Python Code description and dependencies

# Jupyter Lab Python Environment

This project is built and run within the JupyterLab environment. Below are the instillation steps for JupyterLab and the python dependencies and libraries the code requires to run.

## Instillation

Pip install jupyterlab

Launch JupyterLab from CMD window:

* Jupyter-lab

Jupyter home page

* <https://jupyter.org/>

# Python Dependencies for both two- and three-phase codes

From the command window use: *python -m pip install* …

## Matplotlib dependency

### Installation

pip install -U matplotlib

* <https://matplotlib.org/stable/users/installing/index.html>

## Plotly Dependencies

### installation

pip install plotly

* <https://plotly.com/python/getting-started/>

## Kaleido [static image export]

### Instillation

pip install -U kaleido

* <https://plotly.com/python/static-image-export/>
* <https://github.com/plotly/Kaleido>

## Scipy Dependencies

### Instillation

pip install scipy

* <https://scipy.org/install/>

## Numpy Dependencies

### Instillation

pip install numpy

* <https://numpy.org/install/>

## Csv loader Dependencies

### Instillation

pip install python-csv

* <https://pypi.org/project/python-csv/>

## Easygui Dependencies

### Instillation

pip install easygui

* <https://pypi.org/project/easygui/>

## Pandas Dependencies

### Instillation

pip install pandas

* <https://www.pythoncentral.io/how-to-install-pandas-in-python/>

## WX.Python Dependencies (GUI)

### Instillation

pip install -U wxPython

* <https://wiki.wxpython.org/How%20to%20install%20wxPython>

## ipywidgets: Interactive HTML Widgets

### Instillation

pip install ipywidgets

* **ipywidgets**, also known as jupyter-widgets or simply widgets, are [interactive HTML widgets](https://github.com/jupyter-widgets/ipywidgets/blob/master/docs/source/examples/Index.ipynb) for Jupyter notebooks and the IPython kernel.
* This package contains the python implementation of the core interactive widgets bundled in ipywidgets.
* <https://pypi.org/project/ipywidgets/>

**Troubleshoot:** If the code errors on the ipywidgets then run the following code to downgrade the ipywidgets and jupyterlab-widgets

Error code:

* AttributeError: type object 'DOMWidget' has no attribute '\_ipython\_display\_'

Downgrade these packages using the following code:

1. python -m pip install ipywidgets==7.7.1
2. python -m jupyterlab-widgets==1.1.1

* <https://github.com/polakowo/vectorbt/issues/488>

# Input Data format

Input data MUST be a CSV file (coma separated values) with no headers with the following data:

* Pressure (psi), Mercury saturation (decimal)

[Example data format]

3.480912, 0.0

4.061064, 0.01

4.786254, 0.017

5.80152, 0.028

6.671748, 0.046

7.687014, 0.064

8.70228, 0.077

10.442736, 0.103

…

# Two-phase code Description from Jupyter Lab

**PC-to-RP Method**

Two-Phase Relative Permeability from Capillary Pressure Calculator

Author: *Dr. Nathan Moodie*

## SECTION 1

1. Load Required Python Packages and Tools
2. Load Input Data
   1. Read file and parse data into variables 'pc' and 'shg'
      1. Data file must be in CSV format with NO headers.
      2. Data file must contain capillary pressure in the first column and mercury saturation in the second column
3. Set Constants for the algorithm
   1. MICP porosity and permeability
   2. Interfacial Tension [gamma] (gas/water)
   3. Contact Angle [theta] (gas/water)
   4. *b* exponent (gas and water)\*
   5. *c* exponent (gas and water)\*
      1. \*Values used are estimated from laboratory measured relative permeability published in the literature [Bennion and bachu]
4. Plot Input Data and Constants

## SECTION 2a

Calculate the effective permeability and first and second derivative of the MICP data

1. Set lithology factor τ and plot effective permeability (ke)
   1. Lithology factor (τ) in Purcell’s ke formula is an unknown
   2. Minimization algorithm adjusts τ so that the maximum effective permeability (ke\_max) = permeability (k) from the MICP data
   3. Plot ke and ke\_max
2. Calculate Effective Permeability in the gas/water fluid pair (ke-g/w)
   1. Identify maximum effective permeability for each fluid pair using Purcell's ke formula
      1. Purcell, W R. “Capillary Pressures - Their Measurement Using Mercury and the Calculation of Permeability Therefrom.” Journal of Petroleum Technology 1, no. 2 (1949): 39–48.
3. Calculate 1st and 2nd derivatives and angle of the 1st derivative of the capillary pressure data
   1. Identify water saturation (Sw) critical points
   2. Sw\_max is where the sign of the second derivative goes from negative to positive
   3. Sw\_min is more difficult - Coded the following two options that use the algorithm to identify the points
      1. Sw\_min is set to the value at 89 degrees = swmin
      2. Sw\_min is set to the value at maximum change in angle of the first derivative
   4. Identify effective permeability critical points
      1. effective water (kew) in gas/water pair identified at Sw\_max point
4. Plot ke data and critical points

## SECTION 2b

1. Effective Non-Wetting Permeability Algorithm
   1. Identify effective nonwetting-phase permeability critical points (keg)
      1. Swanson (1981) theorized that the point where the nonwetting-phase is flowing through the bulk of the pore network can be identified from the MICP data by plotting the log bulk volume vs. log capillary pressure
         1. Swanson, B F. “A Simple Correlation between Permeabilities and Mercury Capillary Pressures.” Journal of Petroleum Technology 33, no. 12 (1981): 2,492-498,504.
      2. The algorithm identifies this point at 1/2\*angle between trendline1 and trendline2
         1. where trendline1 = early trend in the data over the first few data points after the first
         2. always skip the first data point
   2. IMPORTANT: Select ONLY TWO points for trendline 1 from the following chart.
   3. Trendline 1 is defined by selecting two points from the chart
   4. The user must select only two points and move to the next cell manually
2. Plot Results
   1. Table of keg calculated data
   2. Plot of trendline data for keg identification and values
   3. Table of variables calculated from MICP data

## SECTION 2c

1. Critical Endpoint UI GUI
   1. Ask User how to identify critical endpoints
      1. Define maximum gas and water rel perm under 2-phase flowing conditions
      2. Define gas/water relative permeability at Sw=1 and Sw=0
      3. Select minimum water saturation in gas/water pair [Sw\_min]
      4. Select maximum water saturation in gas/water pair [Sw\_max]
   2. If user select to use algorithm endpoint values, skip the next section
   3. If user selects to pick saturation endpoints from the chart, then go to the next section.
      1. Select minimum and maximum saturation points TWO POINTS ONLY
         1. if the user hasn't selected to pick points from the chart the code will not read any inputs selected from this chart
         2. Move to next code box manually after selection
      2. click on "CLICK HERE NEXT"

## SECTION 2d

1. Calculate Saturation Ranges
   1. Saturation of the MICP tests (or other lab data) likely contains more data points than are practical for a simulator input file
   2. Input data needs to be re-mapped to the desired number of relative permeability/saturation data points for the kr formula
2. Calculate Binary-Pair Capillary Pressure
   1. Gas/Water capillary pressure conversion from mercury/air to oil/water and gas/oil fluid system
3. Plot Binary-Pair Capillary Pressure Data
   1. Left Chart
      1. Mercury/air pressure/saturation plot
      2. gas/water pressure/saturation plot
      3. critical pressure/saturation endpoints
   2. Right-Top Chart
      1. Effective non-wetting permeability logBv/logPc plot
   3. Right-Bottom Chart
      1. Effective gas/water permeability/mercury saturation plot

## SECTION 3

1. Relative Permeability Calculator
   1. calculate s\_bar for CO2/water fluid pair
   2. calculate CO2/water relative permeability
2. Display CO2/water relative permeability chart
3. Output data to file
   1. Three available file formats for exported data
      1. csv file
      2. Eclipse GSF/WSF keywords
      3. Eclipse SGWFN keyword

# Three-phase code Description from Jupyter Lab

**PC-to-RP Method**

Three-Phase Relative Permeability from Capillary Pressure Calculator

Author: *Dr. Nathan Moodie*

## SECTION 1

1. Load Required Python Packages and Tools
2. Load Input Data
   1. read file and parse data into variables 'pc' and 'shg'
      1. Data file must be in CSV format of [pressure, saturation] with NO headers.
      2. Data file must contain capillary pressure in the first column and mercury saturation in the second column
3. Set Constants for the algorithm
   1. enter MICP porosity and permeability
   2. Interfacial Tension [gamma] (oil/water and gas/oil)
   3. Contact Angle [theta] (oil/water and gas/oil)
   4. ***b*** exponent (oil in water, water, gas in oil, oil)
   5. ***c*** exponent (oil in water, water, gas in oil, oil)
4. Plot Input Data and Constants

## SECTION 2a

Calculate the effective permeability and first and second derivative of the MICP data

1. Set Lithology Factor τ
   1. Minimization algorithm adjusts τ to minimize the miss-match between max ke and MICP k
2. Calculate Effective Permeability in the oil/water fluid pair (ke-o/w) and in the gas/oil fluid pair (ke-g/o)
   1. Identify maximum effective permeability for each fluid pair using Purcell's formula: oil/water = keow\_max & gas/oil = kego\_max
      1. Purcell, W R. “Capillary Pressures - Their Measurement Using Mercury and the Calculation of Permeability Therefrom.” Journal of Petroleum Technology 1, no. 2 (1949): 39–48.
3. Calculate 1st and 2nd derivatives and angle of the 1st derivateive of the capillary pressure data
   1. Identify water saturation (Sw) and total liquid saturation (Sl = oil + residual water) critical points
      1. Sw\_max and Sl\_max is where the sign of the second derivative goes from negative to positive
      2. Sw\_min is more difficult - Coded the following two options
         1. Sw\_min is set to the value at 89 degrees = swmin
         2. Sw\_min is set to the value at maximum change in angle of the first derivative
      3. Sl\_min = (1 - Sw\_max) + Sw\_min
   2. Identify effective permeability critical points
      1. effective permeability of water (kew) in oil/water pair and effective oil permeability (kego) in gas/oil pair identified at Sw\_max & Sl\_max points
4. Plot ke data and identify maximum points

## SECTION 2b

1. Effective Non-Wetting Permeability Algorithm
   1. Identify effective nonwetting-phase permeability critical points (keow & keg)
      1. Swanson's bulk flow hypothesis for identifying the critical non-wetting-phase saturation applied to the MICP data
         1. Swanson, B F. “A Simple Correlation between Permeabilities and Mercury Capillary Pressures.” Journal of Petroleum Technology 33, no. 12 (1981): 2,492-498,504.
      2. The algorithm will identify the point on a chart of log bulk volume vs. log MICP capillary pressure where it is theorized that the bulk of the nonwetting phase is flowing
      3. This point corresponds to the maximum effective permeability of the non-wetting phase(s)

IMPORTANT: Select ONLY TWO points for trendline 1 from the following chart.

* 1. Trendline 1 defines the early trend in the data over the first 2 to 10 data points
     1. These points are used to identify the non-wetting effective permeability (keg and keow)
  2. The user must select only two points and move to the next cell manually

1. Plot Results
   1. Table of keow and keg calculated data
   2. Plot of trendline data for keow and keg identification and values
   3. Table of variables calculated from MICP data