

Delta-X Applications Workshop Instructions

Table of Contents

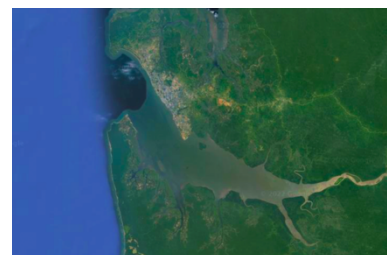
<i>Section 1: Introduction</i>	<i>2</i>
<i>Section 2: Tutorials</i>	<i>3</i>
<i>Section 3: QGIS</i>	<i>4</i>
Installing QGIS	4
<i>Section 4: Google Colab</i>	<i>6</i>
Setting up Colab	6
<i>Section 5: Data Access</i>	<i>8</i>

Section 1: Introduction

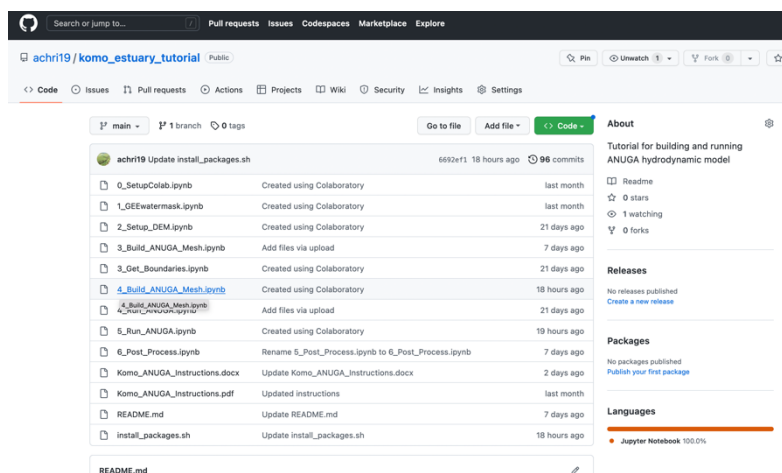
This tutorial introduces the ANUGA modeling software for hydrodynamic modeling. The software was developed by Australian National University and Geoscience Australia. You can find more information on source code, development, and community at https://github.com/GeoscienceAustralia/anuga_core.

Copyright 2004 - 2015 Australian National University and Geoscience Australia. All rights reserved

The tutorials will give an overview of building and running **ANUGA** models, using the Komo River Estuary in Gabon as an example. There are 6 tutorials presented as **Python Jupyter Notebooks** and run within **Google Collaboratory**, an online platform. By using **Google Colab**, we reduce the amount of preparation needed for the tutorials and ensure all participants have a working version of Python.



These instructions will give a brief overview of using **Google Colab** and installing **QGIS**, an open-source software for visualizing geospatial datasets.



https://github.com/achri19/komo_estuary_tutorial

Software:

QGIS – Open-source GIS software for visualizing and processing geospatial data

Python – Open-source software for installing multiple Python tools

ANUGA – Open-source python-based modeling software

Google Earth Engine - Cloud-based geospatial analysis platform

Affiliation: Jet Propulsion Laboratory, California Institute of Technology

Acknowledgement: The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004)

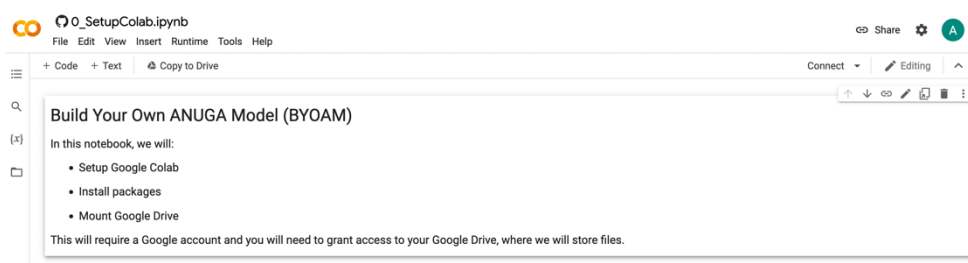
© 2022 California Institute of Technology. Government sponsorship acknowledged

Section 2: Tutorials

There are 6 tutorials in the workshop. The links to the notebooks for this workshop at this Github page. (https://github.com/achri19/komo_estuary_tutorial). All tutorials are designed to run in **Google Colab**, which can be setup using instructions in **Section 4**.

Tutorial 0: Setup Colab

Summary: This notebook will walk through the process of installing packages, connecting to your Google Drive, signing up for **Google Earth Engine**, and using **Google Colab** (a free, online platform for running Python Jupyter Notebooks)



Tutorial 1: GEE Water Masks

Summary: This notebook will introduce **Google Earth Engine** (via **Python**) and produce water masks for the study area. This is a very time-intensive step so all output files will be provided

Tutorial 2: Build Digital Elevation Model

Summary: This notebook will walk through steps to build a DEM for the study area using open-source/publicly available datasets.

Tutorial 3: Boundary Conditions

Summary: This notebook will look at options for setting boundary conditions

Tutorial 4: Mesh Generator

Summary: This notebook will use the **ANUGA** mesh generator to build a uniform, unstructured mesh for the model domain

Tutorial 5: Run ANUGA

Summary: This notebook will run a short **ANUGA** simulation

Tutorial 6: Post-Processing

Summary: This notebook will introduce ways to visualize the model output. We will also use **QGIS** to create animations.

Section 3: QGIS

Installing QGIS


QGIS is a great tool for processing remote sensing data and especially useful for visualizing data quickly and effectively. It is open-source and can be downloaded here:

<https://qgis.org/en/site/forusers/download.html>

For Windows:

Download for Windows

QGIS in OSGeo4W (recommended for regular users):

**OSGeo4W Network Installer**


In the installer choose **Express Install** and select **QGIS** to install the *latest release* or **QGIS LTR** to install the *long term release*. The express installations have several optional packages including non-free software. To avoid those you have to use the **Advanced Install** and choose **qgis** and/or **qgis-ltr** in the desktop section.

CAUTION: Upgrades of old setups from OSGeo4W v1 using this repository are not supported. You need to do a fresh install or use a different directory.

CAUTION: 32 bit binaries are not produced anymore. Also Windows 7 no longer works as we are now using Python 3.9, which dropped support for it.


Standalone installers (MSI) from OSGeo4W packages (recommended for new users)

Latest release (richest on features):

**QGIS Standalone Installer Version 3.24**

sha256

Long term release (most stable):

**QGIS Standalone Installer Version 3.22**

sha256

Note that the MSI installers are much bigger than the previous installers. This is because they include significant larger packages (eg. PROJ 8). The main reason for the switch to MSI were the size limits previously used NSIS has, which was blocking updates of dependencies.

We recommend the standalone installers, specifically Version 3.22

For Mac:



Download for macOS

Official All-in-one, signed installers



Mac Installer Packages for macOS High Sierra (10.13) and newer.

QGIS is not yet notarized as required by macOS Catalina (10.15) security rules. On first launch, please right-click on the QGIS app icon, hold down the Option key, then choose Open.

Latest release (richest on features):


[QGIS macOS Installer Version 3.24](#)




Long term release (most stable):


[QGIS macOS Installer Version 3.22](#)


Alternative build

Mac Installer Packages for macOS High Sierra (10.13) and newer.

Installation instructions are in the Read Me on the disk image. GDAL and Python (both included on the disk image) are installed separately and outside the QGIS app so they are usable on their own. These packages use the python.org Python 3 - other distributions are not supported.


[macOS Installers](#)


We recommend Version 3.22

For Linux:

Download for Linux

For many flavors of GNU/Linux binary packages (rpm and deb) or software repositories (to add to your installation manager) are available. Please select your choice of distro below:

- [Debian/Ubuntu](#)
- [Fedora](#)
- [openSUSE](#)
- [RHEL, CentOS, Scientific Linux, ...](#)
- [Mandriva](#)
- [Slackware](#)
- [Arch Linux](#)
- [Flatpak](#)

[Linux Installation Instructions](#)

Common issues:

- If you're using a PC and get error 2503 or 2503, you need to set permissions on C:\\WINDOWS\\TEMP to Authenticated users = Full Control
 - More info [here](#)

Section 4: Google Colab

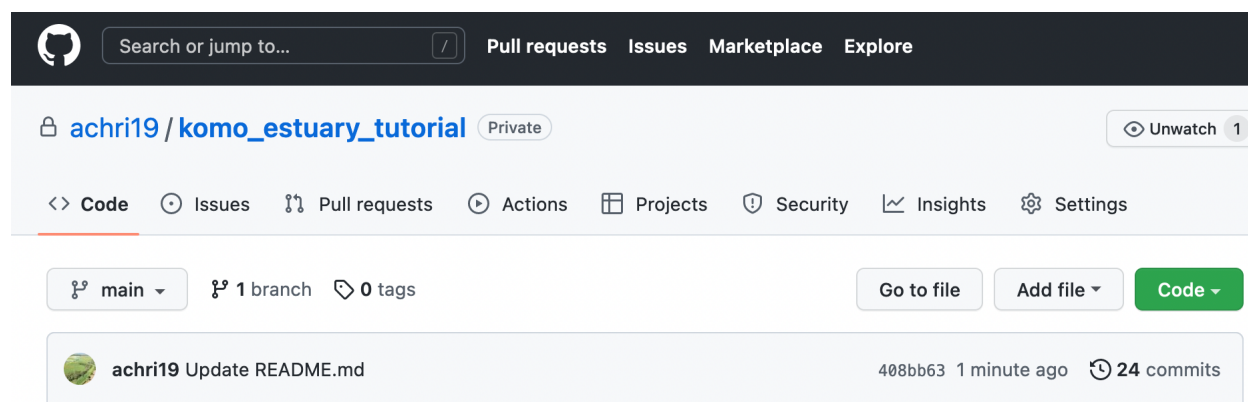
Setting up Colab

For this tutorial, we will be using **Google Collaboratory**, which allows us to run **Python Jupyter Notebooks** online without installing complicated software.

Google Colab is similar to **Python Jupyter Notebooks**, but is run on the Google cloud and comes with many packages already installed. Therefore, it's a great way to teach tutorials for this workshop. The only requirement will be allowing **Colab** access to your Google Drive. Unfortunately, **Colab** cannot access the files on your computer, so we will be accessing data through the shared Google Drive.

Please go through the following steps to practice using **Colab**.

- 1) Go to https://github.com/achri19/komo_estuary_tutorial to find the Github site where we are hosting the test notebook.

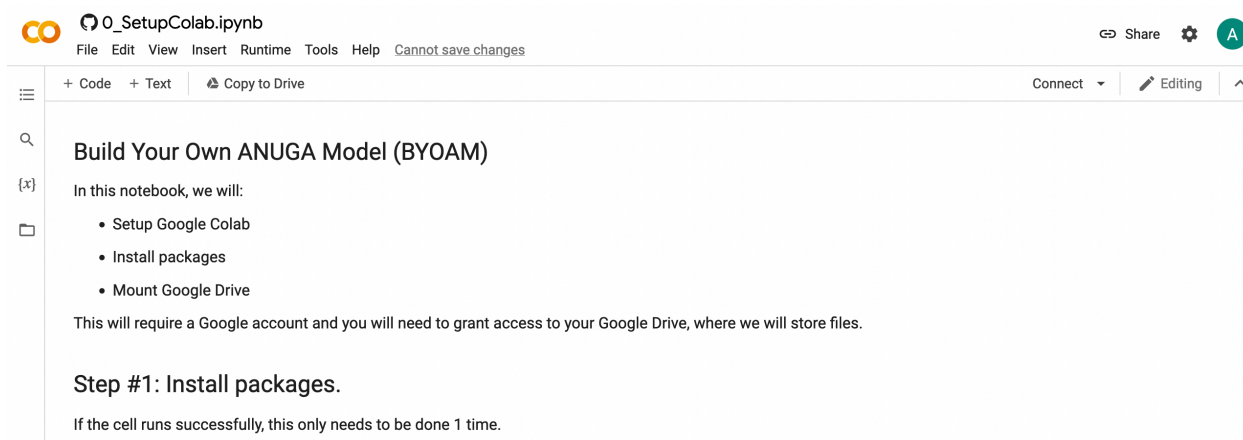


You do not need a Github account to access this repository.

- 2) Click on the “Open in **Colab**” button within the README section.

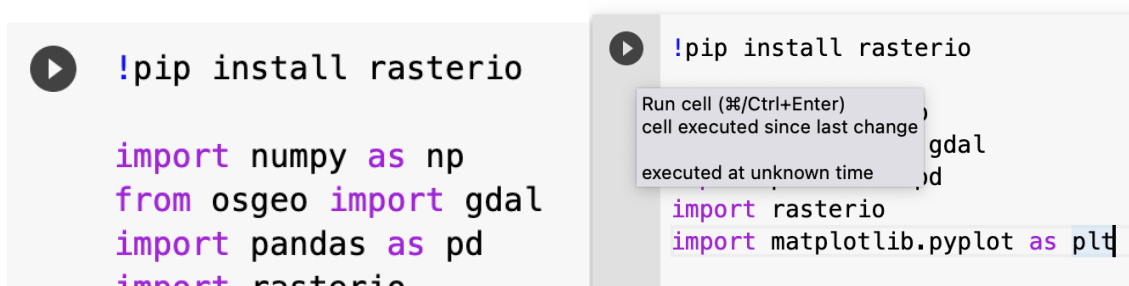


- 3) This will open a new window and a beautiful test notebook.



The screenshot shows a Google Colab notebook interface. At the top, the title is '0_SetupColab.ipynb'. Below the title bar, there are tabs for 'Code', 'Text', and 'Copy to Drive'. The main content area has a search icon and a list of files. The first file is 'Build Your Own ANUGA Model (BYOAM)'. Below the file name, it says 'In this notebook, we will:' followed by a bulleted list: 'Setup Google Colab', 'Install packages', and 'Mount Google Drive'. Below the list, it says 'This will require a Google account and you will need to grant access to your Google Drive, where we will store files.' Below that, it says 'Step #1: Install packages.' and 'If the cell runs successfully, this only needs to be done 1 time.'

- 4) Read the instructions and run through each cell (snippet of code separated into boxes)
- 5) When you hover over the first gray box, you will see a “play” button in the upper left corner. **Click it** to run the first cell. You can also use a command to run this cell, often it’s SHIFT + ENTER or CTRL+ENTER, depending on your system. If you hover over the “play” button, it will tell you what the command is.



The image shows two side-by-side code snippets from a Colab notebook. The left snippet is a code cell with a play button icon in the top left corner. The code is:

```
!pip install rasterio

import numpy as np
from osgeo import gdal
import pandas as pd
import rasterio
```

The right snippet is a code cell with a play button icon in the top left corner. The code is:

```
!pip install rasterio

import rasterio
import matplotlib.pyplot as plt
```

Overlaid on the right snippet is a tooltip that says: 'Run cell (⌘/Ctrl+Enter) cell executed since last change', 'executed at unknown time', and 'gdal', 'pd'.

If you run into any problems with this process, please reach out to alexandra.l.christensen@jpl.nasa.gov

Section 5: Data Access

All data used in this tutorial are publicly available. Some data are provided in the Google Drive and some are downloaded within the notebooks. Due to the long processing time of some steps, we are providing some pre-processed datasets.

More coming soon...