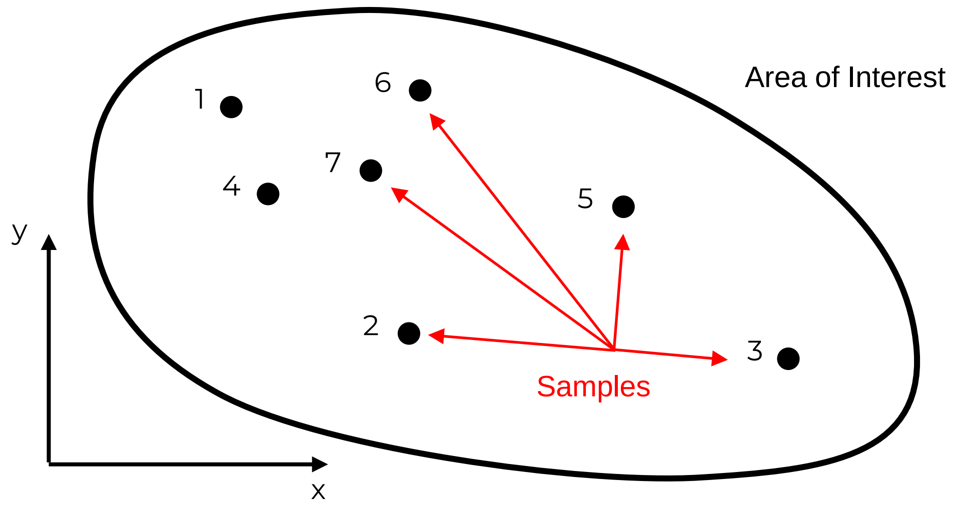
Spatial Uncertainty Modeling with GeostatsPy

Lecture outline . . .

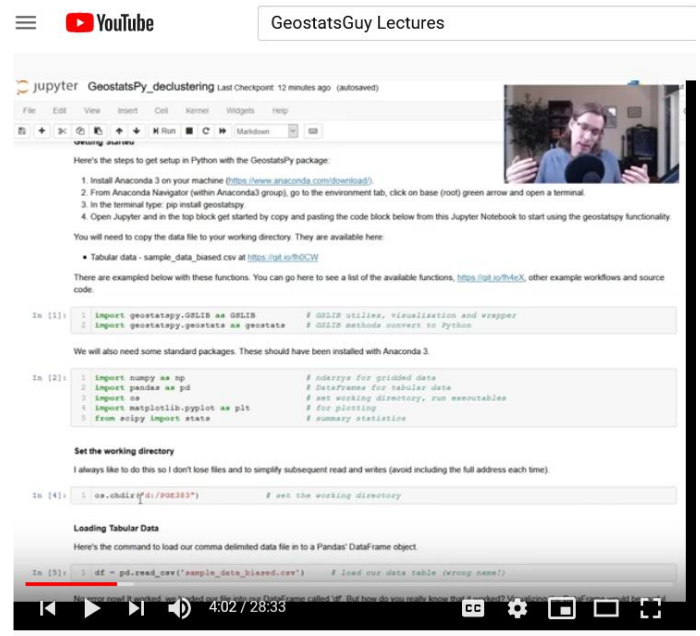
- **Spatial Data Declustering**
- **Interactive Demo with GeostatsPy**
- **Workflow with GeostatsPy**

Motivation

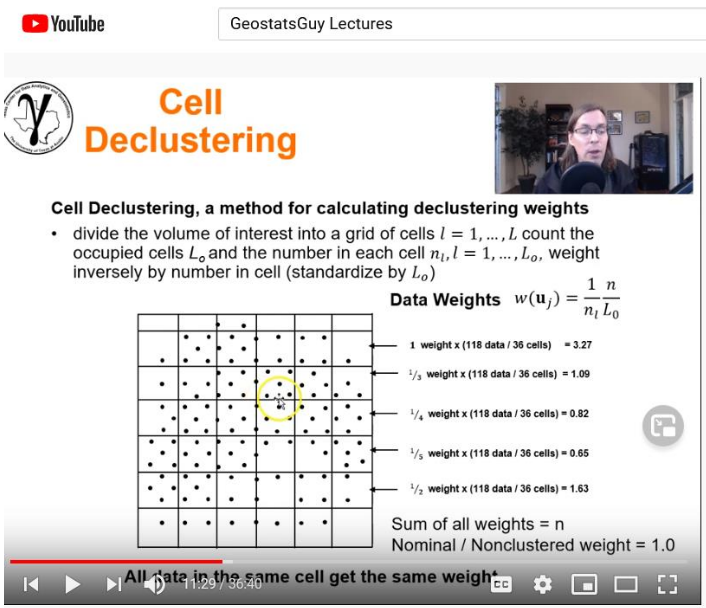
Biased, naïve statistics from biased spatial data samples result in a biased uncertainty model.



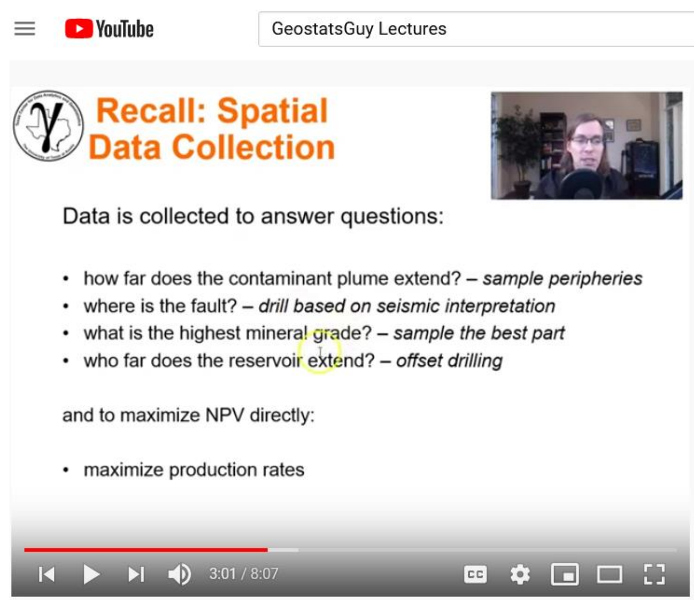
Recorded Lectures



9dPython Data Analytics Reboot: Spatial Declustering



9d Data Analytics Reboot: Spatial Declustering



9c Data Analytics Reboot: Spatial Bias

Lecture outline . . .

- **Spatial Data Declustering**

Spatial Data Collection

Subsurface data is collected to answer questions:

- how far does the contaminant plume extend? – *sample peripheries*
- where is the fault? – *drill based on seismic interpretation*
- what is the highest mineral grade? – *sample the best part*
- who far does the reservoir extend? – *offset drilling*

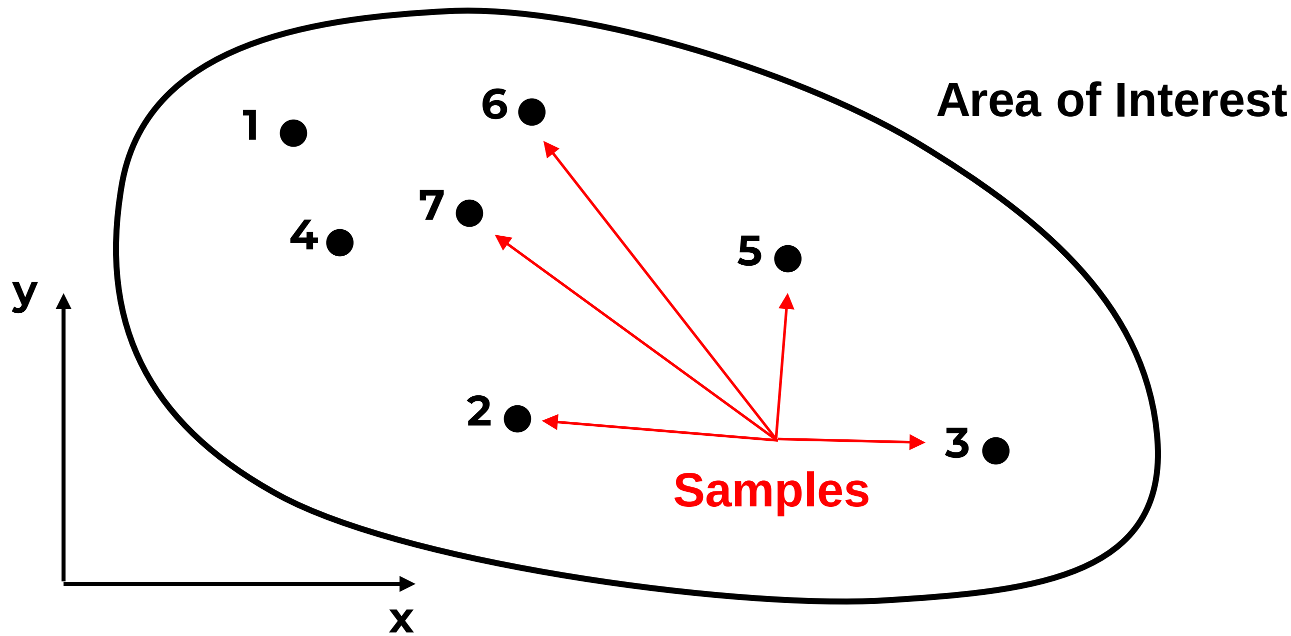
and to maximize value directly:

- maximize production rates
- maximize recovery of a resource
- high grade early value for shorter project pay off period

Clustered Sample

Let's make an estimate for an Area / Volume of Interest:

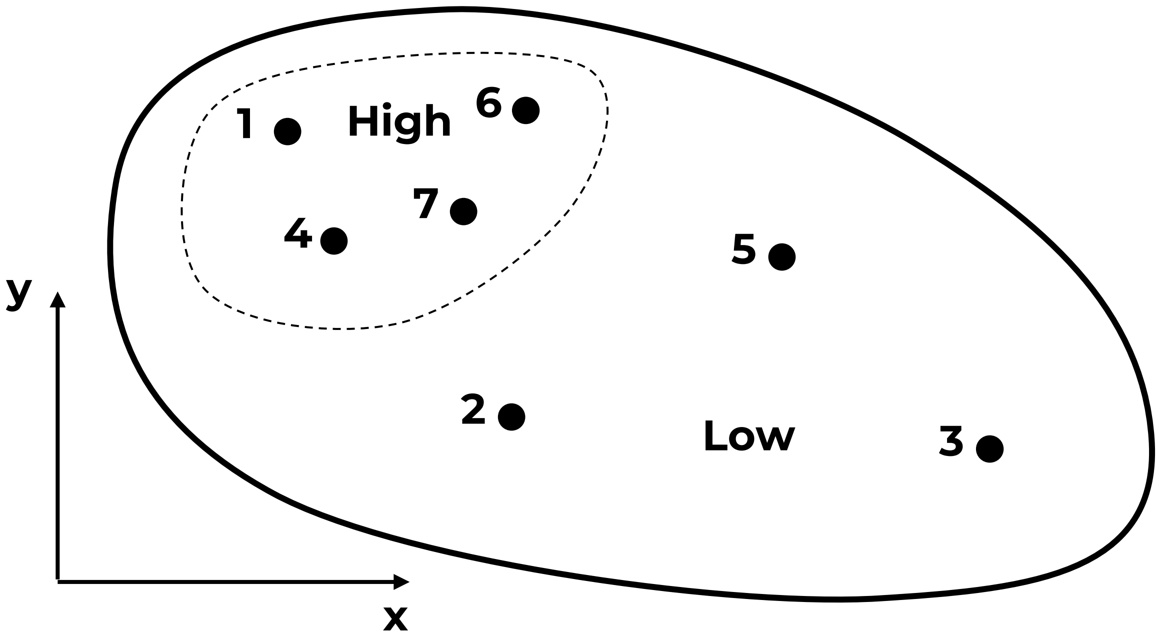
- Inference of the population from a sample.



- To assess the average porosity to calculate OIP

Clustered Sample

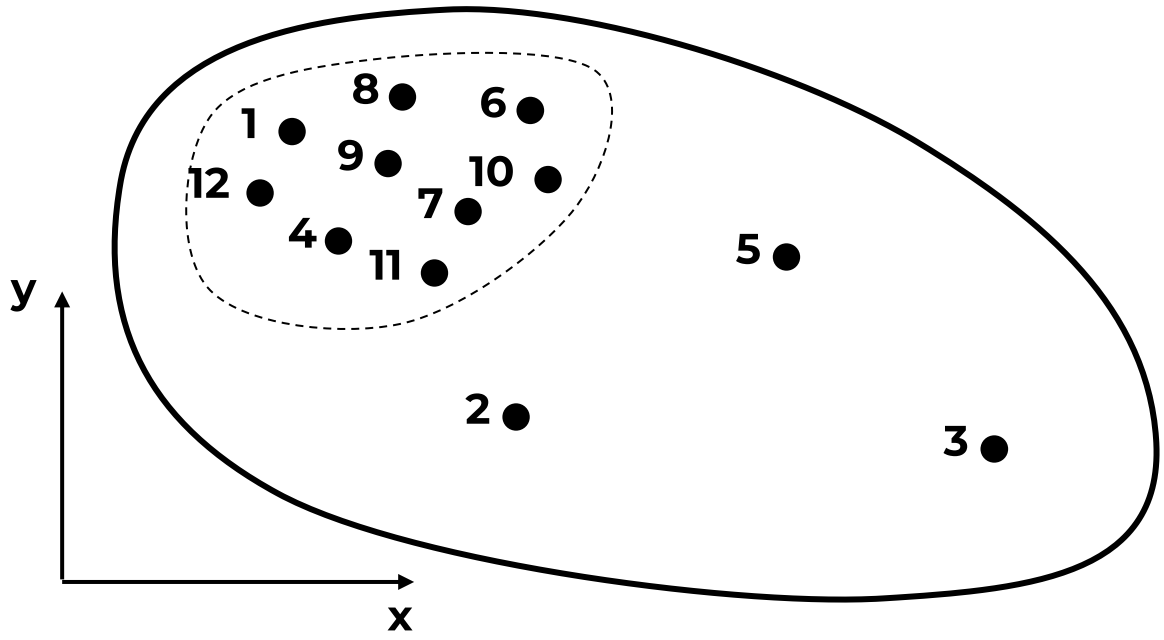
Let's make an estimate for an Area / Volume of Interest:



- What if we knew from seismic that the reservoir quality is better in the top left area?

Clustered Sample

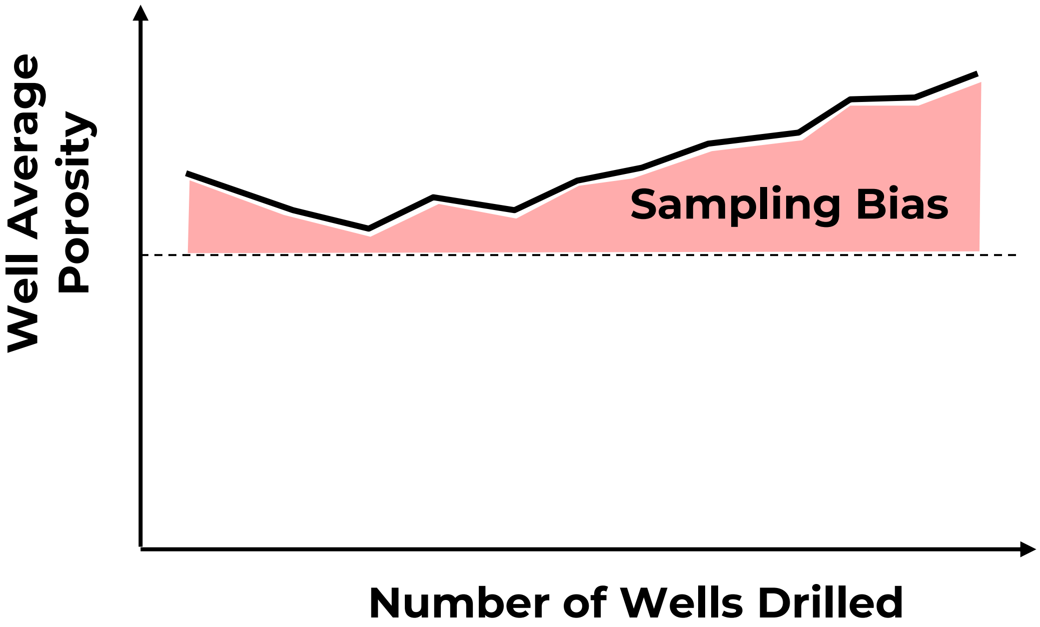
Let’s make an estimate for an Area / Volume of Interest:



- What if we kept drilling in the high value region of the area of interest?

Clustered Sample

How would our estimate of average porosity change as we drilled more wells?:

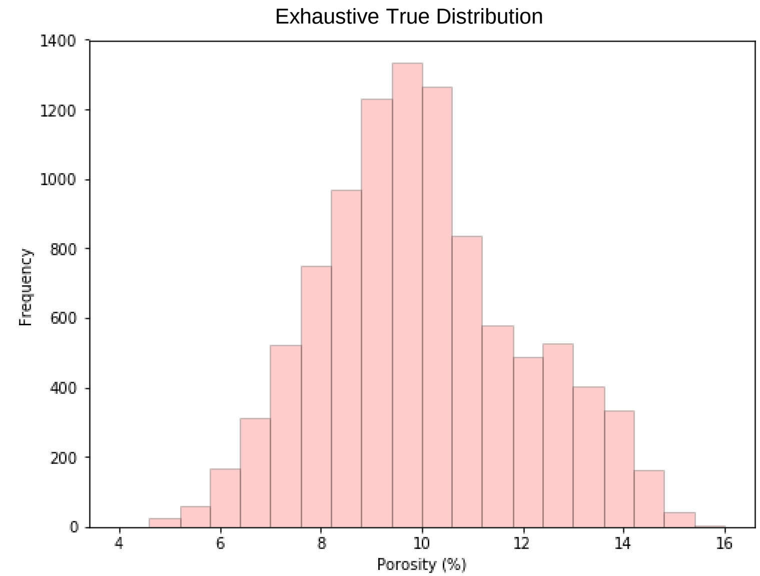


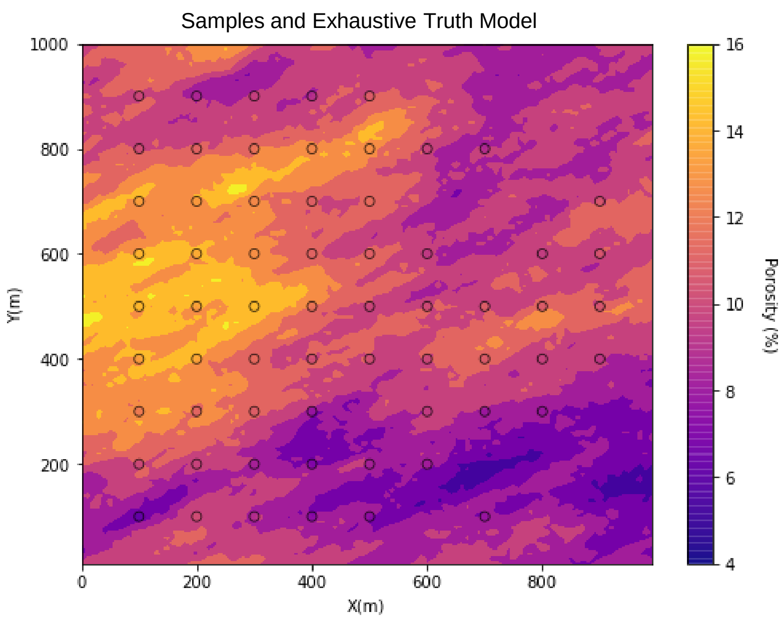
- The naïve sample average becomes more biased!
- We need a method to correct for clustered samples.

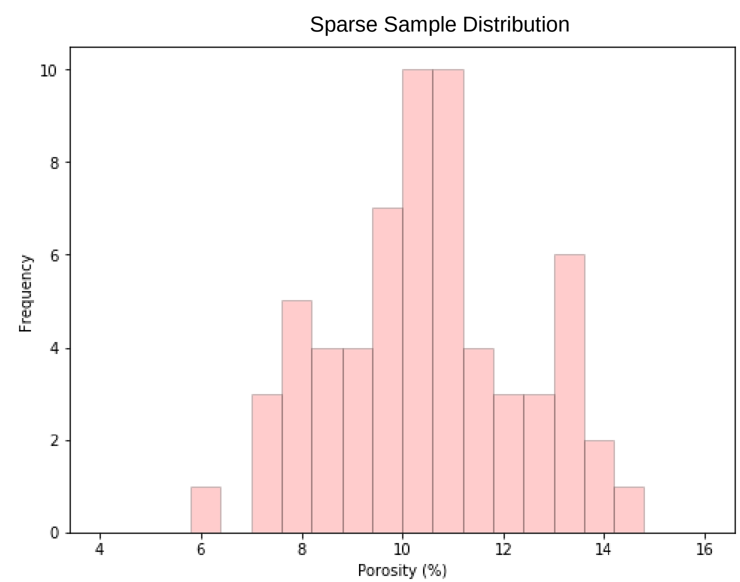
Some Clustered Data

Here's data and x-ray vision:

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





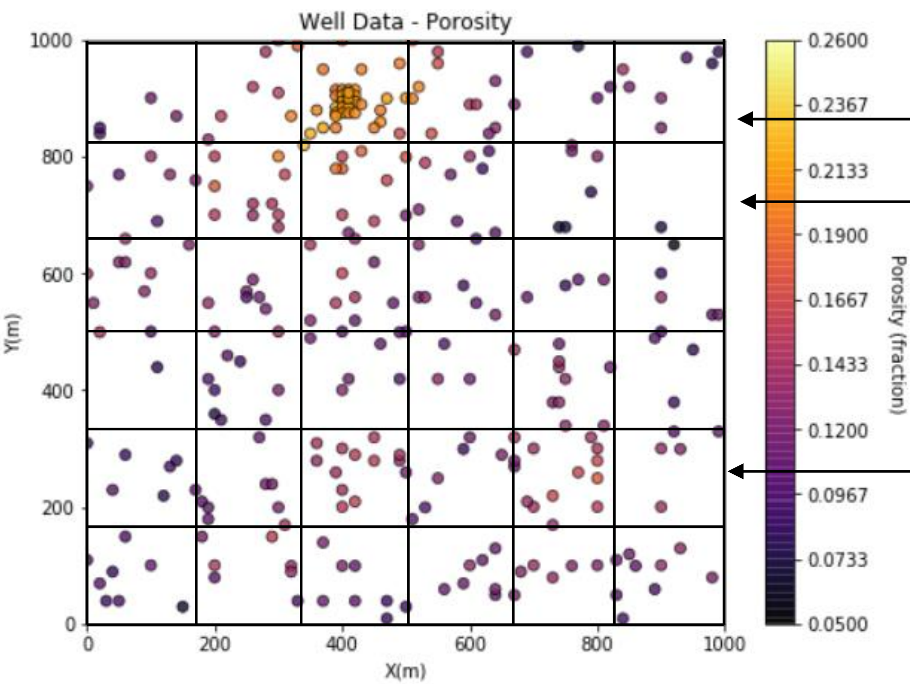


Truth Mean = 10.0 , Clustered Sample Mean = 10.48 , Error = 4.8 %

Cell Declustering

Cell Declustering, a method for calculating declustering weights

- 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)



Data Weights $$w(\mathbf{u}_j) = \frac{1}{n_l}\frac{n}{L_0}$$

1/7 weight x (289 data / 36 cells) = 3.27

1 weight x (289 data / 36 cells) = 1.09

$\frac{1}{4}$ weight x (289 data / 36 cells) = 1.63

Sum of all weights = n
Nominal / nonclustered weight = 1.0

All data in the same cell get the same weight.

Michael Pyrcz, The University of Texas at Austin

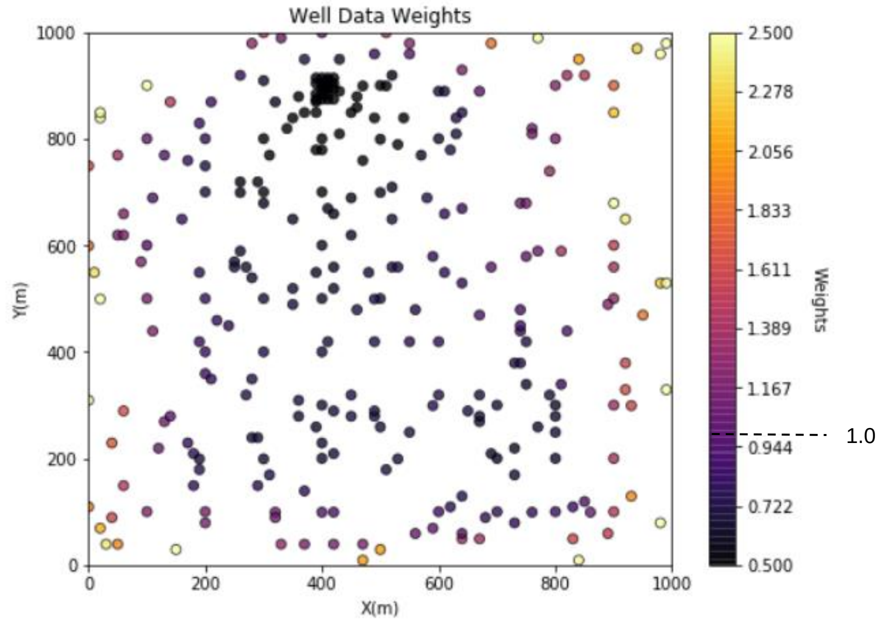
Declustering Weights

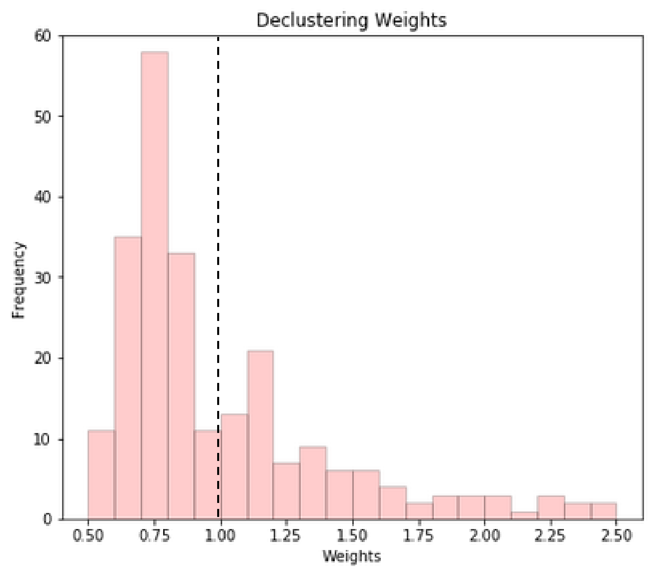
- Declustering weights
    1. 1.0 nominal weight
    2. < 1.0 reduced weight
    3. > 1.0 increased weight

- Note: some software programs assume:

$$\sum_{i}^{n} w(\mathbf{u_i}) = 1$$

then 'nominal weight' is $\frac{1}{n}$





Declustered Distribution

- Updated distribution with declustering weights

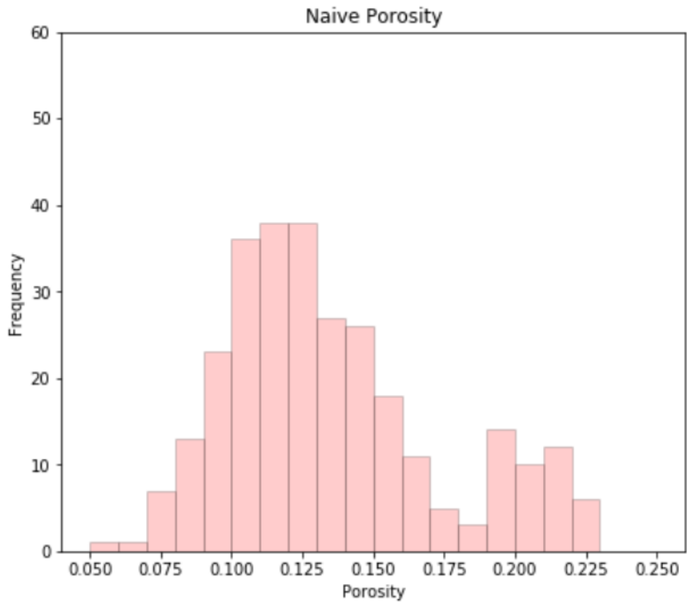
- Now data file / table include values and paired weights based on spatial arrangement.

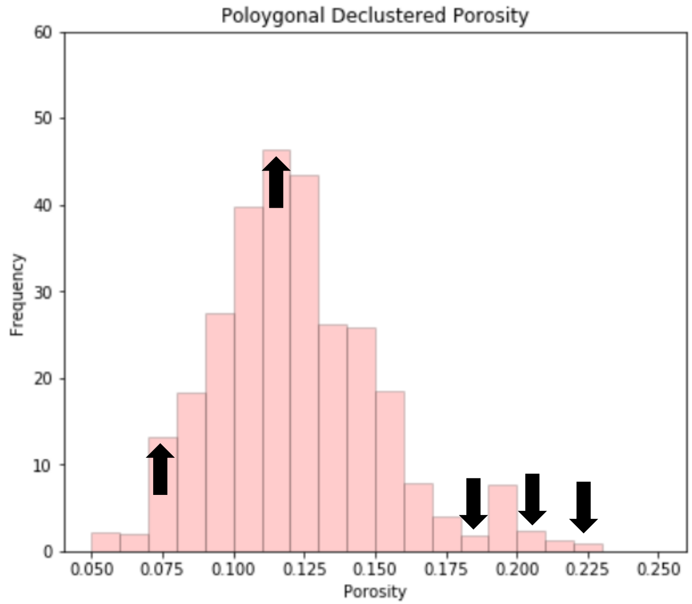
- Possible to calculate any weighted statistic.

  – For example, declustered mean:

$$\bar{z} = \frac{\sum_i^n w(\mathbf{u}_i) z(\mathbf{u}_i)}{\sum_i^n w(\mathbf{u}_i) = n}$$

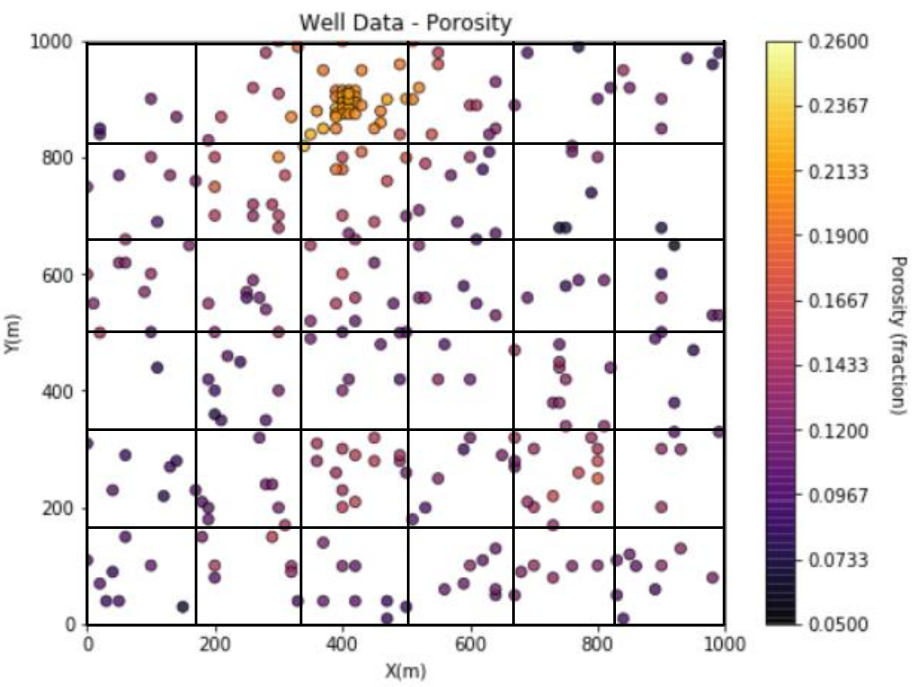
- Python MatPlotLib hist allows for a vector of weights.

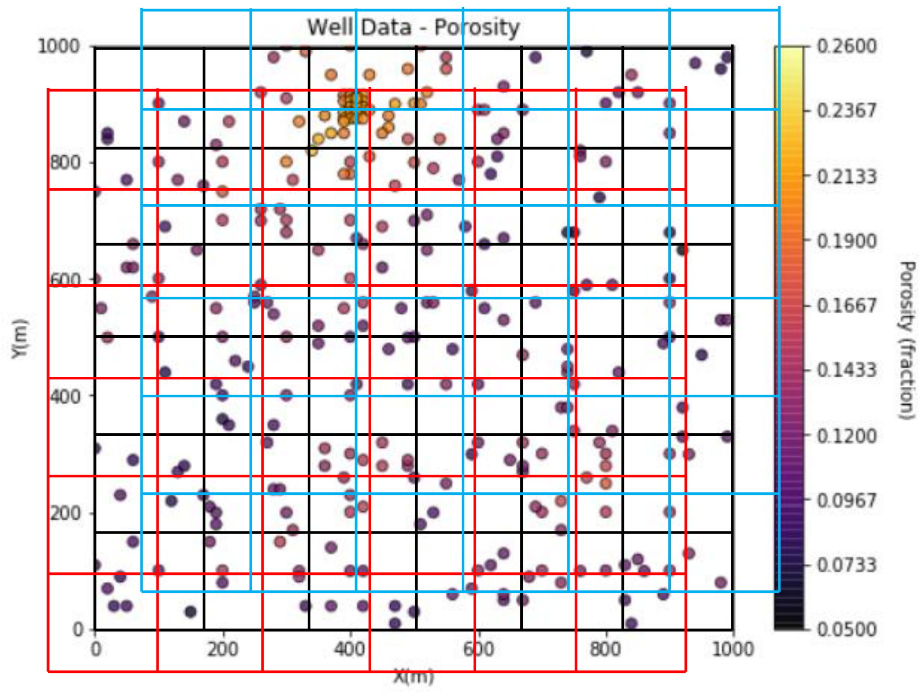




Cell-based Declustering Offsets

- The result is sensitive to exact location of the cell mesh

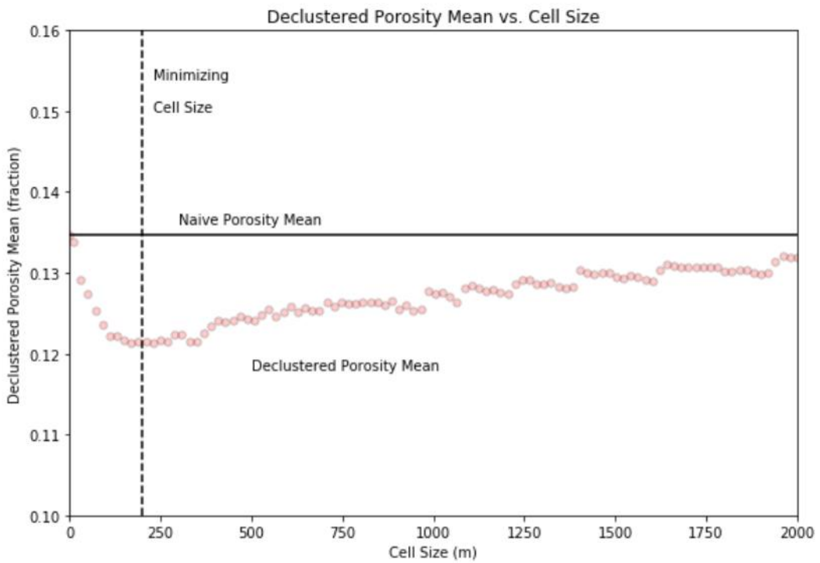




- This sensitivity is removed by iterativing the mesh position, calculating the weights for each and averaging the result.

Cell Size Selection

- 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 by visual inspection and some sensitivity studies
- Could choose the cell size so that there is **approximately one datum per cell in the sparsely sampled areas**, the nominal spacing

Open Source Spatial Data Analytics in Python with GeostatsPy II

Spatial Uncertainty Modeling with GeostatsPy

Lecture outline . . .

- **Interactive Demo with GeostatsPy**
- **Explore the impact of cell size and cell offsets**

Interactive Spatial Data Declustering Demonstration

Michael Pyrcz, Associate Professor, University of Texas at Austin

Twitter | *GitHub* | *Website* | *GoogleScholar* | *Book* | *YouTube* | *LinkedIn*

The Interactive Workflow

Here's a simple workflow for calculating and evaluating spatial data declustering weights smapling bias due to clustered sampled mitigation.

Basic Univariate Summary Statistics and Data Distribution Representativity Plotting in Python with GeostatsPy

Here's a simple workflow with some basic univariate statistics and distribution representativity. This should help you get started data declustering to address spatial sampling bias.

Geostatistical Sampling Representativity

In general, we should assume that all spatial data that we work with is biased.

Source of Spatial Sampling Bias

Data is collected to answer questions:

Interactive_Declustering.ipynb

Spatial Uncertainty Modeling with GeostatsPy

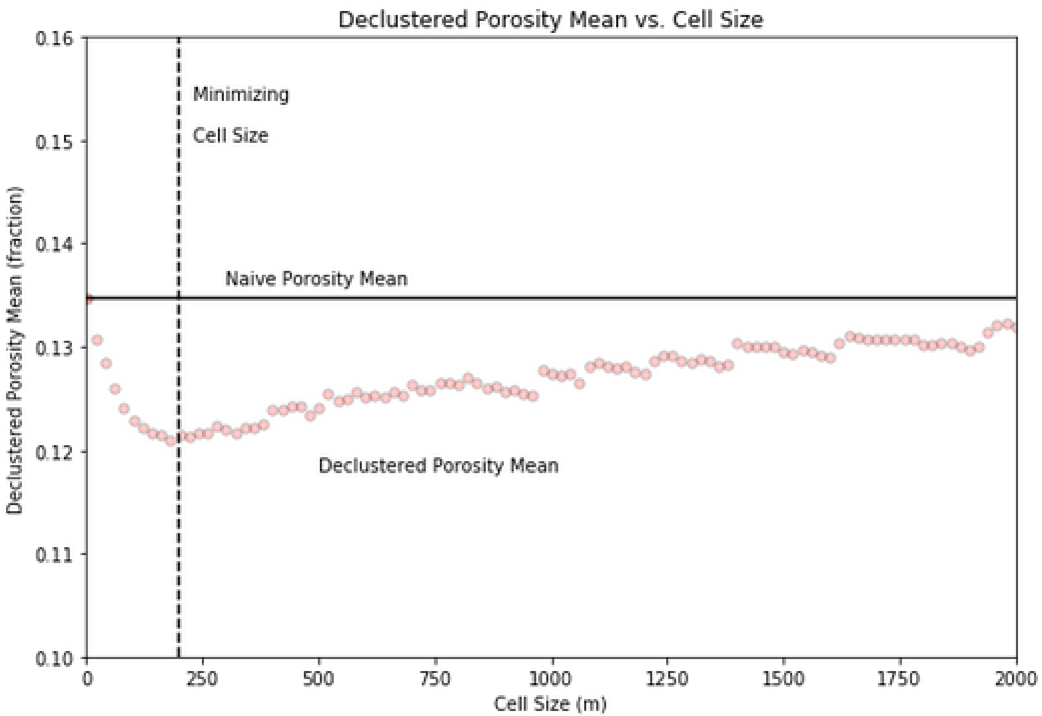
Lecture outline . . .

- Workflow with GeostatsPy

Spatial Simulation Workflow with GeostatsPy

Let's walkthrough a more thorough a spatial data declustering workflow:

- calculate data weights
- visualize and QC the results



Python Jupyter variogram calculation (GeostatsPy_declustering.ipynb).

Spatial Uncertainty Modeling with GeostatsPy

Lecture outline . . .

- **Spatial Simulation**
- **Interactive Demo with GeostatsPy**
- **Workflow with GeostatsPy**