Multivariate Modeling
Geostatistical Subsurface Modeling

Lecture outline . . .

- **Who am I?**
- **Motivation / Goals**
- **Class Description / Objectives**
- **The Plan**
- **Resources**

Introduction

Prerequisites

Feature Selection

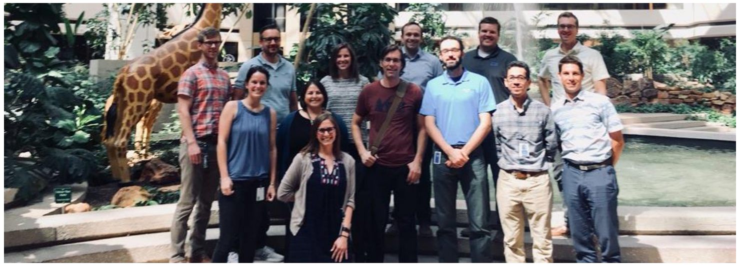
Multivariate Modeling

Conclusions

Who Am I?

Spring 2018 Class of Introduction to Geostatistics

Oil and Gas University, Florence, Italy



Anadarko, Midland, TX

Michael Pyrcz

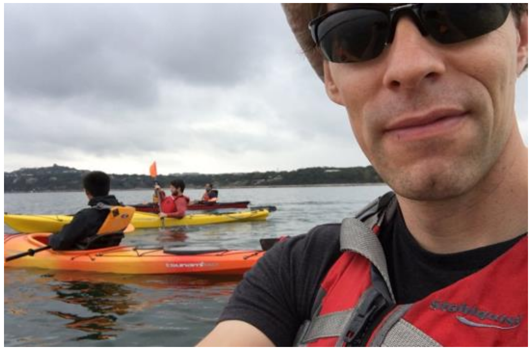
1. Pyrcz: is pronounced “perch”

2. I’m New: new to UT PGE, started August, 2017. Everything is new!

3. I have practical experience: over 17 years of experience in consulting, teaching and industrial R&D in statistical modeling, reservoir modeling and uncertainty characterization.

Who Am I?

Fall 2018 Class of Introduction to Geostatistics



Fall 2017 PGE 383

Michael Pyrcz

4. Flexible: got ideas, feedback to improve the learning opportunities. Let's work together to reach our learning objective.

5. Available: I have an open door policy. Drop by my office. Drop a line anytime.

6. An Engineer, but: My B.Sc. was Mining Engineering, my M.Sc. started as Geotechnical Engineering (then skipped to Ph.D.) and my Ph.D. was in Quantitative Geology. I spent 13 years in Earth Science R&D working with geological and geophysical reservoir modeling. I speak geo.

Who Am I?

AAPG SEPM Panel Discussion on Modeling

CPGE Webinar on Big Data

Michael Pyrcz

8. Active in Outreach, Social Media and Professional Organizations

• associate editor with Computers and Geosciences, editorial board of Mathematical Geosciences for the International Association of Mathematical Geosciences

• program chair for SPE Data Analytics Technical Section

• associate editor with Computers and Geosciences

• author of the textbook “Geostatistical Reservoir Modeling”

• board member for Mathematical Geosciences

• GeostatsGuy on Twitter, GitHub, GeostatsGuy Lectures on YouTube

I’m committed to supporting / partnering for development opportunities of working professionals

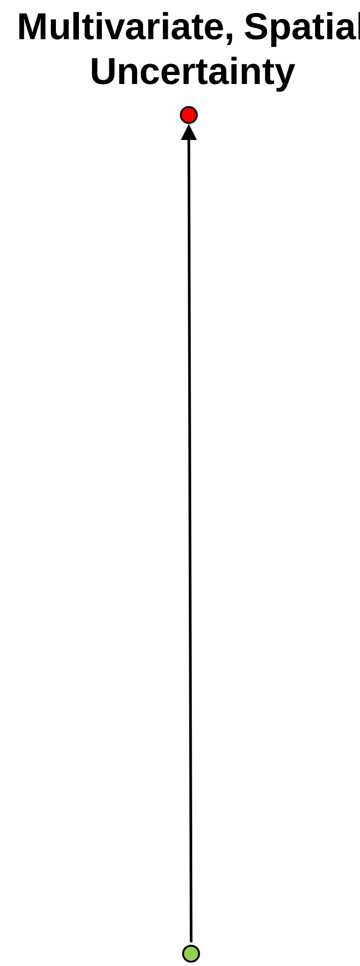
What Will You Learn?

The Goal – multivariate, spatial estimation and uncertainty

Based on previous discussions with **Daniel Pinkston and Eric Radjef** we identified, prioritized this topic

Learn an uncertainty approach for dealing with multivariate spatial uncertainty

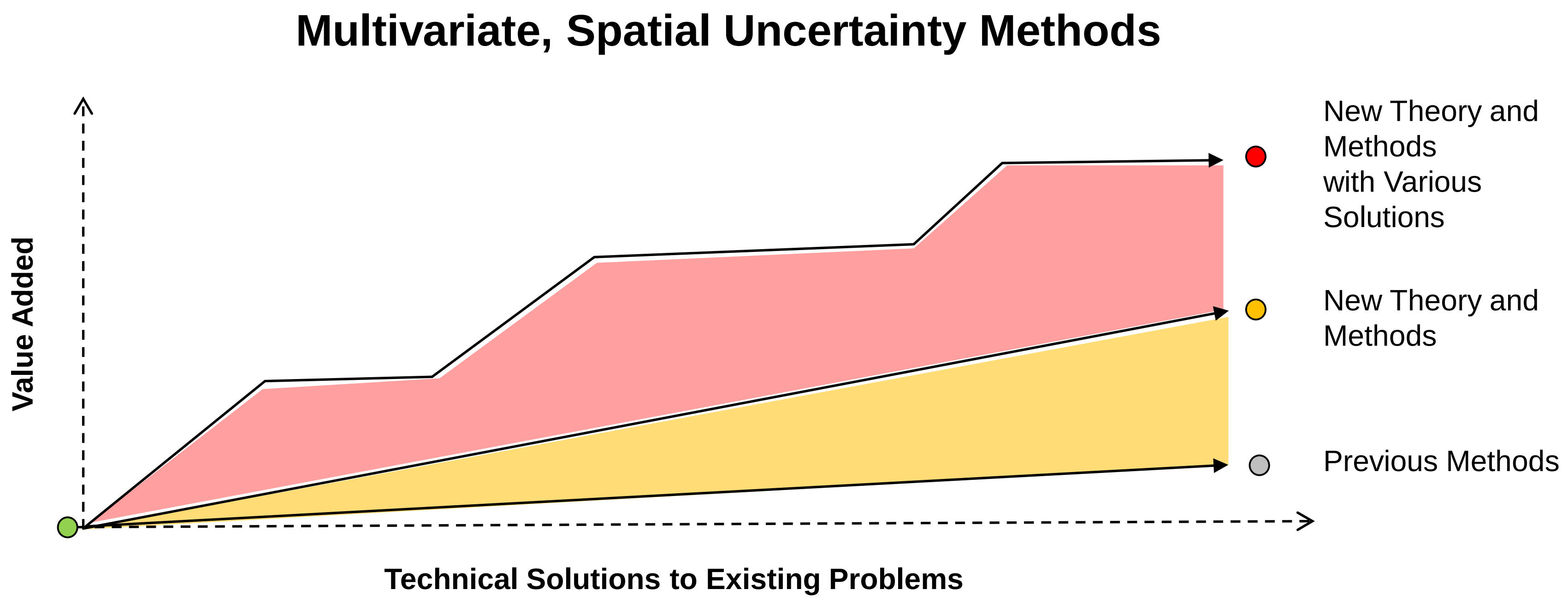
- More details on the workflow from:
  - Pyrcz and Deutsch (2014) Section 4.1.4 Multivariate Mapping
  - Originally published in an SPE paper by Deutsch, Ren and Leuangthong (2005)



What Will You Learn?

Today is an investment in learning

- Build operational capability
- Provide incremental value.



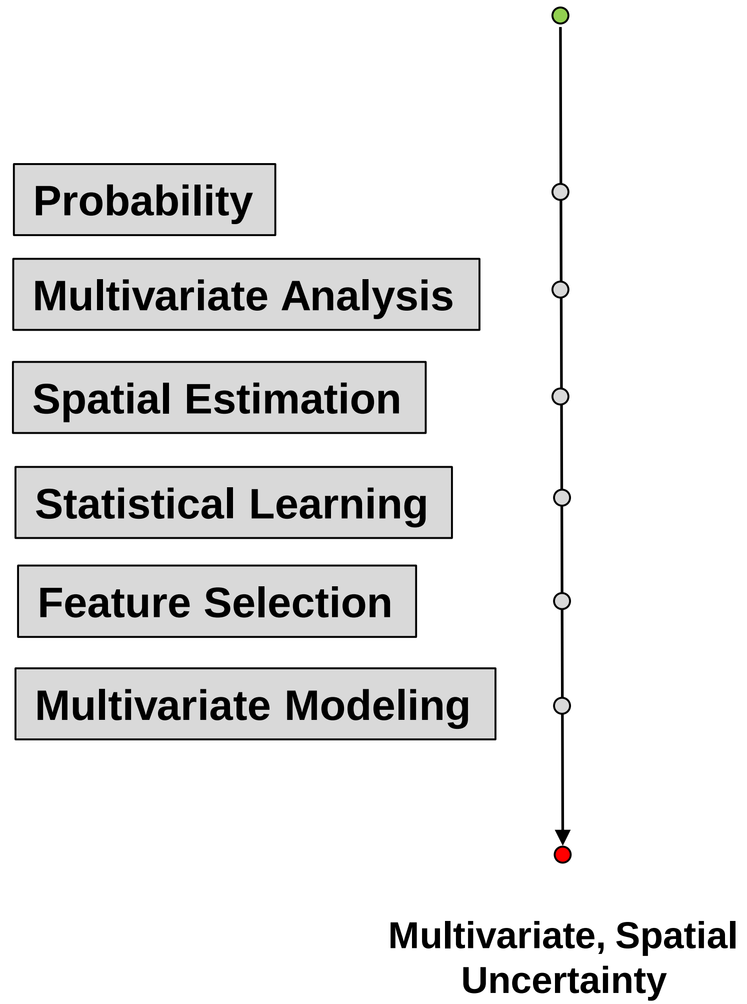
What Will You Learn?

Reaching our Goal – we need to cover some material to get there!

Today we will:

- Cover the building blocks
- Put together some workflow components

Of course, full workflow development would require time to investigate the problem and available data.



What Will You Learn?

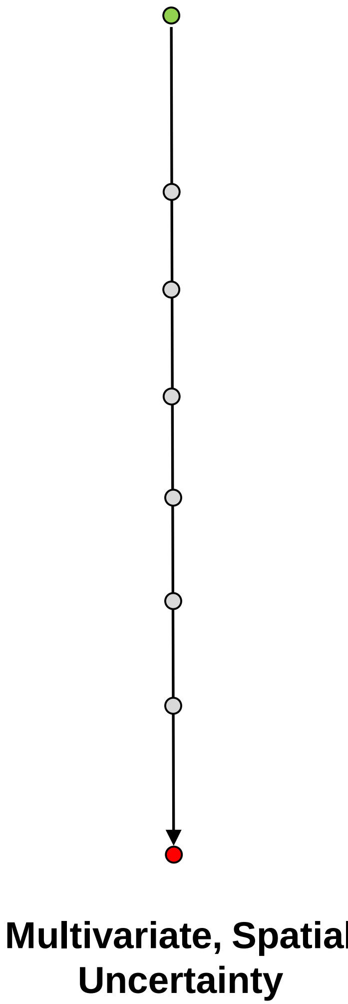
The Agenda for Today

Addressing the basic building blocks to reach our goal by the end of the day.

Are we flexible on timing for each section?

One Day Course - Spatial, Multivariate Uncertainty

Topic	Time
Introduction: objectives, plan	8:00 - 8:30
Probability – Frequentist and Bayesian concepts	8:30 - 10:00
Break, Discussions	10:00 - 10:15
Multivariate Analysis – correlation, marginal, conditional and joint	10:15 - 11:00
Spatial Continuity – stationarity, variogram calculatio	11:00 - 12:00
Lunch	12:00 - 1:00
Spatial Estimation – trend modeling, kriging estimation and uncertainty	1:00 - 2:00
Statistical Learning – training and testing, model bias and variance	2:00 - 2:30
Feature Selection – curse of dimensionality, over fit and ranking	2:30 - 3:30
Break, Discussions	3:30 - 3:45
Multivariate Spatial Workflows – cosimulation and updating methods	3:45 - 4:45
Conclusion – summary, lessons learned, discussion	4:45 -



What Will You Learn?

But there will be deliverables as we go along!

Multivariate Modeling

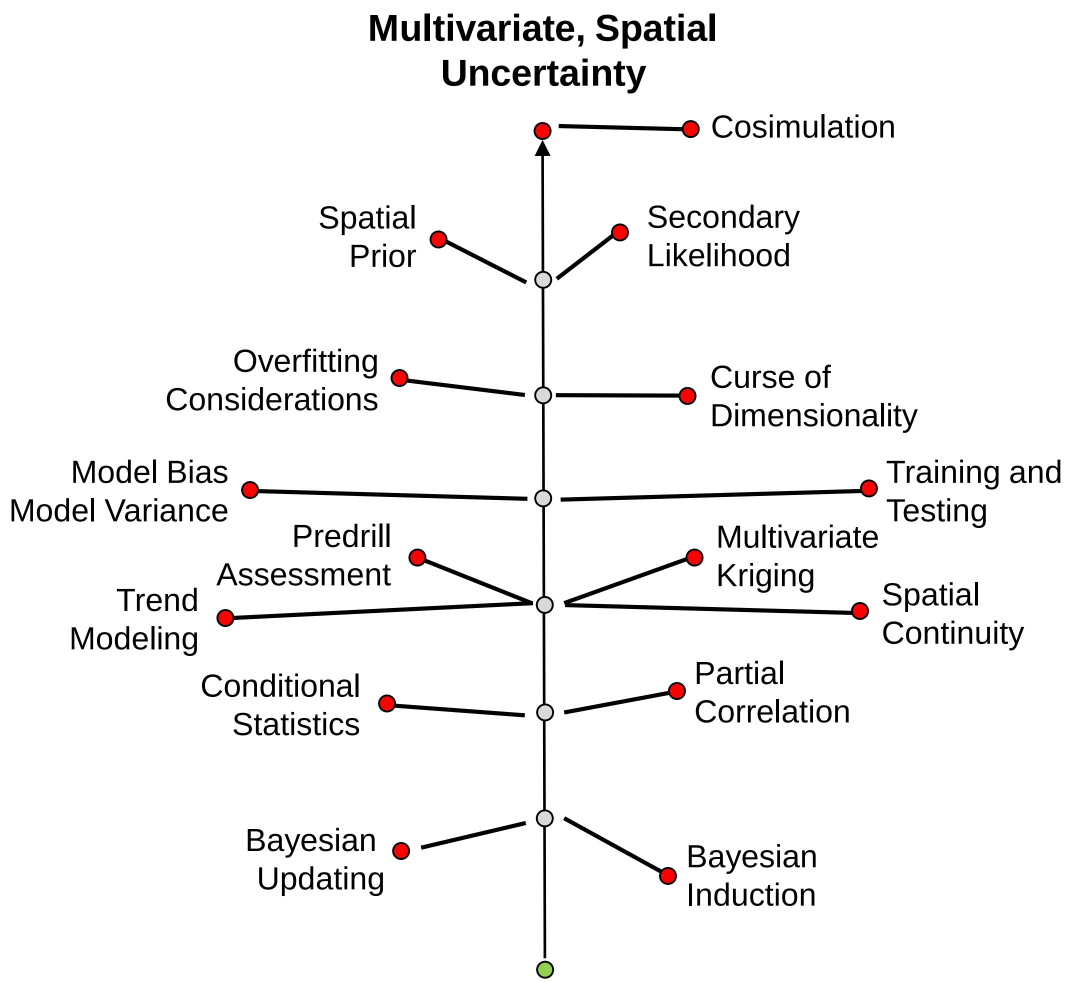
Feature Selection

Statistical Learning

Spatial Estimation

Multivariate Analysis

Probability



What Will You Learn?

There is Much More! – the building blocks can be reimplemented and expanded to address various other problems, opportunities.

There is much more that we can cover.

Probability

Multivariate Analysis

Spatial Estimation

Statistical Learning

Feature Selection

Multivariate Modeling

- **Statistical Inference**
- **Representative Statistics**
- **Debiasing**
- **Uncertainty Sources**
- **Trend Modeling**
- **Model Optimization**
- **Discrete Uncertainty**
- **Facies Models**
- **Object-based Modeling**
- **Support Vector Machines**
- **Fair Spatial Model Testing**
- **Stochastic Simulations**
- **Value of Information**

Multivariate, Spatial Uncertainty

How Will You Learn All of That?

Here’s the Plan:

1. **Interactive** lectures / discussion to cover the theory

2. Live **demonstrations**

3. Simple, well documented **experiential learning** in **Excel** and **Python**

We will **adjust for success**:

- Let me know if you are lost, stuck, something is not working or you aren’t learning!
- e.g. we could switch from experiential to live demo
- e.g. we could use less Python and more Excel

Feedback welcome as we proceed.

Why Excel and Python?

Excel (without Visual Basic Applications):

- Everyone has it
- Most know the basics, many are really good at it
- It is very easy to interrogate, click on any cell, see the equations!
- You can actually build complicated methods and workflows, up from simple operations

'If you can't explain it simply, you don't understand it well enough!'

\- Albert Einstein

Python:

- Is very powerful, the most resources and assistance
- Packages allow us to put together workflows with limited old-fashioned 'coding'
- Leverage the world's brilliance

'Certainly there's a phenomenon around open source. You know free software will be a vibrant area. 'There will be a lot of nest things that get done there.'

\- Bill Gates

'20 years with C++ and FORTRAN, but with Python I code less, but get more done.'

\- Michael Pyrcz

Jupyter Notebooks?

Python with Jupyter Notebooks

- Workflows that integrate blocks of code, documentation, results

GeostatsPy: Monte Carlo Simulation for Subsurface Data Analytics in Python

Michael Pyrcz, Associate Professor, University of Texas at Austin

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

PGE 383 Exercise: Monte Carlo Simulation for Subsurface Data Analytics in Python

Here's a simple workflow, demonstration of Monte Carlo simulation for subsurface uncertainty modeling workflows. This should help you get started with building subsurface models that integrate uncertainty sources.

Monte Carlo Simulation

Definition: random sampling from a distribution

Procedure:

1. Model the representative distribution (CDF)
2. Draw a random value from a uniform [0,1] distribution (p-value)
3. Apply the inverse of the CDF to calculate the associated realization

In practice, Monte Carlo simulation refers to the workflow with multiple realizations drawn to buld an uncertainty model.

$$X^\ell = F_x(p^\ell), \ \forall \ \ell = 1, \dots, L$$

where X^ℓ is the realization of the variable X drawn from its CDF, F_x, with cumulative probability, p-value, p^ℓ.

It would be trivial to apply Monte Carlo simulation to a single variable, after many realizations one would get back the original distribution. The general approach is to:

1. Model all distributions for the input, variables of interest $F_{x_1}, \dots, F_{x_m}$.
2. For each realization draw $p_1^\ell, \dots, p_m^\ell$, p-values
3. Apply the inverse of each distribution to calculate a realization of each variable, $X_j^\ell = F_{x_j^\ell}^{-1}(p_j^\ell), \ \forall \ j = 1, \cdots, m$ variables.
4. Apply each set of variables for a ℓ realization to the transfer function to calculate the ouptput realization, $Y^\ell = F(X_1^\ell, \dots, X_m^\ell)$.

Monte Carlo Simulation (MCS) is extremely powerful

- Possible to easily simulate uncertainty models for complicated systems
- Simulations are conducted by drawing values at random from specified uncertainty distributions for each variable
- A single realization of each variable, $X_1^\ell, X_2^\ell, \dots, X_m^\ell$ is applied to the transfer function to calculate the realization of the variable of interest (output, decision criteria):

$$Y^\ell = F(X_1^\ell, \dots, X_m^\ell), \ \forall \ \ell = 1, \dots, L$$

- The MCS method builds empirical uncertainty models by random sampling

Let's take a simple example, OIP is oil-in-place calculated as the product of reservoir volume, V, average porosity, $\overline{\phi}$, and oil saturaton, $\overline{s_o}$:

$$OIP^\ell = V^\ell \cdot \overline{\phi}^\ell \cdot \overline{s_o}^\ell, \ \forall \ \ell = 1, \dots, L$$

It would be difficult to directly calculate the OIP distribution as a combination of all these different distributions.

- The distributions could all have different forms (parametric or non-parametric)
- We use MCS to empirically work this out by sampling
- Repeat to calculate enough realizations for analysis.

Let's set the minimum and maximum values for plotting.

```
apor_min = 0.1; apor_max = 0.2          # average porosity min and max
vol_min = 0.0; vol_max = 4000000        # vol min and max
```

In the *NumPy* package we have handy methods for Monte Carlo simulation from parametric distributions. We can actually draw all L realizations at once for each variable and store them in ndarrays (each ndarray with realizations $\ell = 1, \dots, L$).

```
apor = np.random.normal(apor_mean, apor_stdev, size=L) # average porosity MCS simulation L times and store
in array
vol = np.random.lognormal(vol_mu, vol_sigma, size=L)   # volume ...
so = np.random.uniform(so_min, so_max, size=L)         # saturation oil
```

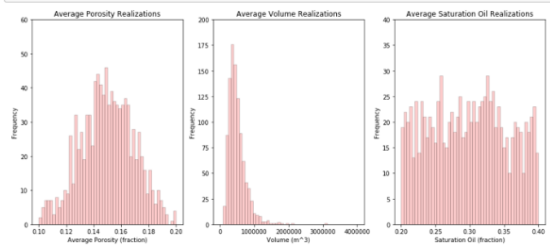
Let's plot the distributions of the realizations of each variable to make sure the match the form of the parametric distributions that we selected.

```
plt.subplot(131)
GSLIB.hist_st(apor,apor_min,apor_max,log=False,cumul=False,bins=50,weights=None,xlabel="Average Porosity
(fraction)",title="Average Porosity Realizations")
plt.ylim(0.0,60)

plt.subplot(132)
GSLIB.hist_st(vol,vol_min,vol_max,log=False,cumul=False,bins=50,weights=None,xlabel="Volume (m^3)",title
="Average Volume Realizations")
plt.ylim(0.0,200)

plt.subplot(133)
GSLIB.hist_st(so,so_min,so_max,log=False,cumul=False,bins=50,weights=None,xlabel="Saturation Oil (fractio
n)",title="Average Saturation Oil Realizations")
plt.ylim(0.0,40)

plt.subplots_adjust(left=0.0, bottom=0.0, right=2.0, top=1.2, wspace=0.2, hspace=0.2)
plt.show()
```

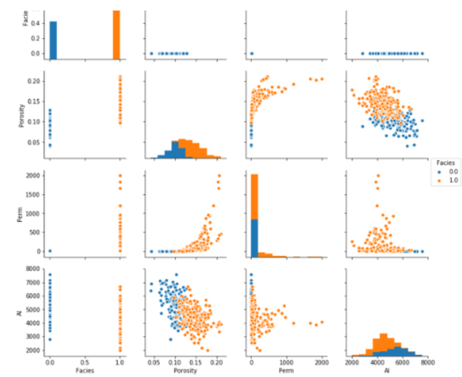


This looks good, the shapes are Gaussian, lognormal and uniform and the central tendency and dispersion make sense given the parameters that we selected.

Now we can use broadcast methods to calculate the output realizations of OIP, based on this equation.

$$OIP^\ell = V^\ell \cdot \overline{\phi}^\ell \cdot \overline{s_o}^\ell \cdot 6.29 \quad \forall \ \ell = 1, \dots, L$$

where 6.28 $bbls/m^3$.

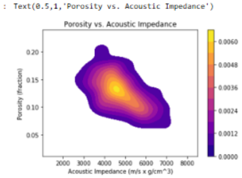


Joint, Conditional and Marginals

We can use kernel density estimation to estimate the joint probabilities density function (pdf) for the paired data, a 2D pdf! We could use this to estimate any required joint, marginal and conditional probability (care must be taken with normalization). Let's use the seaborn package 'kdeplot' function to estimate the joint pdf for porosity and acoustic impedance.

```
ax = sns.kdeplot(df['AI'].values,df['Porosity'].values, shade=True, n_levels = 10,cmap=cmap,cbar= True, sh
ade_lowest = False)
ax.set_xlabel('Acoustic Impedance (m/s x g/cm^3)'); ax.set_ylabel('Porosity (fraction)'); ax.set_title('Po
rosity vs. Acoustic Impedance')
```

```
Text(0.5,1,'Porosity vs. Acoustic Impedance')
```



I think is it useful to visualize the joint pdfs with the marginal pdfs on a single plot. We can use seaborn's 'jointplot' to accomplish this.

```
ax = sns.jointplot('AI','Porosity', df, kind='kde',shade = False, n_levels = 10,cmap=cmap, shade_lowest =
True);
```

- Work with a variety of kernels (Python, R, C, javascript, etc.)
- Make professional workflows with Markdown docs
- Use containers and run online (e.g. Docker)

GeostatsPy?

GeostatsPy

- Set of Functions in Python
  - GeostatsPy is a set of Python functions for most of the required workflow steps
  - Much is reimplemented in Python.
  - Package written by myself, we will tailor, augment to support training.
  - I welcome feedback.
  - Open Source anyone can use it
  - Free for any use
  - Download it from PyPi with:

'pip install geostatspy'

Project description

GeostatsPy Package

The GeostatsPy Package brings GSLIB: Geostatistical Library (Deutsch and Journel, 1998) functions to Python. GSLIB is extrememly robust and practical code for building spatial modeling workflows. I specifically wanted it in Python to support my students in my **Data Analytics**, **Geostatistics** and **Machine Learning** courses. I find my students benefit from hands-on opportunities, infact it is hard to imagine teaching these topics without providing the opportunity to handle the numerical methods and build workflows.

This package includes 2 parts:

1. geostatspy.gslib includes low tech wrappers of GSLIB functionality (note: some functions require access to GSLIB executables)
2. geostatspy.geostats includes GSLIB functions rewritten in Python.

Package Inventory

Here's a list and some details on each of the functions available.

geostatspy.gslib Functions

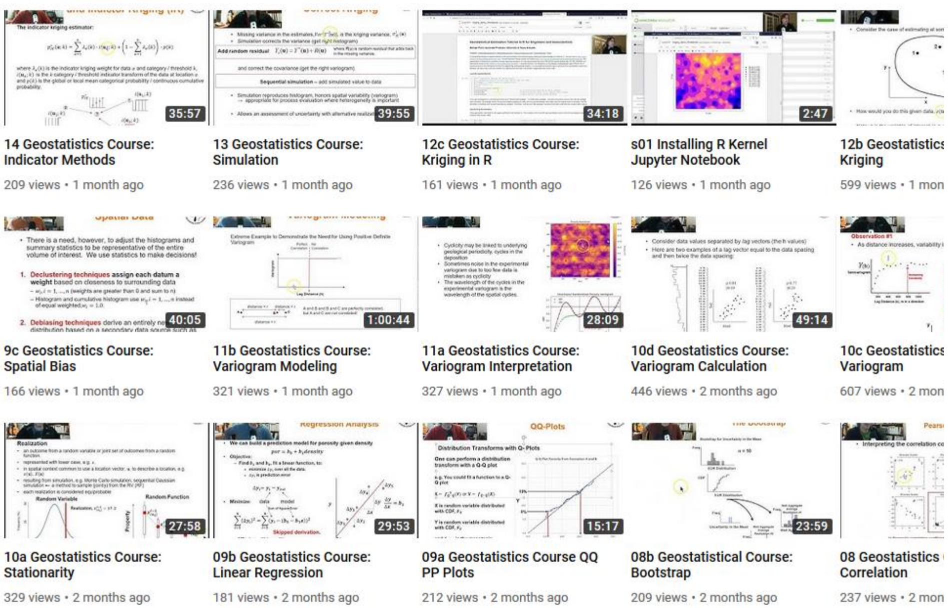
Utilities to support moving between Python DataFrames and ndarrays, and Data Tables, Gridded Data and Models in Geo-EAS file format (standard to GSLIB):

1. **ndarray2GSLIB** - utility to convert 1D or 2D numpy ndarray to a GSLIB Geo-EAS file for use with GSLIB methods
2. **GSLIB2ndarray** - utility to convert GSLIB Geo-EAS files to a 1D or 2D numpy ndarray for use with Python methods

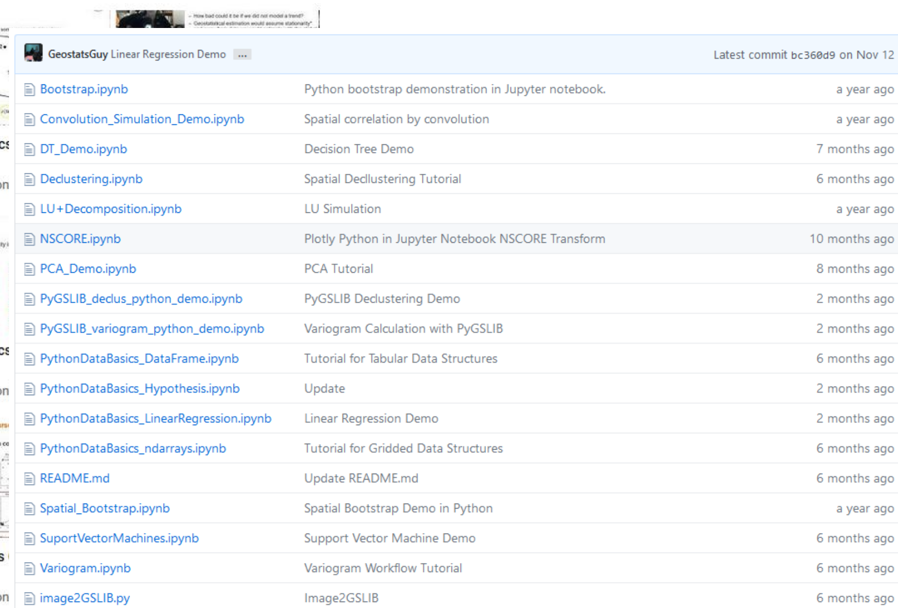
Really, How Will I Do That?

Resources During and After the Workshop

1. I can provide covered lecture material in .pdf format.
2. All lectures, demos and workflows from the undergraduate class are available to you (YouTube and GitHub).
3. I'm always happy to discuss.



GeostatsGuy Lectures Channel on YouTube



GeostatsGuy Repositories on GitHub

More Resources: Books

1. **GSLIB: Geostatistical Software Library and User's Guide**, *Deutsch and Journel* – want to get started doing geostatistics now? GSLIB open source guide – start now!
2. **Applied Geostatistics with SGeMS**, *Remy, Boucher and Wu* – user guide for SGeMS open-source. Adds user friendly interface and interactive 3D visualization.
3. **Introduction to Applied Geostatistics**, *Isaaks and Srivastava* – very 'read-able' mix of theory and practice, but stops early with estimation.
4. **Geostatistics: Modeling Spatial Uncertainty**, *Chiles and Delfiner* – best coverage of modern theory and deep concepts, may not be accessible to all.
5. **Geostatistical Reservoir Modeling**, *Pyrcz and Deutsch* – good coverage of theory and practice, accessible, strong linkages to geology and coverage of modern methods.
6. **Multiple-point Geostatistics: Stochastic Modeling with Training Images**, *Mariethoz and Caers* – best coverage on multiple-point geostatistics, theory and practice.
7. **Petroleum Geostatistics**, *Caers* – concise treatment of petroleum workflows.
8. **Reservoir Model Design**, *Ringrose and Bentley* – good coverage of workflow design and tradecraft with examples.
9. **Statistics for Petroleum Engineers and Geoscientists**, *Jensen, Lake, Corbett, and Goggin* – good coverage of statistical and probabilistic approach for petroleum engineering.
10. Other Theory: **Geostatistics for Natural Resources Evaluation**, **Mining Geostatistics**

Why Learn About Geostatistical Subsurface Modeling?

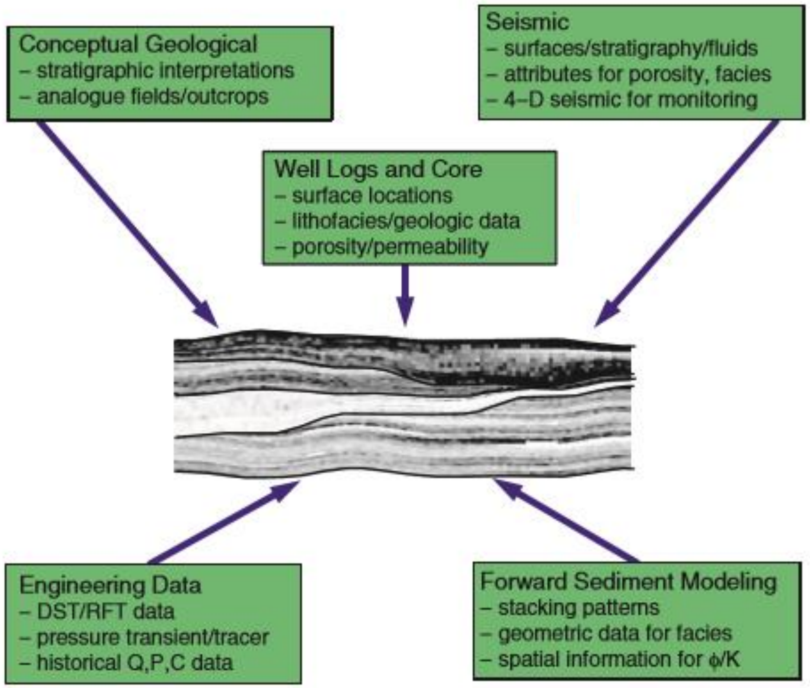
Why should you have a greater proficiency on reservoir modeling?

Level 1: Basic Understanding

Most reservoir asset subsurface teams develop a **stochastic 3D reservoir model**.

• If you work with the subsurface, you will work with stochastic reservoir models!

• **Understand adjacent disciplines** and workflows in you team.



Subsurface asset integration (Pyrcz and Deutsch, 2014).

Why Learn About Geostatistical Subsurface Modeling?

Why should you have a greater proficiency on reservoir modeling?

Level 2: Improved Communication

Reservoir modeling sits in the middle of the subsurface team and integrates all available engineering, geological and geophysical information.

• Improved reservoir modeling capability results in **improved communication and integration** in the subsurface team.

TABLE 2.1. RESERVOIR CONCEPTS AND ASSOCIATED GEOLOGICAL AND GEOSTATISTICAL EXPRESSIONS

Concept	Geological Expression	Geostatistical Expression
Major changes in relationships between reservoir bodies	Architectural complexes and complex sets	Regions—separate units and model with unique methods and input statistics
Changes in reservoir properties within reservoir bodies	Basinward and landward stepping Fining/Coarsening up	Nonstationary mean
Stacking patterns ~~if~~reservoir bodies	Organization, disorganization, compartmentalization, compensation	Attraction, repulsion, minimum and maximum spacing distributions, interaction rules
Major direction of continuity	Paleo-flow direction	Major direction of continuity, locally variable azimuth model

Subsurface concepts, with their geological and geostatitistical expressions (modified from Pyrcz and Deutsch, 2014).

Why Learn About Geostatistical Subsurface Modeling?

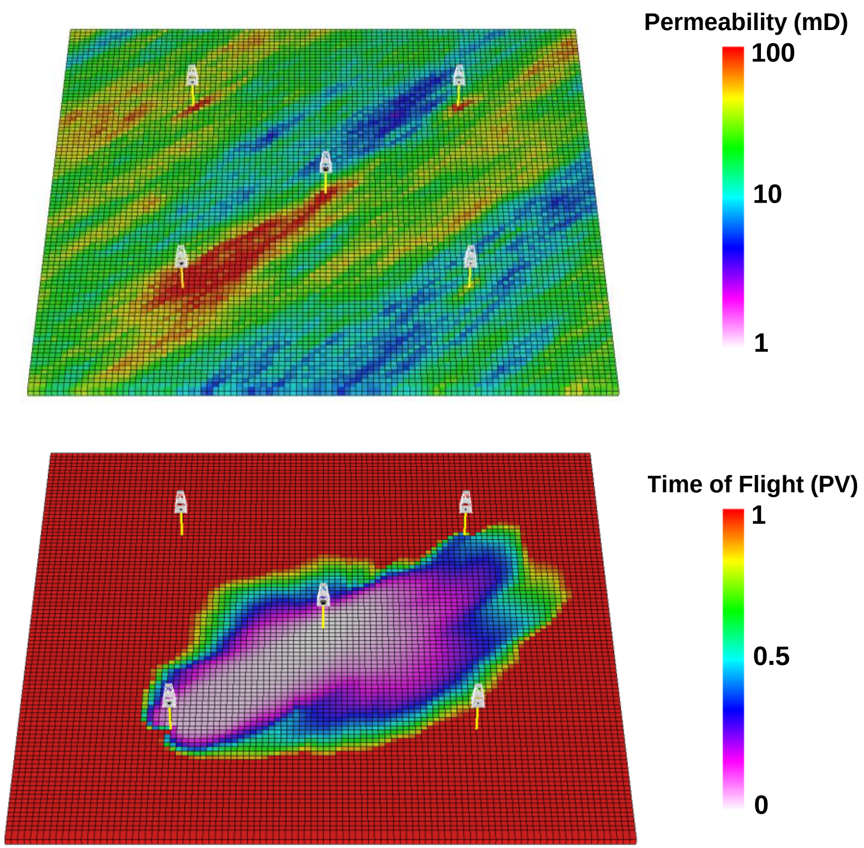
Why should you have a greater proficiency on reservoir modeling?

Level 3: Maximize Your Impact

Reservoir models are directly applied for forecasting that support decision making.

- **Best integration of your knowledge** into the subsurface model.

- If your expertise does NOT impact the model, you may NOT impact the development decision!



Permeabilty heterogeneity and flow response.

Why Learn About Geostatistical Subsurface Modeling?

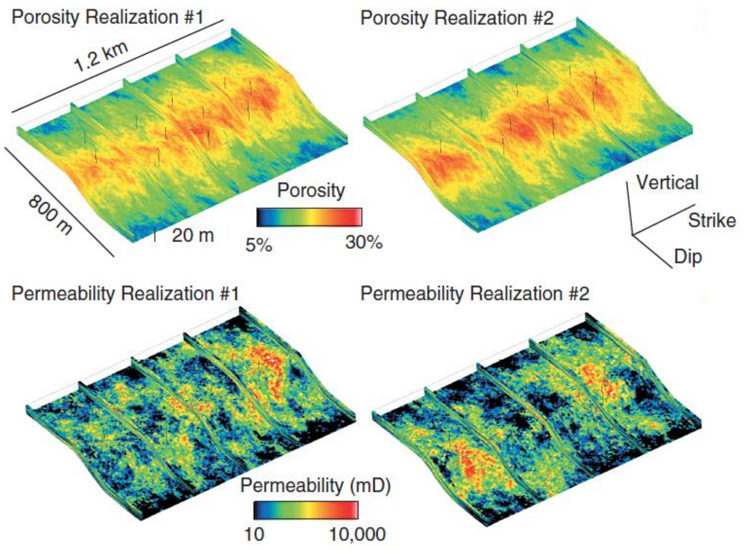
Why should you have a greater proficiency on reservoir modeling?

Level 4: Build Subsurface Models

Most subsurface modelers are geoscientists and engineers that learned on the job.

Become a subsurface modeler!

- Black box, uninformed reservoir modeling will result in bad decisions.
- Advanced knowledge unlocks novel workflows to solve difficult subsurface problems.



Subsurface asset integration.

The Subsurface Modeling Steps

- We could spend more time together! I do a lot of training, e.g. 3 day course:
  - Some Prerequisites
  - Data Preparation
  - Univariate and Multivariate Analysis
  - Spatial Analysis
  - Estimation and Trend Modeling
  - Stochastic Simulation
  - Uncertainty Analysis
  - Model Checking
  - Decision Making
- For each section
  - Lectures and demos
  - Subsurface inference and modeling
  - Completion of project update documentation and presentation

Tailored to geoscientists, engineers, managers etc.

Introduction

Prerequisites

Data Preparation

Univariate Analysis

Multivariate Analysis

Spatial Characterization

Spatial Estimation

Spatial Simulation

Uncertainty Analysis

Model Checking

Decision Making

More on Coding

More on Software / Coding:

- This is not a coding / software workshop.
- I can't teach Python in 1 day.
- We will demonstrate well-documented workflows in Python.
- We will focus on the steps, inputs and outputs.
- Don't be concerned if you don't completely understand the code.

But if coding to solve subsurface problems is your interest:

- I teach that also!
- I have a lot of basic tutorials and example workflows to solve practical subsurface problems
- Will conduct a Data Science Bootcamp starting this summer with Professor John Foster from the Institute for Computational Engineering and Science (ICES) at the University of Texas at Austin.

Aside: Benefit of Coding

Reasons All Geoscientists and Engineers Should Learn to Code

Transparency – *no compiler accepts hand waiving!* Coding forces your logic to be uncovered for any other scientist or engineer to review.

Reproducibility – *run it, get an answer, hand it over, run it, get the same answer.* This is a main principle of the scientific method.

Quantification – *programs need numbers.* Feed the program and discover new ways to look at the world.

Open-source – *leverage a world of brilliance.* Check out packages, snippets and be amazed with what great minds have freely shared.

Break Down Barriers – *don't throw it over the fence*. Sit at the table with the developers and share more of your subject matter expertise for a better product.

Deployment – *share it with others and multiply the impact*. Performance metrics or altruism, your good work benefits many others.

Efficiency – *minimize the boring parts of the job*. Build a suite of scripts for automation of common tasks and spend more time doing science and engineering!

Always Time to Do it Again! – *how many times did you only do it once?* It probably takes 2-4 times as long to script and automate a workflow. Usually worth it.

Be Like Us – *it will change you*. Users feel limited, programmers truly harness the power of their applications and hardware.

Aside: Benefit of Coding

Reasons All Geoscientists and Engineers Should Learn to Code
Some Caveats

1. Any type of coding, scripting, workflow automation matched to your working environment is great. We don't all need to be C++ experts.

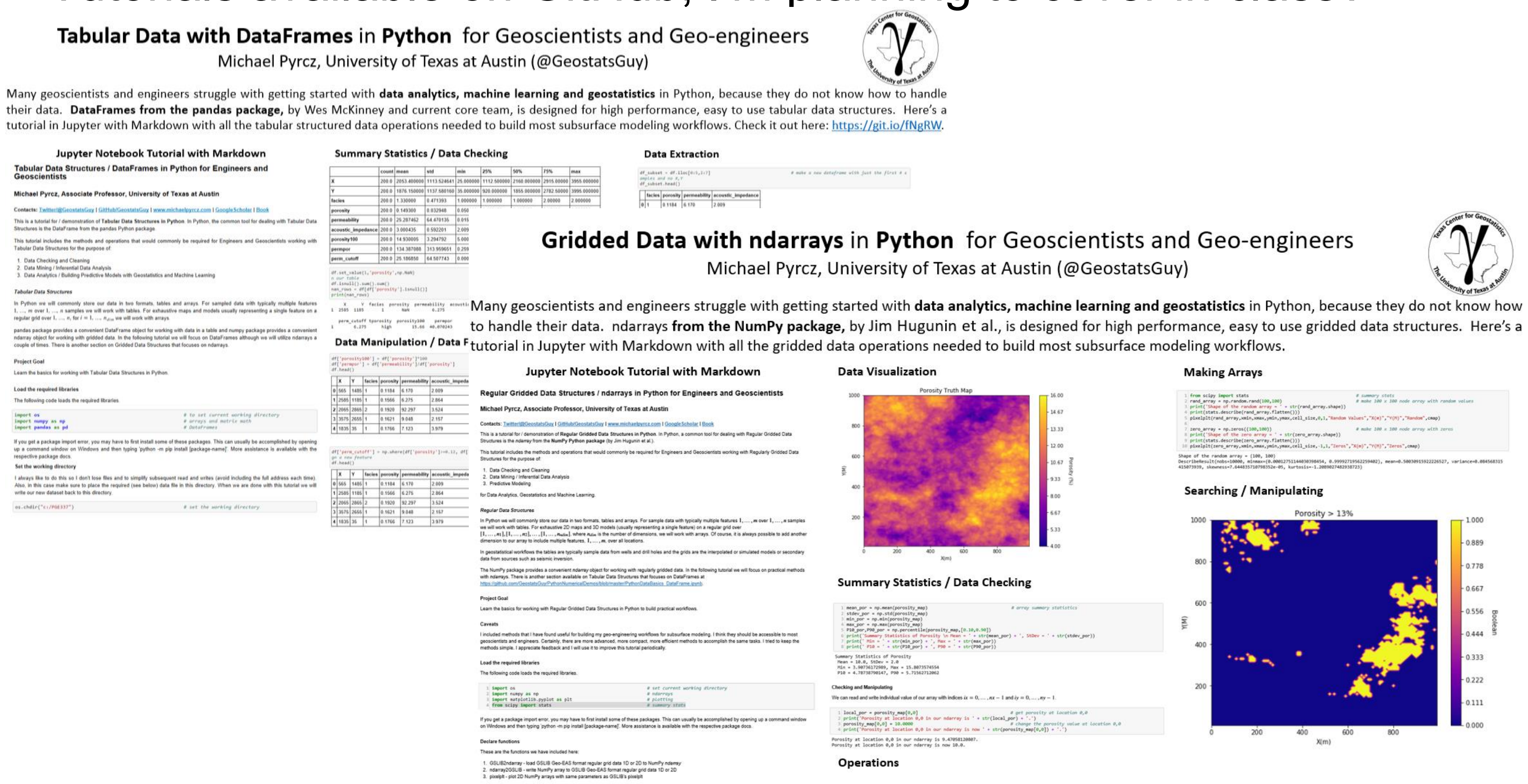
2. I respect the experience component of geoscience and engineering expertise. This is beyond coding and is essential to workflow logic development, best use of data etc.

3. Some expert judgement will remain subjective and not completely reproducible. I'm not advocating for the geoscientist or engineer being replaced by a computer.

Example Python Coding Demonstrations / Tutorials

Python Basics

- Numpy for arrays (ndarrays) for gridded data
- Pandas for tabulated data (DataFrames) for well data
- Tutorials available on GitHub, I'm planning to cover in class?



Multivariate Modeling
Geostatistical Subsurface Modeling

Lecture outline . . .

- Who am I?
- Motivation / Goals
- Class Description / Objectives
- The plan
- Resources

Introduction

Prerequisites

Feature Selection

Multivariate Modeling

Conclusions