



中国石油大学
CHINA UNIVERSITY OF PETROLEUM

JET 101

石油工程数值分析及数据可视化方法

王斌

石油工程学院 水射流实验室

邮箱: binwang.0213@gmail.com

CUPB

厚积薄发，开物成物

考核与要求

■ 总成绩 = 课堂小测验10% + 作业40% + 期中大作业 20% + 期末大作业 30%

- 课堂小测验（5次） – 截至日期 当天课程结束
- 课后作业（5次） – 截至日期 下一周上课之前
- 期中大作业（固定编程题） – 截至日期 2021年1月7日
 - 提交报告和源代码
 - 固定选题
- 期末大作业（开放编程题） – 截至日期 2022年1月28日
 - 提交报告和源代码
 - 自由选题



Python科学计算

□ 学习目标

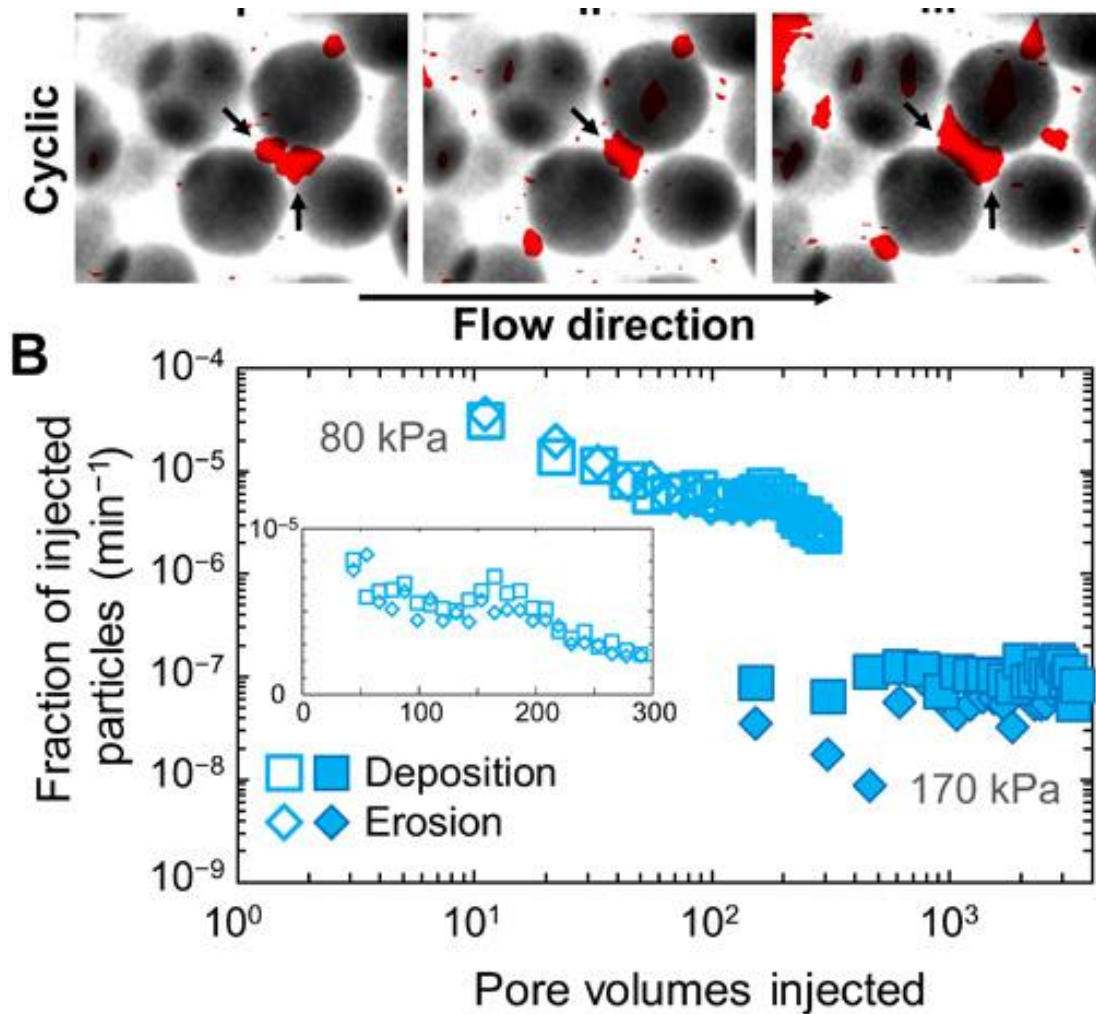
学习目标	学习成果	效果考察	课程活动
应该掌握哪些知识？	应该能够做哪些事情？	怎么考察学生？	应该怎么学习？
1. Numpy和Matplotlib入门 2. Python实战 <ul style="list-style-type: none">• 向量点乘• 矩阵转置• 矩阵相乘 3. 搜索并使用第三方库(Excel, PyVista)	1. 能够使用Numpy创建向量和矩阵 2. 能够使用Matplotlib画图 3. 能够手动对简单线性代数问题进行编程 4. 能够安装第三方库	1. 测试学生是否能运行代码 2. 测试Notebook使用情况 3. 课堂测验简单问题 <ul style="list-style-type: none">• 安装pyvista并运行• 随机生成二维点云并画图	1. 观看Python网络课程 2. 完成课后作业7, 8, 9 3. 启动中期Project



Python科学计算

□ Numpy和Matplotlib入门

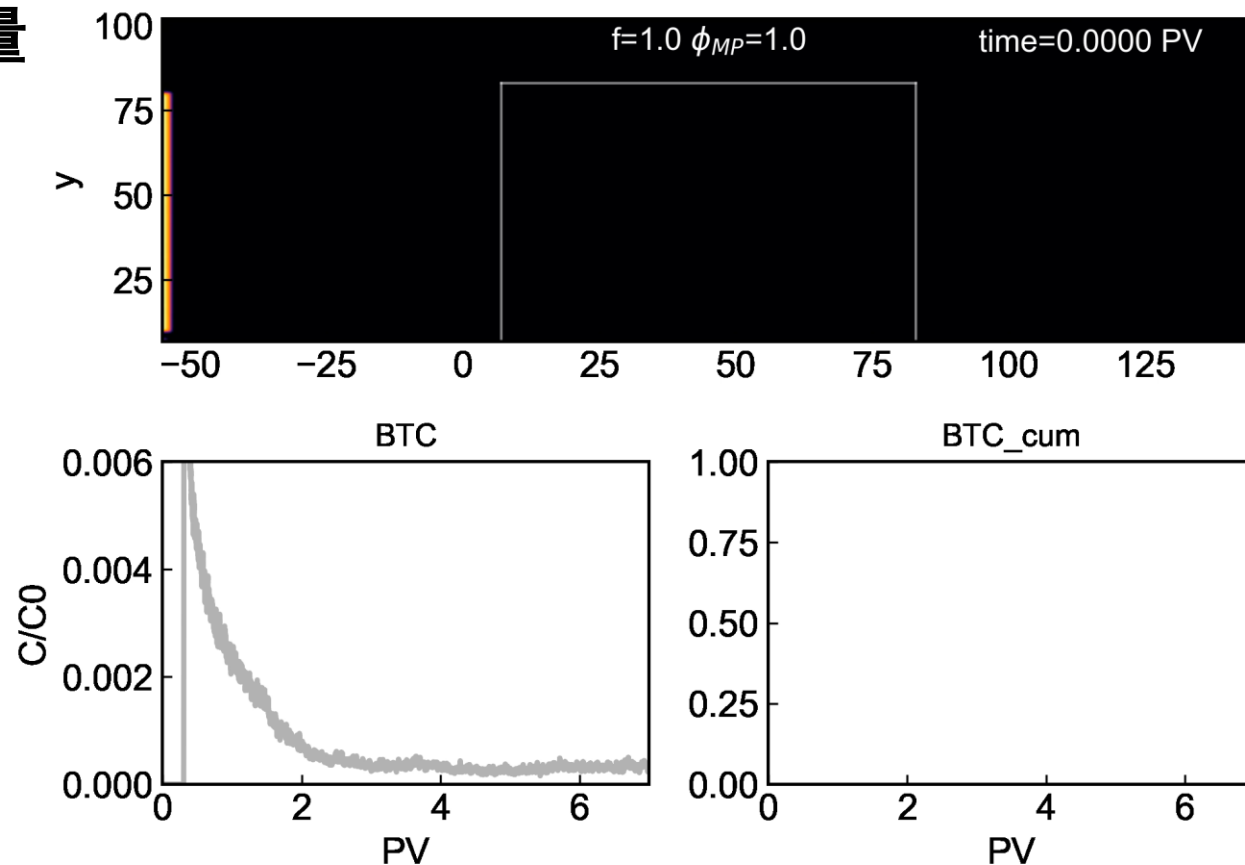
- Numpy 创建1维向量,2维矩阵,3维张量
- nD Array的形状和访问(slice)
- Matplotlib 画线图 1D
- Matplotlib 画点阵 2D
- Matplotlib 画图片 2D
- PyVista 画三维线图 (bonus)



Python科学计算

□ Numpy和Matplotlib入门

- Numpy 创建1维向量,2维矩阵,3维张量
- nD Array的形状和访问(slice)
- Matplotlib 画线图 1D
- Matplotlib 画点阵 2D
- Matplotlib 画图片 2D
- PyVista 画三维线图 (bonus)



Python科学计算

□ 线性代数复习与实现

数据类型

1

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 7 \end{bmatrix} \begin{bmatrix} 3 & 2 \\ 5 & 4 \end{bmatrix}$$

Dot product

$$\mathbf{a} \cdot \mathbf{b} = \sum a_i b_j$$

Cross product

$$\mathbf{a} \times \mathbf{b} = (a_2 b_3 - a_3 b_2, a_1 b_3 - a_3 b_1, a_1 b_2 - a_2 b_1)$$

Matrix Transpose

$$\mathbf{A} = [a_{ij}], \mathbf{B} = \mathbf{A}^T = [b_{ij}], b_{ij} = a_{ji}$$

Matrix Multiplication

$$\mathbf{A} = [a_{ij}], \mathbf{B} = [b_{ij}], \mathbf{C} = [c_{ij}] = \sum a_{ik} b_{kj}$$



□ 线性代数复习与实现 – 点乘Dot product

$$\mathbf{a} \cdot \mathbf{b} = \sum a_i b_i = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$$

算法（虚拟代码）

Input: 向量 $\mathbf{a} = (a_i)_n$, 向量 $\mathbf{b} = (b_i)_n$, $i \leq n$

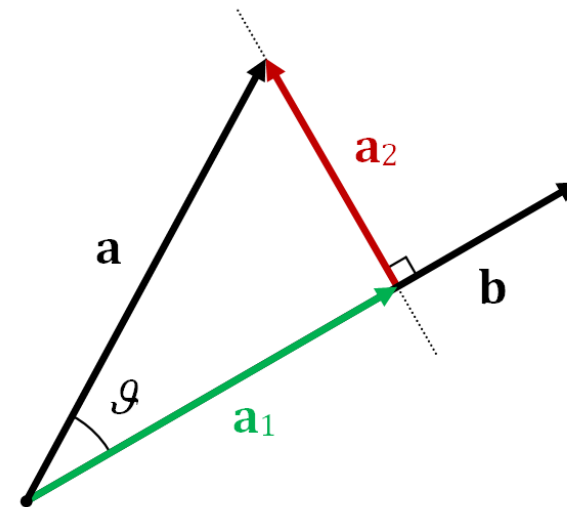
Output: 向量 $\mathbf{c} = (c_i)_n$

Let \mathbf{c} be a new vector

For $i = 1 \dots n$

Set $c_i = a_i b_i$

Return \mathbf{c}



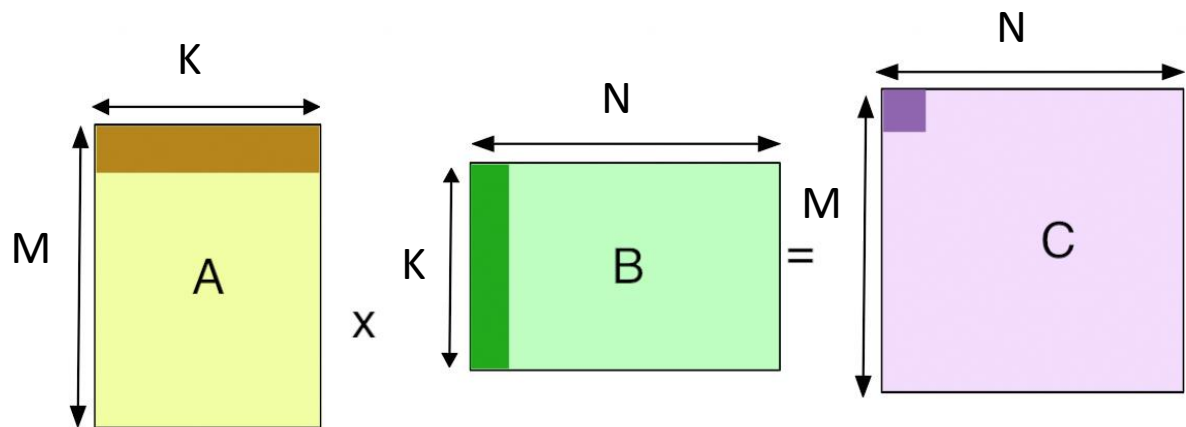
Python科学计算

□ 线性代数复习与实现 – 矩阵乘法Matrix Multiplication

算法（虚拟代码）

$$\mathbf{A} = [a_{ij}], \mathbf{B} = [b_{ij}]$$

$$\mathbf{C} = [c_{ij}] = \mathbf{AB} = \sum a_{ik} b_{kj}$$



<https://petewarden.com/2015/04/20/why-gemm-is-at-the-heart-of-deep-learning/>

Input: $\mathbf{A} = [a_{ik}]_{M \times K}$, $\mathbf{B} = [b_{kj}]_{K \times N}$

Output: $\mathbf{C} = [c_{ij}]_{M \times N}$

Let \mathbf{C} be a new matrix

For $i = 1 \dots M$

 For $j = 1 \dots N$

 Let $\text{sum} = 0$

 For $k = 1 \dots K$

 Set $\text{sum} = \text{sum} + a_{ik} b_{kj}$

$c_{ij} = \text{sum}$

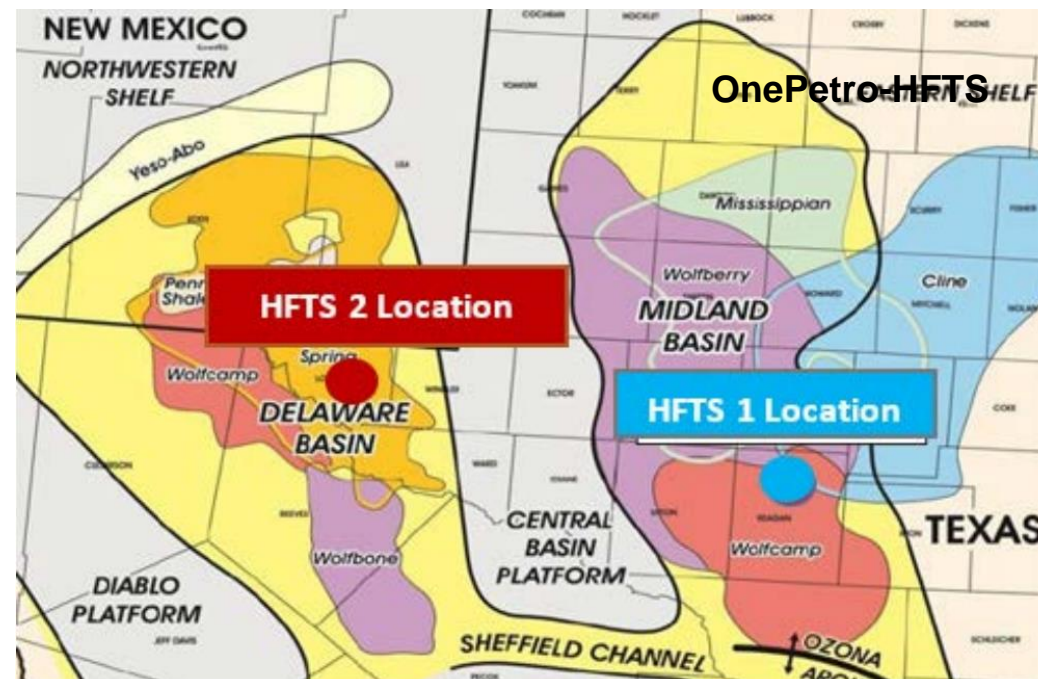
Return \mathbf{C}



Python科学计算

□ 中期Project 预览 – HFTS 12 美国压裂矿场实验

- Permian 盆地，非常规页岩油气
- 11口水平井，1口斜取芯井，1口直探井
- 450段水力压裂
- 全面的水力压裂监测
 - ✓ 微地震，支撑剂示踪，试井分析
 - ✓ 永久式光纤，裂缝面取芯，形貌扫描
 - ✓ 地球化学时域取样，井下压力传感器
- 会议报告29次，现场数据开源



30多家单位联合，累计投入9000万美元

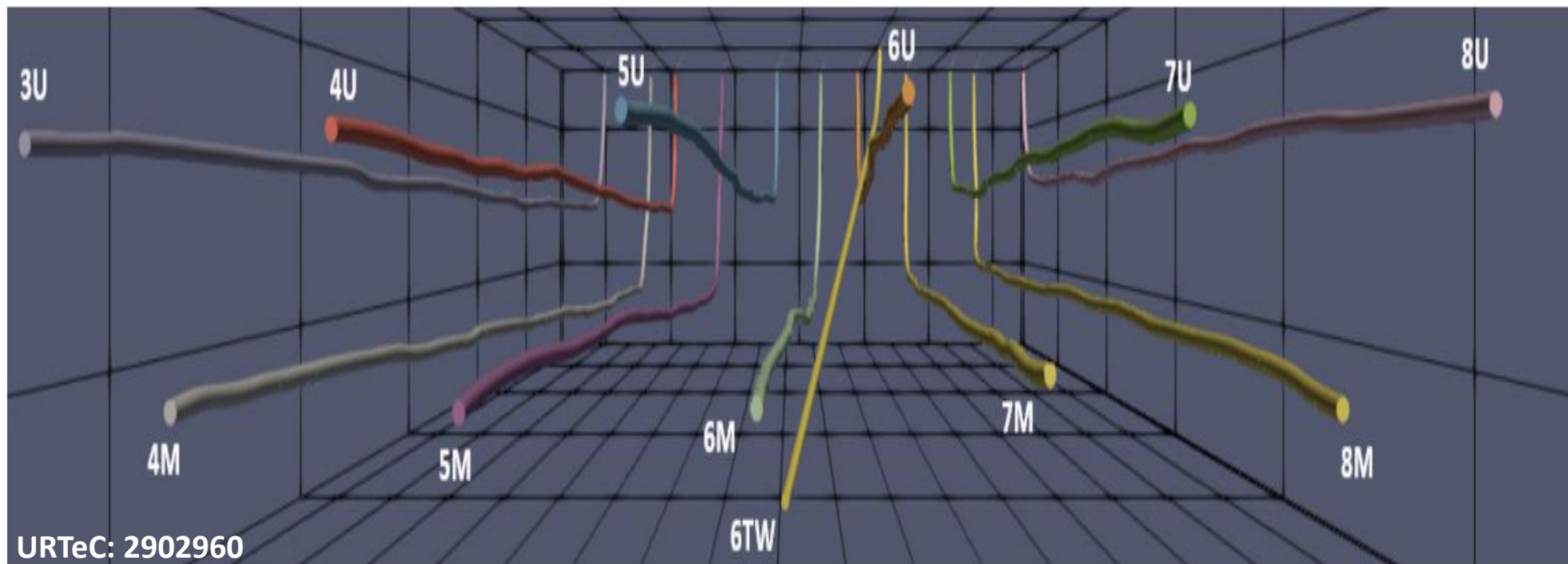


Python科学计算

□ 中期Project 预览 – HFTS 12 美国压裂矿场实验

开源数据: <https://edx.netl.doe.gov/group/hfts-1-phase-1-group>

数据总量: 186 GB

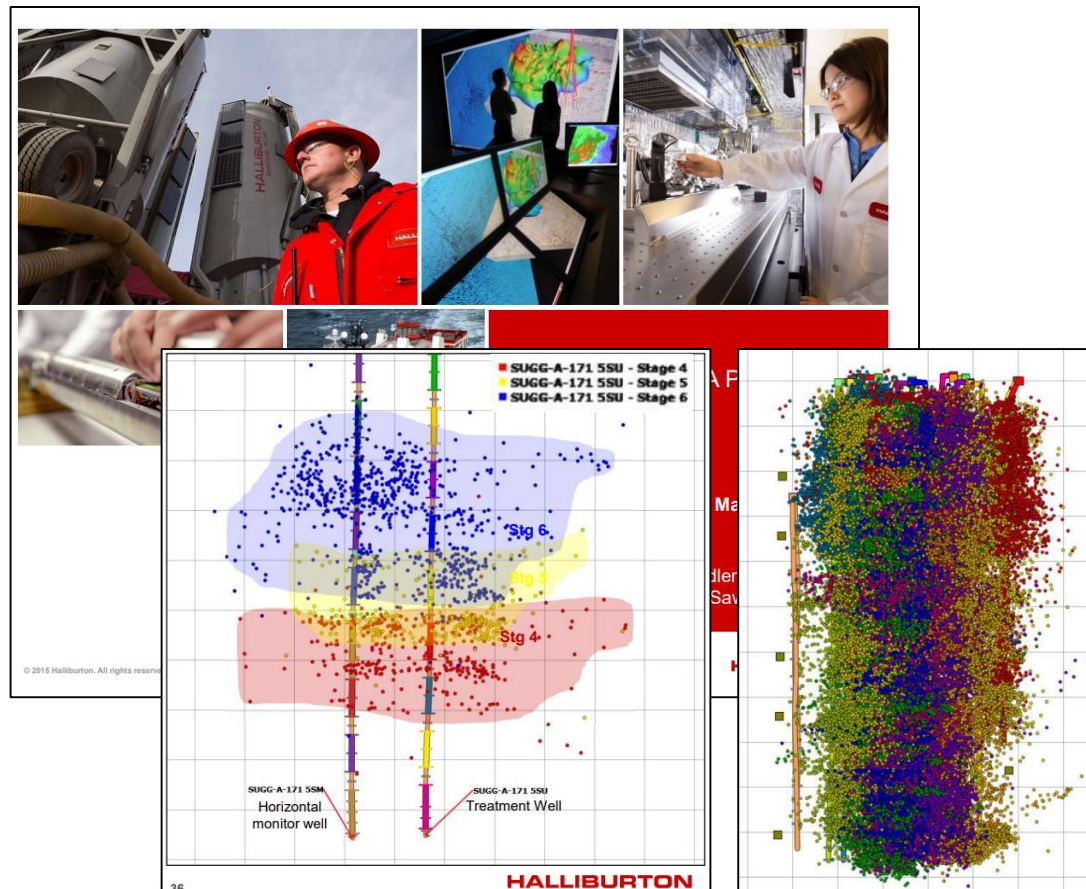


URTeC: 2902960

Python科学计算

□ 中期Project 预览 – HFTS 微地震数据处理及可视化

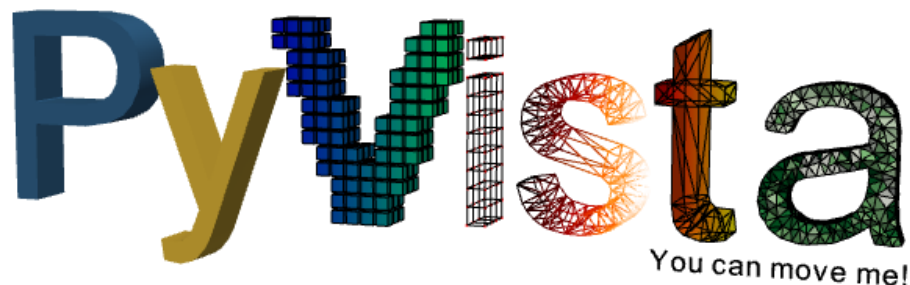
- 组队或者单人（不超过3人）
- OnePetro阅读文献，选择一口井
- openpyxl读单井数据Excel
- Matplotlib画2D井眼轨迹
- Matplotlib微地震散点图
- PyVista三维画图（奖励任务）



Python科学计算

□ 搜索和安装第三方扩展包

- Google 搜索 “关键词” + Python Github
- pip install 安装
- 查看官方案例学习与测试



OpenPyXL 



Python科学计算

□ 学习目标

学习目标	学习成果	效果考察	课程活动
应该掌握哪些知识？	应该能够做哪些事情？	怎么考察学生？	应该怎么学习？
1. Numpy和Matplotlib入门 2. Python实战 <ul style="list-style-type: none">• 向量点乘• 矩阵转置• 矩阵相乘 3. 搜索并使用第三方库(Excel)	1. 能够使用Numpy创建向量和矩阵 2. 能够使用Matplotlib画图 3. 能够手动对简单线性代数问题进行编程 4. 能够安装第三方库	1. 测试学生是否能运行代码 2. 测试Notebook使用情况 3. 课堂测验简单问题 <ul style="list-style-type: none">• 安装pyvista并运行• 随机生成二维点云并画图	1. 观看Python网络课程 2. 完成课后作业7, 8, 9 3. 启动中期Project



□ 线性代数复习与实现 – 叉乘Dot product

$$\mathbf{a} \times \mathbf{b} = (a_2b_3 - a_3b_2, a_1b_3 - a_3b_1, a_1b_2 - a_2b_1)$$

算法（虚拟代码）

Input: 向量 $\mathbf{a} = (a_i)_n$, 向量 $\mathbf{b} = (b_i)_n$, $2 \leq i \leq 3$

Output: 标量 c

Let c be a new variable

Set $c = a_2b_3 - a_3b_2, a_1b_3 - a_3b_1, a_1b_2 - a_2b_1$

Return c

