Vital Role of Training and Education in Big Data Applications

David A. Yuen, University of Minnesota@ Minneapolis and Ultimate Vision Technology, Beijing, China

Gabriele Morra, Louisiana University @ Layfette, and Ultimate Vision Technology, Beijing , China

Outline

- 1. Dave Yuen Big Data Impact on Society, Need to train students
- outside computer science, need for applications
- 2. Gabriele Morra Things that potential student should learn based on Python and examples in geosciences

- 3. Dave Yuen Future development , especially in China
- Curriculum Reformation

Big Data has arrived in the 2010’s

- Big Data
- Confluence of factors : Internet, Mobile Phone and GPU all came together around 2007
- Artificial Intelligence , Machine Learning, Deep Learning
- AI revolution,
- Two Important practical Applications : self-driven automobile
- Smart cities
- Scientific applications: detection of gravitational waves (Noble Prize 2017), high-energy physics, prediction of physical properties

All of these are Data-Driven activities

- Too much emphasis and directions have been coming from computer
- Scientists, especially in education

- We need domain scientists who understand the application to take
- Directions, particularly in mass-media and culture Big Data and A.I.
- activities in the arts (translation of ancient languages in museums
- for governments, online education of mathematical concepts, such as complex variables, logarithms)

Need for teaching Big Data and Data Science to the Mass

- Campus wide efforts in USA
- University of Washington, Columbia University, New York University
- UC Berkeley, Rice University……..
- Similar to what was happening in supercomputing efforts 33 years ago in 1984 , (University of Illinois, Princeton University, Cornell University, UC San Diego)
- Another model would be training centers in China, Mc Data and now
- **Mac-Teach anybody can eat a hamburger (piece of cake)**
- like learning English for Asian students, because of bad English teachers at universities and high schools, they need tutors and trainers from outside

Now I give the floor to Gabriele F. Morra from University of Louisiana

Students and Professionals need to Rapidly Learn Programming, Numerical Modeling and Big Data Analysis. But how?

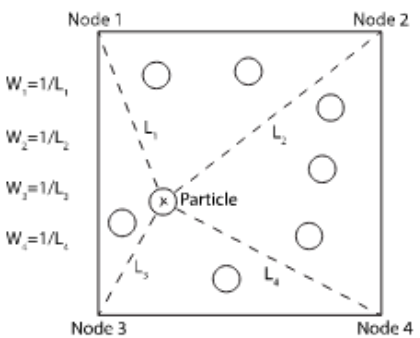
- In the Past people used Matlab, a proprietary software. Python today offers the same, but in an open and fast growing environment. It reminds the time when Linux took over Unix.
- It is efficient for ODE solutions with the Numpy module.
- It offers the same visualization tools of Matlab with Matplotlib
- It goas at c-speed with just in time compilation (Cython, etc)
- Can be parallelized easily with mpi4py, PyCuda (for GPUs) and petsc4py)
- It is designed for processing large amount of data (e.g. Pandas)

Prasanna Gunawardana

Prasanna started his master at UL in Physics coming from an engineering degree in Sri Lanka. In one year he learned to Program, the Particles in Cell Method, and geodynamic modeling of lithosphere and mantle.

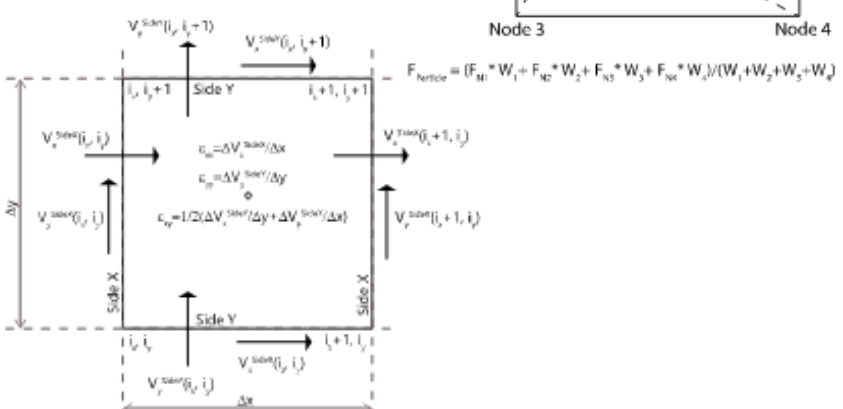
```
def projectLatticeToParticles(w1,w2,w3,w4,trIX,trIY,f,ft):
        ft[:]=(w1[:]*f[trIX[:],
                trIY[:]]+w2[:]*f[trIX[:]+1,
                trIY[:]]+w3[:]*f[trIX[:],
                trIY[:]+1]+w4[:]*f[trIX[:]+1,
                trIY[:]+1])/(w1[:]+w2[:]+w3[:]+w4[:])
        return (ft)
```

One line of code. Vectorized. Extremely fast.



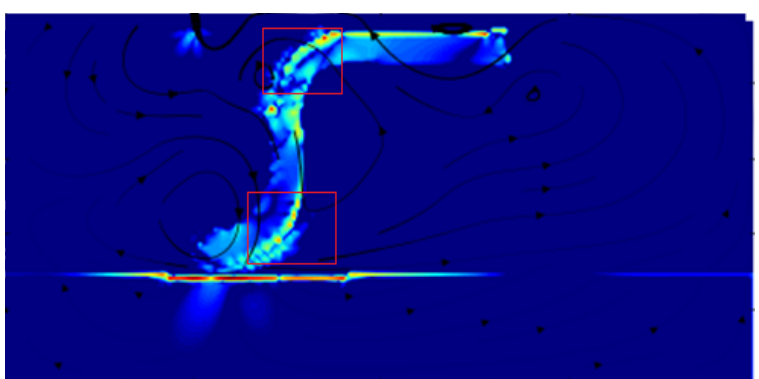

```
def projectParticlesToLattice(w1,w2,w3,w4,trIX,trIY,f,ft,tw):
        f[:,:]=0.0
        tw[:,:]=0.0
        f[trIX[:],trIY[:]]+=ft[:]*w1[:]
        f[trIX[:]+1,trIY[:]]+=ft[:]*w2[:]
        f[trIX[:],trIY[:]+1]+=ft[:]*w3[:]
        f[trIX[:]+1,trIY[:]+1]+=ft[:]*w4[:]
        tw[trIX[:],trIY[:]]+=w1[:]
        tw[trIX[:]+1,trIY[:]]+=w2[:]
        tw[trIX[:],trIY[:]+1]+=w3[:]
        tw[trIX[:]+1,trIY[:]+1]+=w4[:]
        f[:,:]/=tw[:,:]
        return (f)
```

Compact. Easy to Understand and modify.

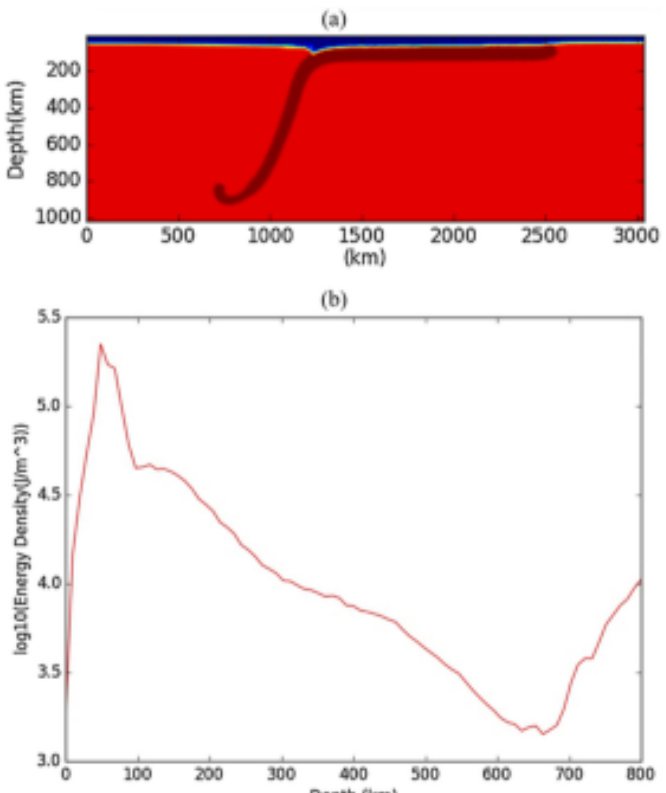


Student 1: Prasanna Gunawardana

Prasanna started his master at UL in Physics coming from an engineering degree in Sri Lanka. In one year he learned to Program, the Particles in Cell Method, and geodynamic modeling of lithosphere and mantle.



Gunawardana and Morra,
Journal of Geodynamics 106 (2017) 33–45



How is Numerical Python so fast?

1. Vectorization of most operations. Highly optimized if NO LOOPS.

```
In [27]: %timeit c=addArray(a,b) #standard python
1 loops, best of 3: 639 ms per loop
In [28]: %timeit c=a+b #NumPy arrays broadcasting
100 loops, best of 3: 3.74 ms per loop
```

2. Cython (=C in Python) implementation of difficult routines.

```
cimport numpy as np
def setNegativeValuesToZero(int n, int m, np.ndarray[double, ndim=2] a):
    cdef int i, j
    for i in range(n):
        for j in range(m):
            if a[i,j]<0:
                a[i,j]=0.
```

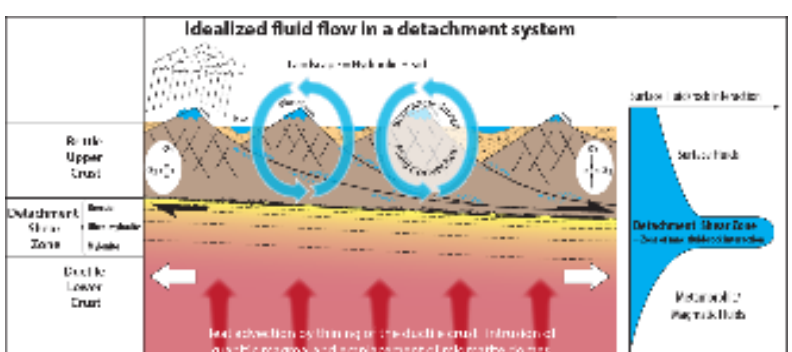
3. Going parallel with the extension libraries (mpi4py, pyCuda, petsc4py).

```
from mpi4py import MPI
if rank == 0:
    data = np.arange(10000, dtype=np.float64)
    comm.Send(data, dest=1, tag=13)
elif rank == 1:
    data = np.empty(10000, dtype=np.float64)
    comm.Recv(data, source=0, tag=13)
```

Student 2: Justin (not the real name)

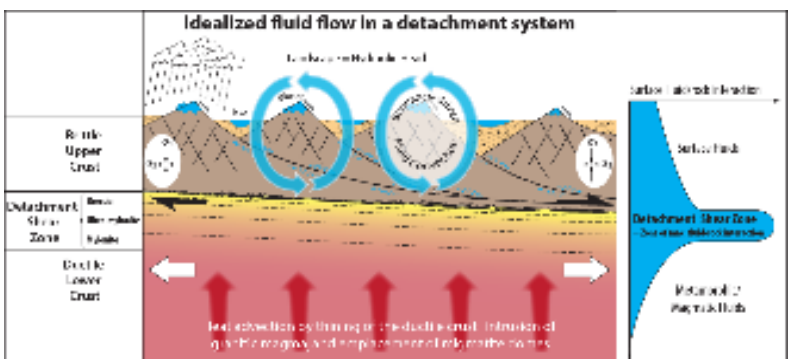
A student in difficulty, close to leave without completing his studies. By using Python he could learn to write a finite volume solver for Darcy Flow in 3 months, just using the numerical libraries, NumPy. He completed successfully his thesis by writing a code of only 50 lines.

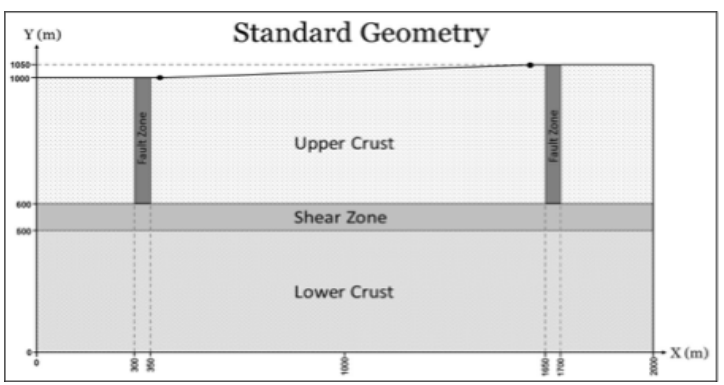
```
diffusivity1D=diffusivity.reshape(nxc*nyc)
diffSpMatrix=sparse.spdiags([diffusivity1D],[0],nxc*nyc,nxc*nyc).tocsc()

# create the Laplacian Operator
LaplacianOp=PIC.sparseVariableLaplacianOperator(nxp,nyp,dx,dy,diffSpMatrix)
LaplacianOp=PIC.addBC(LaplacianOp,nxp,nyp)*dx*dx
L=sparse.eye(nxp*nyp).tocsc()-LaplacianOp*deltaTime

# create initial pressure
pHydro1 = np.outer(np.ones(X.size),((yMax-Y)*density*gravity))
pressure2[F1toF2]=(X-0.325)*density*gravity*0.05
pressure2[postF2]=(1.675-0.325)*density*gravity*0.05
pFluid= np.outer(pressure2,np.ones(Y.size)) #+ pHydro1
pFluid=pFluid.reshape(nxp*nyp)
for thisStep in np.arange(1000):
   pFluid = la.spsolve(L,pFluid)
pFluid=pFluid.reshape(nxp,nyp)

G=PIC.sparseGradientOperator(nxp,nyp,dx,dy)
Zout = PIC.sparseZoomInOperator(nxp,nyp,dx,dy).transpose()

vx=-diffSpMatrix*G[0].dot(pFluid.reshape(nxp*nyp))
vy=-diffSpMatrix*G[1].dot(pFluid.reshape(nxp*nyp))
vx=Zout.dot(vx); vx=vx.reshape(nxp,nyp)
vy=Zout.dot(vy); vy=vy.reshape(nxp,nyp)
```

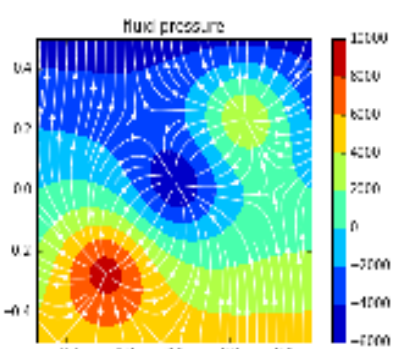


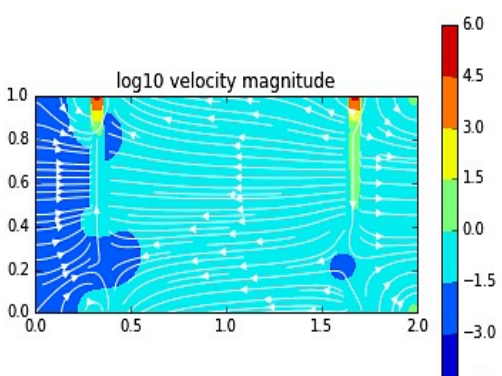
Student 2: Justin (not the real name)

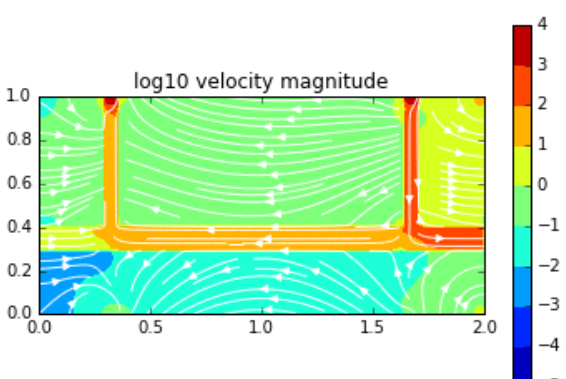
A student in difficulty, close to leave without completing his studies. By using Python he could learn to write a finite volume solver for Darcy Flow in 3 months, just using the numerical libraries, NumPy. He completed successfully his thesis by writing a code of only 50 lines.











Student case 3:
Kyle Killion, Rajeev Kumar, Celia Taylor

Three students of the Southern Methodist University, completing a Master of Science in Data Science, decided to do their capstone work on geodynamics. They contacted me and we started to work on volcano seismicity.

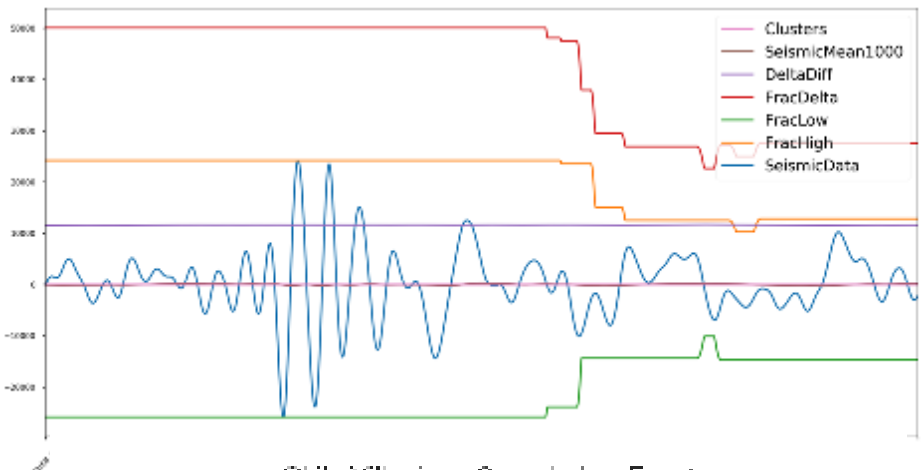
Python was central in all we did.

- Machine Learning was from SciKits Learn of Python.
- Seismic data were downloaded and processed using ObsPy.
- Data preparation and processing was all done with Python.

The Students in only three months were able to reproduce the results of professional volcanologists analyzing the strombolian activity of Villarica in Chile.

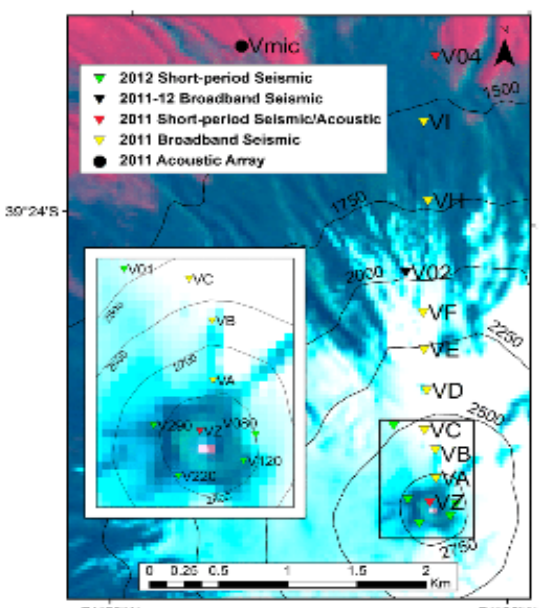
Student case 3:
Kyle Killion, Rajeev Kumar, Celia Taylor

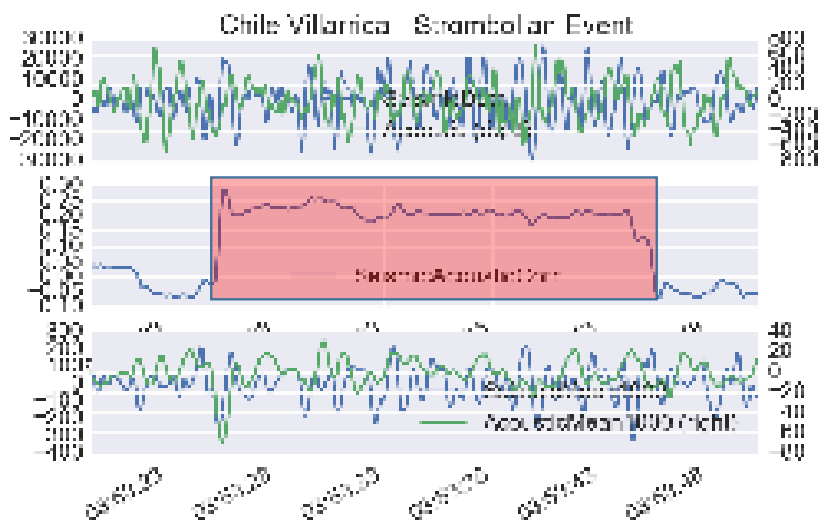
The data was gathered from publically availab
IRIS website with Obs Py http://ds.iris.edu/



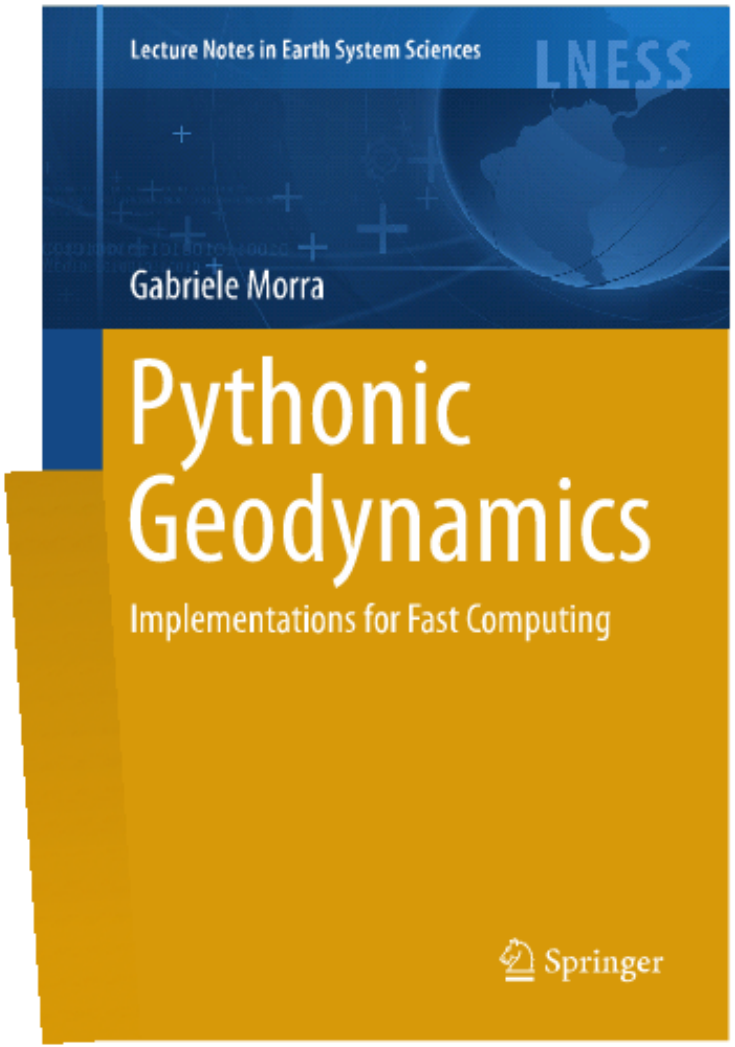
Chile's Villarrica Volcano

Richardson, Joshua P., Waite, Gregory P., Palma, Jose Luis,
"Varying seismic-acoustic properties of the fluctuating lava lake at Villarrica volcano, Chile,"

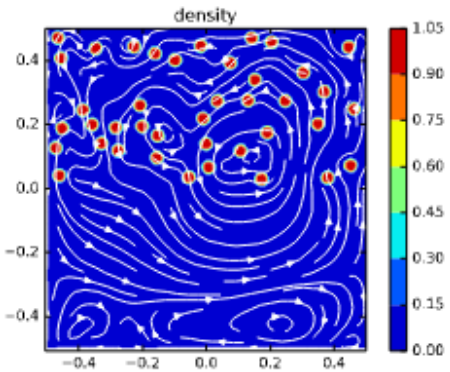




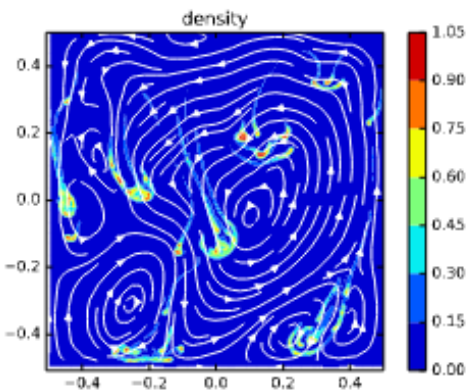
These methods and many more in this book



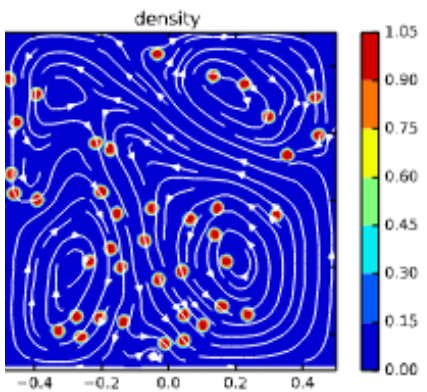
An introductory level book on Geodynamics with Python, for graduate and undergraduate students.



Soft particles



Stiff particles



Let me share with you the future of our vision

- At the recent 19 th Congress in Beijing
- President Xi Jinping propounded the need for educational reformation in China,
- Curriculum changes in many fields in particular Big Data
- Goal is year 2025
- We just established a new company called Mac-Teach at Beijing

Led by Xianying Wang as leader for Mac-Teach

Mac-Teach is a company based in Beijing

- .Staffed by both Chinese instructors and foreign professors
- We will help both private institutions and universities
- With training in Big Data , AI, DL visualization and technicalconsulting
- We will first help **China University of Geosciences in** Wuhan to open Up a brand-new bachelor's program starting in 2018

- First class in 2018 , aiming for 100 to 150 freshmen (18 years old)

- With 7 to 8 new courses to be designed by us.

The FUTURE

- We must bring applications to Big Data Education
- Domain scientists must be brought into play !
- This same **modus operandi** in China can be introduced elsewhere