USING MATLAB IN GEOSCIENCES

My Experience in teaching Undergraduate Atmospheric Science

Anantha Aiyyer

October 19, 2015

North Carolina State University

INTRO

Introduction

- Faculty Member in Dept. of Marine, Earth and Atmospheric Sciences at North Carolina State University.
- Research Areas: Tropical weather and large scale atmospheric dynamics (e.g., atmospheric waves, hurricanes).

Introduction

- I am NOT a MATLAB Expert!
- However, I have been programming in one language or other since high school.
- I can, given a little time, write code in MATLAB because I know programming principles and I code nearly daily for my research.
- MATLAB can be a very effective first language to learn programming principles. I learned BASIC in high school!

QUANTITATIVE SKILLS IN ATMOSPHERIC SC.

Quantitative Skills in Atmospheric Science

- Typically, students entering our undergraduate program have had a fascination with weather from childhood Tornadoes, Hurricanes, snowstorms
- Unlike physics/math/computer science, however, our students do not come with an expectation of heavy math and computation

Quantitative Skills in Atmospheric Science

- Students in our program are required to take two Calculus and physics courses, followed by a course in differential equations and a programming course
- Students begin taking core atmospheric science classes starting second year
- Core classes include Thermodynamics, Fluid Dynamics (I and II), Atmospheric Physics, Air Quality and Climate, and Synoptic-Dynamic Meteorology.

ISSUES WITH STUDENT PREPARATION

Issues with Student Preparation

- Expectation: Students getting to core and elective Atmospheric science course are well prepared in calculus and programming principles
- Reality: Students are ill prepared in all three pre-requisite areas: Calculus, Physics and Programming
- Not unique to my institution. Colleagues from other places (esp., large public schools) share the same assessment.

DOWNSIDE OF OUTSOURCING QUANTITATIVE COURSES

Outsourcing Issues

- Typically class sizes in introductory programming, math and physics are large (50 and up; some 200 and up)
- Atmospheric Science students are a small minority; spread across sections with engineering and science majors
- Classroom activities in large shared courses lack context based examples; or are skewed towards engineering or physics
- Our Students focus on surviving instead of **learning**

WHY MATLAB

Why MATLAB?

- A couple of years ago, our atmospheric science faculty came together to change our programming language requirement
- Prior to this, students could take any programming language. This was not effective.
- MATLAB was the popular choice for us as some faculty use it for their research, and others like me were willing to learn

MATLAB Strategy

- Get all atmospheric science students to take MATLAB; preferably all together in one section.
- A section taught by the math department seemed like a good choice.
- Faculty in our department were encouraged to incorporate MATLAB in their class activities. Some agreed.

MATLAB Strategy

- Prior to our MATLAB adoption, I was involved in developing a new course to help improve the math preparation of our students
- My effort was supported by an NSF grant that also funded my research activities (2009–2014)
- Copying from physics, I started teaching Mathematical Methods in Atmospheric Science
- I added a lab section (taught by a TA) that incorporates MATLAB in all activities.

Faculty participation

- In our program, we now have five courses that use MATLAB:
- Introduction to Atmospheric Science (2nd year), Math methods in Atmospheric Science (2nd year); Atmospheric Thermodynamics (2nd year); Atmospheric Dynamics (3rd year) and Fluid Physics (4th year).

- However, a few core courses still do not use MATLAB (or any programming language). This is a challenge because not all faculty members use MATLAB or can code at all.

- We continue to have further discussion in our department to encourage a more wide spread adoption of MATLAB.

EXAMPLES

Example 1

The Gradient Operator

$$\nabla = \hat{i}\frac{\partial}{\partial x} + \hat{j}\frac{\partial}{\partial y} + \frac{\partial}{\partial z}$$

An important operator that finds applications to key fluid dynamic concepts. One of them, the Pressure Gradient Force which is:

$$-\frac{1}{\rho}\nabla P(x, y, z, t)$$

Example 1

MATLAB exercise with the Gradient Operator

Lab Goals

- To plot a scalar field using contours
- To overlay vectors on the contours
- To relate pressure gradient to pressure gradient force

Given the hypothetical pressure field:

$$P(x, y) = A[Cos(kx) + Sin(ly)]$$

Where $k = 2\pi/L_x$ and $l = 2\pi/L_y$ are respectively, zonal and meridional wavenumbers. A is the amplitude.

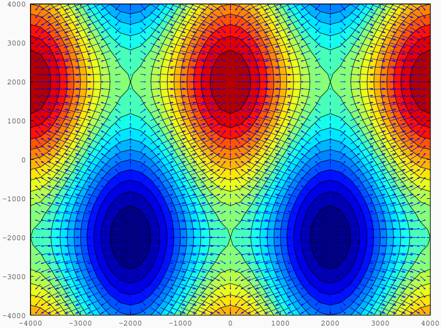
Example 1

Lab Activities

- Write a MATLAB program to visualize P using contours.
- Change the zonal and meridional wavelengths and compare changes in pressure patterns.
- Evaluate ∇P to get the 2-D gradient vector field
- Add lines to MATLAB code to overlay gradient vectors.
- In what direction does the pressure gradient point? What about pressure gradient force? Explain why the pressure gradient force direction differs from the direction of the gradient

Example 1

Typical figure produced by the student showing gradient vectors pointing at the local high values. (i.e. vectors point up gradient or up slope)



Example 2

MATLAB exercise with Finite Differences

Lab Goal

- To calculate first and second order differentials using centered difference method.

$$\frac{\partial \psi}{\partial x}$$
$$\frac{\partial^2 \psi}{\partial x^2}$$

Example 2

Lab Activity: Background

Consider a one-dimensional field $\psi(x)$. If we have measurements of ψ at equal intervals of x, spaced Δx apart, then we can approximate the first and second derivatives of ψ by the centered difference formula given as below:

$$\frac{\partial \psi}{\partial x} = \frac{\psi(x+\Delta x) - \psi(x-\Delta x)}{2\Delta x}$$

$$\frac{\partial^2 \psi}{\partial x^2} = \frac{\psi(x+\Delta x) + \psi(x-\Delta x) - 2\psi(x)}{\Delta x^2}$$

Example 2

Lab Activity 1

Using the formula for geostrophic velocity:

$$\vec{V}_g = \frac{1}{\rho f}\hat{k} \times \hat{\nabla} P$$

Show that the geostrophic vorticity can then be written as:

$$\zeta_g = \frac{1}{\rho f}(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2})$$

Example 2

Lab Activity 2

Consider the pressure field,

$$P(x, y) = ACos(kx)Sin(ly) \tag{1}$$

Write MATLAB code to draw contours of P with color shading.

Example 2

Lab Activity 3 For the given pressure field P(x,y), derive the expression for vorticty, ζ_g using:

$$\zeta_g = \frac{1}{\rho f}(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2})$$

Your result will be the exact, Analytical expression for ζ_g.

$$\zeta_g(x, y) = -(k^2 + l^2)ACos(kx)Sin(ly) \tag{2}$$

Now plot contours of ζ_g using the analytical formula you derived. Examine the order of magnitude of the values of vorticity.

Example 2

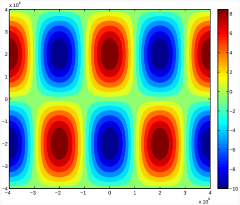
Lab Activity 4 Now instead of using the exact analytical form, use the approximate center difference form shown below. Calculate the vorticity in MATLAB and plot it.

$$
\begin{aligned}
\zeta_g(j,i) &= \frac{1}{f\rho}\Big(\frac{P(j,i+1)+P(j,i-1)+2P(j,i)}{\Delta x^2} \\
&+ \frac{P(j+1,i)+P(j-1,i)-2P(j,i)}{\Delta y^2}\Big)
\end{aligned}
$$

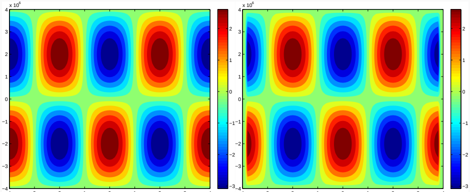
Compare your plots for Pressure and the two plots for vorticity and answer additional questions.

Example 2

(a) Pressure Field



(b) Vorticity fields (analytical and approximate finite difference)



STUDENT PERSPECTIVES

Student Perspectives

Question: Do you feel that knowing how to code is important for your career in meteorology?

- Yes, ...every career path in the field relies so much on modeling
- ”...makes you more attractive to employers in the industry
- Every model we look at in class is generated by code in some way, and that is just a small example of its uses for meteorologists
- knowing MATLAB was a nice plus on my resume when I applied for the Early Alert internship

Student Perspectives

Question: What aspects of MATLAB do you find the most difficult?

- MATLAB is probably one of the easier coding software I've encountered
- I find most of the aspects in class to be difficult..
- ...hardest part of using Matlab is taking a problem in the language of math and writing it as code..
- My biggest difficulty in MATLAB has been understanding loops.

Student Perspectives

Other Questions (selected)

- Is matlab the first programming language you were exposed to?
  Java, Python, C

- Do you feel you learned much in your the MATLAB course you took in the mathematics department?
  Overwhelming response was NO

- Would you consider using MATLAB to do assignments in other classes [that did not require you to use it]?
  Overwhelming response was YES

LESSONS LEARNED

Lessons Learned

- Students in Geoscience may not enter the program with the expectation of doing as much quantitative analysis as in mathematics or physics.

- Some students will have a an aversion/phobia to learning MATLAB or any programming language.

Lessons Learned

- Instructors need to be more empathetic. Most of us are self-taught, but we can't always expect students to be that motivated; at least initially.
- Other faculty members in a program need to buy in to the idea of using MATLAB in their courses.
  In our program we may offer our own MATLAB intro course
- We need, especially in Atmospheric Sciences, a good set of MATLAB based activities for each course in the curriculum.
- Since most Atmospheric Science programs follow the American Meteorological Society guidelines, this should help in standardizing programming activities.

Some Recommendations

- Stronger community involvement in developing MATLAB resources, especially in Atmospheric Science.
- Better contours over maps and functions (meteorology specific) in MATLAB. Compare with IDV and NCL

FINAL THOUGHTS

What I plan to do Further

- Contribute to the online MATLAB activities using my lab excercises
- Encourage students in my department to start a Coding Club and volunteer to be the faculty adviser to start with
- Expand programming activities to include version control and online archiving for easy sharing (e.g., github, bitbucket)

Questions?