries I: Matrix-Vector Multiplication

Feifan Fan

Imperial College London

feifan.fan19@imperial.ac.uk

Autumn 2021

Why Matrix-Vector Multiplication?

- In this course we are going to talk about the **computational** part of Linear Algebra, which is basically matrix multiplications.
- However, the **Matrix-Matrix Multiplications** seemed to be more complicated, in both **amount of computation** and **time complexity for computation,** especially when both matrices are large.
- So we could start with a special case to get a more clear insight of how to make matrix multiplication computationally: **multiply a matrix with a vector** instead, which clearly reduces the complexity of the problem.

Problem Description

Consider the following problem:

$$\text{Find the value of } b = Ax, \text{ where } A \in \mathbb{C}^{m \times n}, x \in \mathbb{C}^n.$$

Note that here we are talking about vectors and matrices over the **complex vector space** $\mathbb{C}^n$, and so are we for the rest of the course.

- Here what we are going to do is,
  - giving out a **generalised formula** for (elements of) resulted b
  - using the formula to **devise an algorithm** to compute the result.
- And we have two approches to solve this problem.

Given that $b = Ax$ where A and x are in correct dimensions. If we write the structure of them explicitly we would have:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}, x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

And if we expand $b = Ax$ via the **basic matrix multiplication rule**,

the result would be:

$$b = Ax =$$

Approach I: Basic Interpretation (cont.)

To generalise the result we get, we take the i th component of the resulted vector b, and it would be:

$$b_i =$$

And this is the formula we wanted with **basic interpretation** of the matrix-vector multiplication. Then we would use this idea to devise an algorithm for computing $b = Ax$.

Algorithm: with Basic Interpretation

Before we introduce the algorithm, let's think how to devise an algorithm properly in a top-down level:

To devise an algorithm, we could generally follow the follwing steps:

- What do we **have**, and what would be finally **want to get**?
- How to get what we want via **simple ideas**?
- Can you **break up** the original problem into **smaller problems**?
- How to generally solve these with **maths behind it**?
- How to achieve your schema **in details** with **programming skills**?

We would apply this method to find an algorithm for computing $b = Ax$ with basic interpretation.

Algorithm: with Basic Interpretation (cont.)

- What do we **have**, and what would be finally **want to get**?

  We have a matrix $A \in \mathbb{C}^{m \times n}$ and a vector $x \in \mathbb{C}^n$, and we want the algorithm returns a vector $b = Ax$.

- How to get what we want via **simple ideas**?
- Can you **break up** the original problem into **smaller problems**?

  We could get value of b by calculating the values of elements one-by-one. We would start with calculating b_1, and then repeat the similar procedure to calculate $b_2, b_3 \dots b_m$. Finally the collection of $\{b_i\}$ would be the elements of $b = Ax$.

Algorithm: with Basic Interpretation (cont.)

- How to generally solve these with **maths behind it**?

  Since we have the generalised formla for computing b_i with basic interpretation, we could just apply this formula in each sub-task for computing i th component of b.

- How to achieve your schema **in details** with **programming skills**?

  You could pause at here and think about it. Check your schema with the algorithm in the next page!

  (Hint: Think about how we would do in programming with repeated procedures, and how to apply the summation in the generalised formula with programming strategies.)

Algorithm: with Basic Interpretation (cont.)

Algorithm 1 Matrix-Vector Multiplication with Basic Interpretation

Require: $A \in \mathbb{C}^{m \times n}$, $x \in \mathbb{C}^n$
Ensure: $b = Ax \in \mathbb{C}^m$

$b \leftarrow$ an empty array with length m
for $i = 1 \dots m$ **do**
  $sum \leftarrow 0$
  for $j = 1 \dots n$ **do**
    $sum \leftarrow sum + A[i][j] \times x[j]$
  end for
  $b[i] = sum$
end for
return b

Algorithm: with Basic Interpretation (cont.)

Note that

- The algorithm above only provides you an provision that how the matrix multiplication works with basic interpretation.
- I left it as not the "best" algorithm and gave you space for implementing your own version of function. You are also free to optimise the algorithm above, given that it preserves the functionalities!

The End