2-Day Course – Spatial Modeling with Geostatistics

Prof. Michael J. Pyrcz, Ph.D., P.Eng.
Associate Professor

Hildebrand Department of Petroleum & Geosystems Engineering
University of Texas at Austin

Bureau of Economic Geology, Jackson School of Geosciences
University of Texas at Austin

"In two days, what a geoscientists needs to know about geostatistics, and workflows to get you started with applying geostatistics to impact your work."

Spatial Modeling with Geostatistics

Representative Sampling

Lecture outline . . .

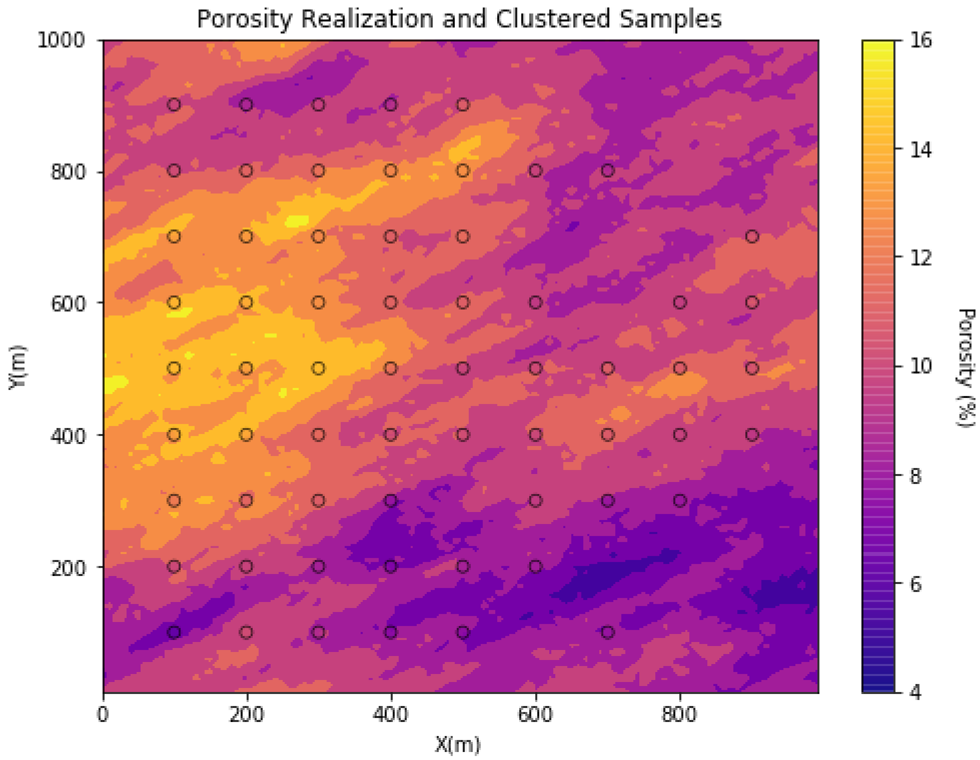
- Motivation
- Cell Declustering
- Debiasing with Secondary Data

PGE 383 Lecture 3: Representativity

Lecture outline . . .

- **Motivation**
- **Cell Declustering**
- **Declustering by Global Estimation**
- **Debiasing with Secondary Data**

Spatially Clustered Data



What is wrong with this sample set?

Spatially Clustered Data

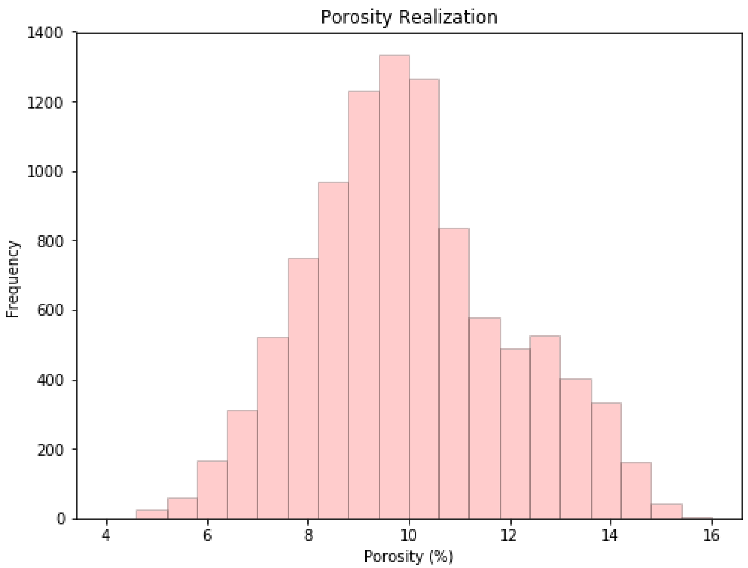
Data are rarely collected for their statistical representativity:

- Wells are drilled in areas with the greatest probability of high production
- Horizontal wells target stratigraphic zones of interest (high pay)
- Core are taken preferentially from good quality reservoir rock
- Certain areas are more accessible
- These “data collection” practices should not be changed; they lead to the best economics and the greatest number of data in portions of the study area that are the most important

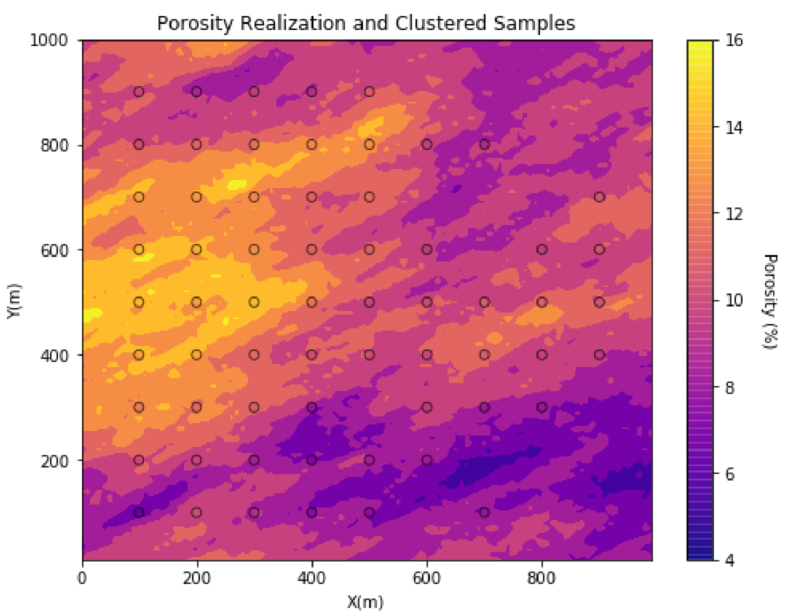
Spatially Clustered Data

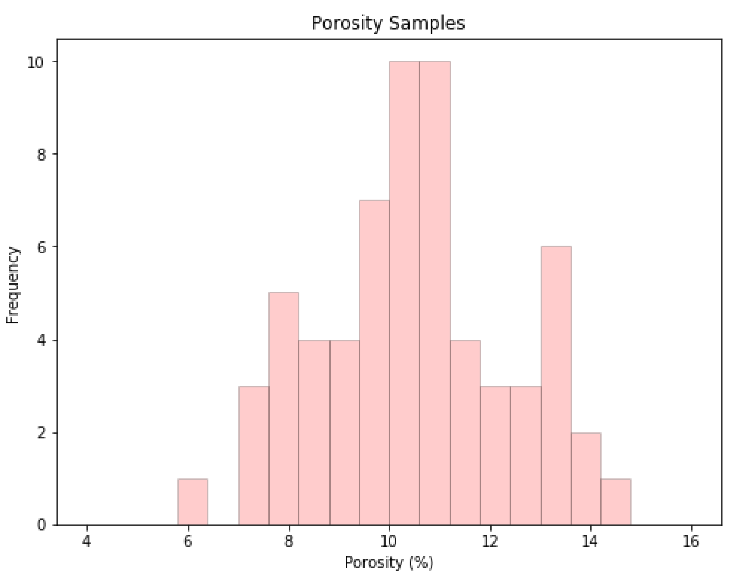
- There is a need, however, to adjust the histograms and summary statistics to be representative of the entire volume of interest.
- Declustering techniques assign each datum a weight based on closeness to surrounding data
  - $w_i, i = 1, \dots, n$ (weights are greater than 0 and sum to n)
  - Histogram and cumulative histogram use $w_i, i = 1, \dots, n$ instead of equal weighted, $w_i = 1.0$.
- Debiasing techniques derive an entirely new distribution based on a secondary data source such as geophysical measurements or expert interpretation

Some Clustered Data



- Location map of 64 wells. with truth model.
- See the error between the samples and the underlying truth model.

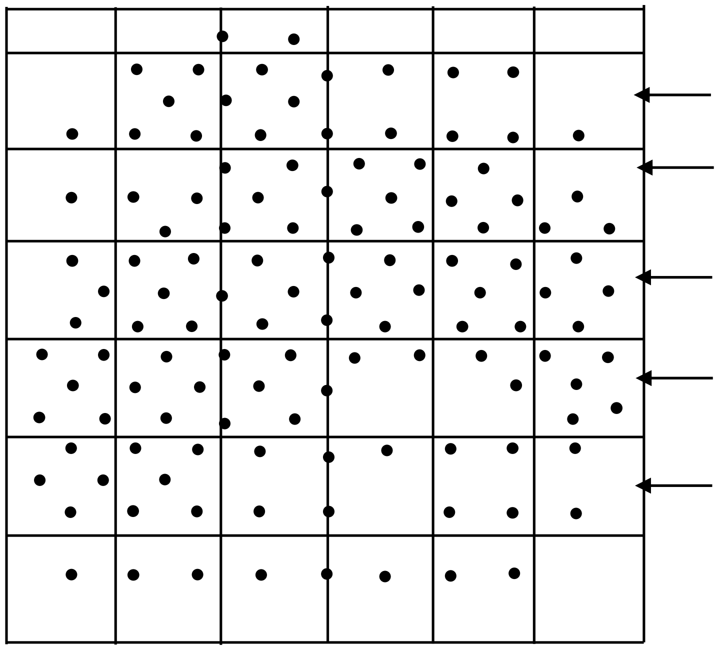




Truth Mean = 10.0 , Clustered Sample Mean = 10.48 , Error = 4.8 %

Cell Declustering

- Another technique, called *Cell Declustering*, is robust in 3-D and when the limits are poorly defined:
  - divide the volume of interest into a grid of cells $l=1,\ldots,L$
  - count the occupied cells L_o and the number in each cell n_l, $l=1,\ldots, L_o$
  - weight inversely by number in cell (standardize by L_o)



1 weight x (118 data / 36 cells) = 3.27

$^1/_3$ weight x (118 data / 36 cells) = 1.09

$^1/_4$ weight x (118 data / 36 cells) = 0.82

$^1/_5$ weight x (118 data / 36 cells) = 0.65

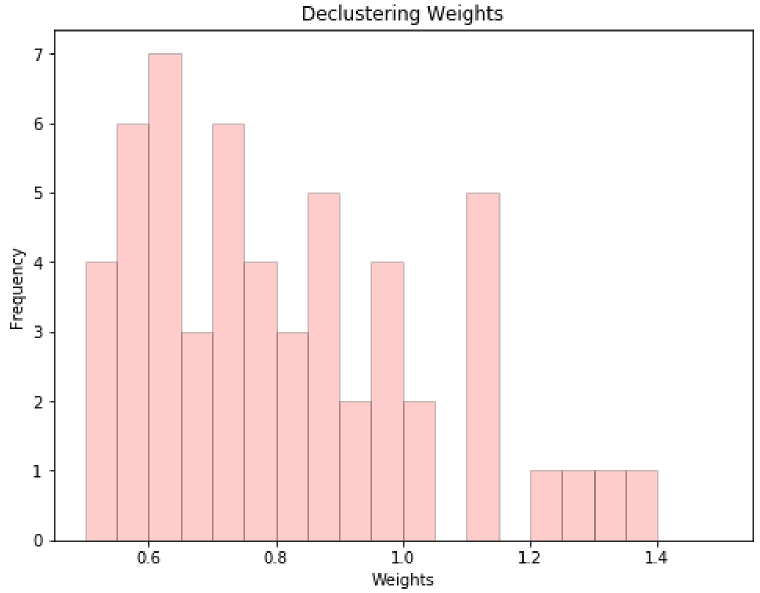
$^1/_2$ weight x (118 data / 36 cells) = 1.63

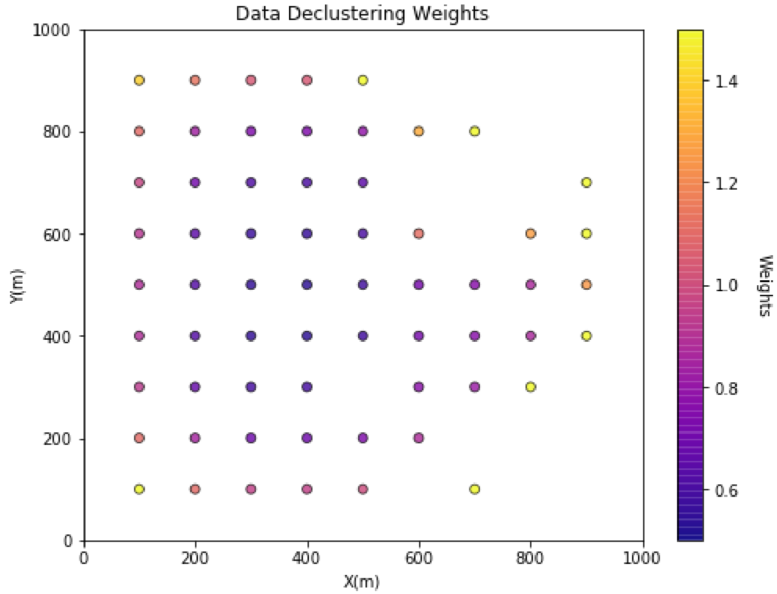
Sum of all weights = n
Nonclustered weight = 1.0

- The issue, of course, is how to choose the cell size…

Declustering Weights

- Declustering weights
  1. 1.0 nominal weight
  2. < 1.0 reduced weight
  3. > 1.0 increased weight

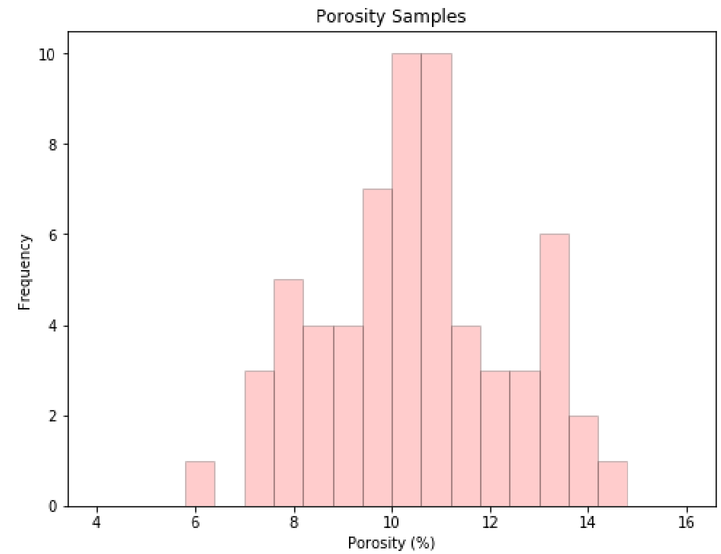




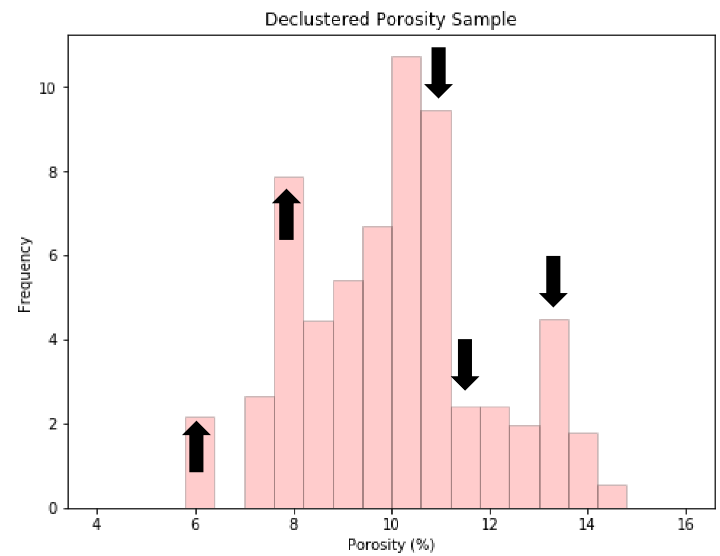
Declustered Distribution

- Updated distribution with declustering weights
- Now data file include values and weights based on spatial arrangement.
- Possible to calculate any weighted statistic.
- For example declustered mean:

$$\bar{z} = \frac{\sum_i^n w_i z_i}{\sum_i^n w_i = n}$$



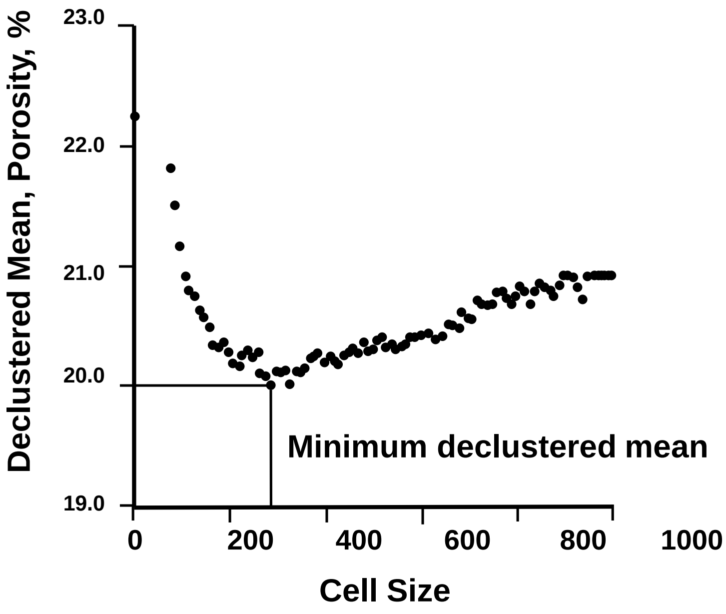
Truth Mean = 10.0 , Clustered Sample Mean = 10.48 , Error = 4.8 %

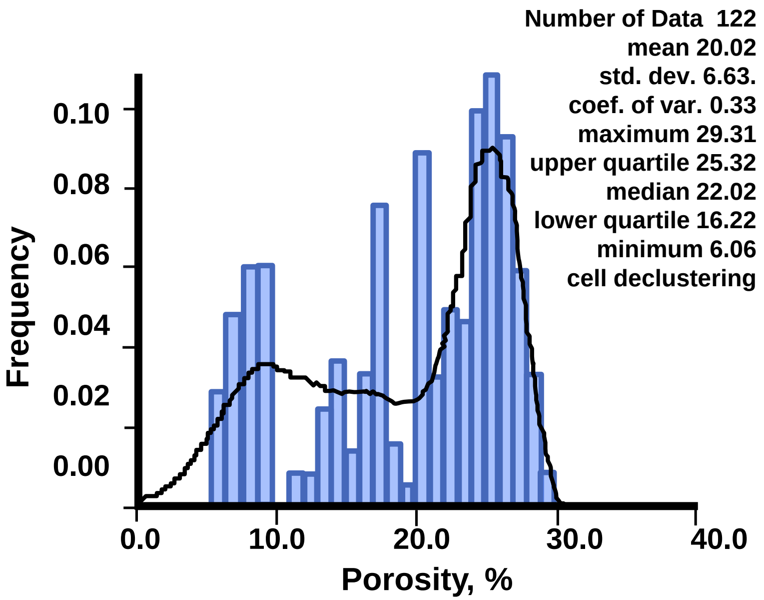


Truth Mean = 10.0 , Clustered Sample Mean = 10.48 , Error = 5.0 %
Declustered Mean = 10.07 , Error = 1.0 %

The Cell Size

- Plot declustered mean versus the cell size for a range of cell sizes:





- There is no theory that says we are looking for a minimum when the values are clustered in high values or a maximum when clustered in low values – it just seems to make sense
- The result can be very sensitive to large scale trends – it is often better to choose the cell size be visual inspection and some sensitivity studies

Python / GSLIB Declustering Demo

- Jupyter Notebook
  - Available with Anaconda (https://www.anaconda.com/)
  - HTML-based
  - A sequence of fields for code, output and documentation (Markdown)
    - » You run each field in sequence
  - Can install a kernel (the ability to work with a programming language), default is Python with Anaconda install.

- Python with GSLIB 'wrapped'
  - I will demonstrate this
  - There is not a good declustering option in R
  - I wrapped by calling Windows executables and reading in the result and building Python MatPlotLib visualizations

- Powerful open source option

Python / GSLIB Declustering Demo

Spatial Declustering in Python for Engineers and Geoscientists

Michael Pyrcz, Associate Professor, University of Texas at Austin

Contacts: Twitter/@GeostatsGuy | GitHub/GeostatsGuy | www.michaelpyrcz.com | GoogleScholar | Book

This is a tutorial for / demonstration of **spatial declustering in Python with simple wrappers and reimplementations of GSLIB: Geostatistical Library methods** (Deutsch and Journel, 1997). Almost every spatial dataset is based on biased sampling. This includes clustering (increased density of samples) over specific ranges of values. For example, more samples in an area of high feature values. Spatial declustering is a process of assigning data weights based on local data density. The cell-based declustering approach (Deutsch and Journel, 1997; Pyrcz and Deutsch, 2014; Pyrcz and Deutsch, 2003, paper is available here: http://gaa.org.au/pdf/DeclusterDebias-CCG.pdf) is based on the use of a mesh over the area of interest. Each datum's weight is inverse to the number of data in each cell. Cell offsets of applied to smooth out influence of mesh origin. Multiple cell sizes are applied and typically the cell size that minimizes the declustered distribution mean is applied for preferential sampling in the high-valued locations (the maximizing cell size is applied if the data is preferential sampled in the low-valued locations). If there is a nominal data spacing with local clusters, then this spacing is the best cell size.

This exercise demonstrates the cell-based declustering approach in Python with wrappers and reimplimentation of GSLIB methods. The steps include:

1. generate a 2D sequential Guassian simulation using a wrapper of GSLIB's sgsim method
2. apply regular sampling to the 2D realization
3. preferentially removing samples in the low-valued locations
4. calculate cell-based declustering weights
5. visualize the location map of the declustering weights and the original exhaustive, sample and the new declustered distributions.

To accomplish this I have provide wrappers or reimplementation in Python for the following GSLIB methods:

1. sgsim - sequantial Gaussian simulation limited to 2D and unconditional
2. hist - histograms plots reimplemented with GSLIB parameters using python methods
3. locmap - location maps reimplemented with GSLIB parameters using python methods
4. pixelplt - pixel plots reimplemented with GSLIB parameters using python methods
5. locpix - my modification of GSLIB to superimpose a location map on a pixel plot reimplemented with GSLIB parameters using Python methods
6. affine - affine correction adjust the mean and standard deviation of a feature reimplemented with GSLIB parameters using Python methods

These methods are all in the functions declared upfront. To run this demo all one has to do is download and place in your working directory the following executables from the GSLIB/bin directory:

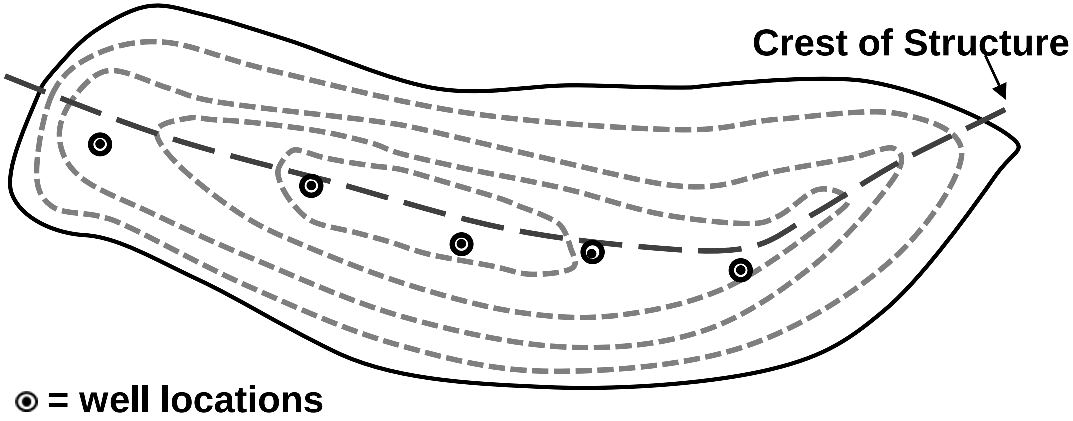
https://github.com/GeostatsGuy/PythonNumericalDemos/blob/master/Declustering.ipynb

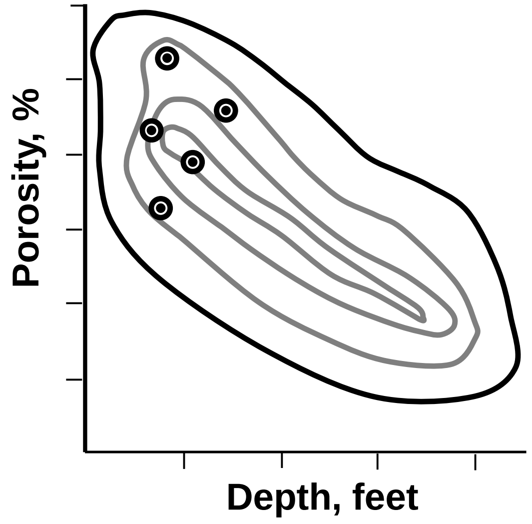
Comments on Cell Declustering

- Fixing the cell size and changing the origin often leads to different declustering weights. To avoid this artifact, a number of different origin locations should be considered for the same cell size. Average the declustering weights for each origin offset
- Perform an areal 2-D declustering when the wells are vertical
- Consider 3-D declustering when there are horizontal or highly deviated wells that preferentially sample certain stratigraphic intervals
- The shape of the cells depends on the geometric configuration of the data - adjust the shape of the cells to conform to major directions of preferential sampling
- Could select the cell size, choose the cell size so that there is approximately one datum per cell in the sparsely sampled areas, the nominal spacing

Comments on Spatial Debiasing

- What do we do when there are too few data or the data are not representative?
- Nothing, unless there is some secondary information



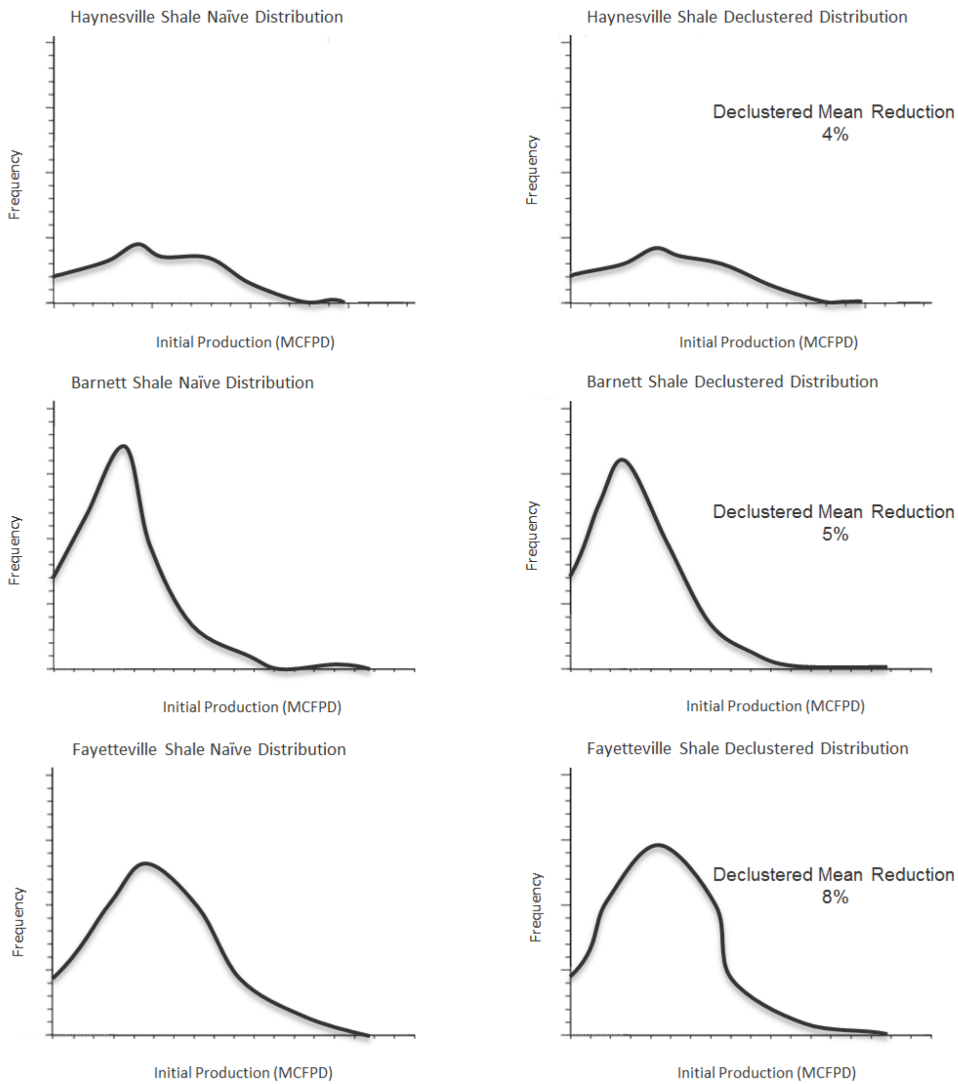


- How can we use this soft data to correct histogram?
  - Extrapolate porosity data using the full depth distribution

Declustering in Unconventionals

Representative Statistics

- Compiled IP datasets for domestic shale plays
  - Filtered datasets to reduce influence of completions
- Representativity an issue even with large datasets and relatively good coverage
  - Observed changes in naïve to declustered means of 4 – 8%



Naïve and declustered distributions from cell-based declustering (modified from Pyrcz et al, 2017)

Probability and Statistics
New Tools

Topic	Application to Subsurface Modeling
Cell Declustering	Given the spatial location of the sample data, calculate declustering weights. *Build representative sample statistics that correct for sampling bias.*
Spatial Debiasing	Use secondary information, such as seismic and facies to calculate a more representative distribution. *Use depth information and knowledge of a compaction trend to build a representative distribution for porosity.*

Spatial Modeling with Geostatistics

Representative Sampling

Lecture outline . . .

- Motivation
- Cell Declustering
- Debiasing with Secondary Data