# PGE 338 Data Analytics and Geostatistics

## Lecture 5: Univariate Distributions

**Lecture outline . . .**

- **Parametric Distributions**
- **Nonparametric Distributions**
- **Monte Carlo Simulation**
- **Bootstrap**
- **Distribution Transforms**

**Michael Pyrcz, The University of Texas at Austin**


- Introduction
- General Concepts
- Univariate
  - PDF / CDF
  - Statistics
  - Distributions
  - Heterogeneity
  - Hypothesis
- Bivariate
- Time Series Analysis
- Spatial Analysis
- Machine Learning
- Uncertainty Analysis

# Motivation

Now we can use our PDF and CDF's to:

- model with parametric and nonparametric methods
- bootstrap and Monte Carlo Simulation to quantify uncertainty
- transforms to new distributions required in many data analytics workflows

# PGE 338 Data Analytics and Geostatistics

## Lecture 5: Univariate Distributions

**Lecture outline . . .**

- **Parametric Distributions**

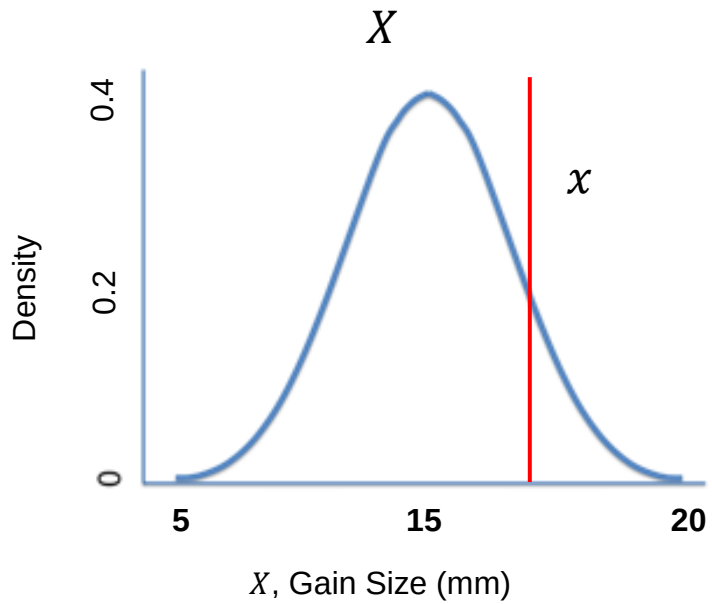
**Michael Pyrcz, The University of Texas at Austin**


- Introduction
- General Concepts
- Univariate
  - PDF / CDF
  - Statistics
  - Distributions
  - Heterogeneity
  - Hypothesis
- Bivariate
- Time Series Analysis
- Spatial Analysis
- Machine Learning
- Uncertainty Analysis

# Recall: Random Variable (RV) Definition

**Recall: Random Variable**

- we do not know the value at a location / time, **it can take on a range of possible values**, fully described with a PDF.
- represented as an **upper-case variable, e.g., $X$, while possible outcomes or data measures are represented with lower case, e.g., $x$.**
- in spatial context common to use a location vector, $\mathbf{u}$, to describe a location, e.g., $x(\mathbf{u})$, $X(\mathbf{u})$




Random variable, grain size uncertain at a location represented with a PDF.

# Where Have We Seen Random Variables?

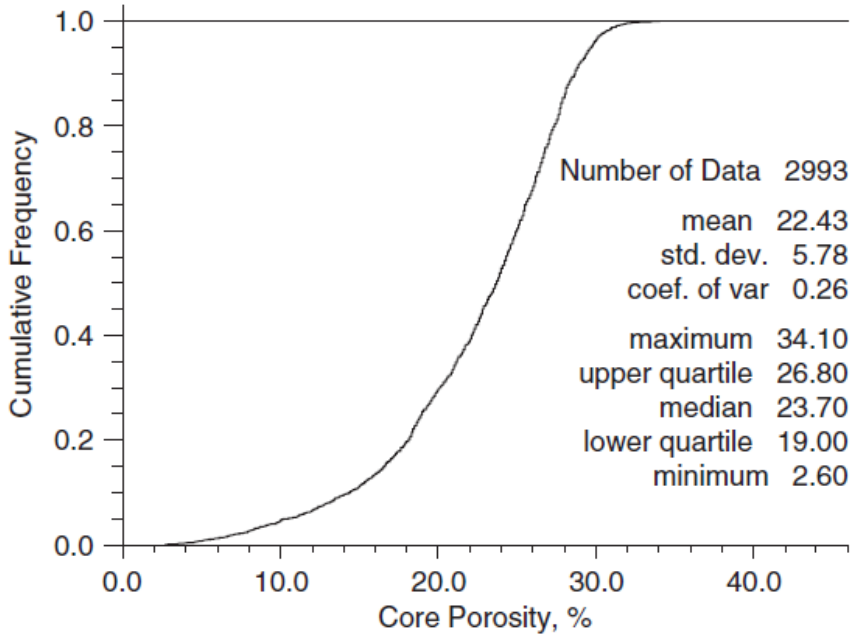
**Our random variables are represented by distributions:**

Cumulative Distribution Function:

$$F_x(x) = P(X \leq x) = \int_{-\infty}^{x} f(u)du$$

**Random Variable**

**A specific value or outcome.**




Porosity CDF.

Statistical Expectation Operators:

For example, expectation of addition of two random variables:

**Random Variables**

$$E[X + Y] = E[X] + E[Y]$$

Image from Pyrcz and Deutsch (2014), Geostatistical Reservoir Modeling.

# Parametric Distributions

**A Variety of Parametric Distributions are Available to:**

- Provide a complete PDF / CDF function with very few parameters to infer
  - Fit distribution to the data, then predict any required probabilities!
  - Extrapolate poorly sampled distribution tails.

- Inferential tool, if distribution is known by theory, e.g., central limit theorem

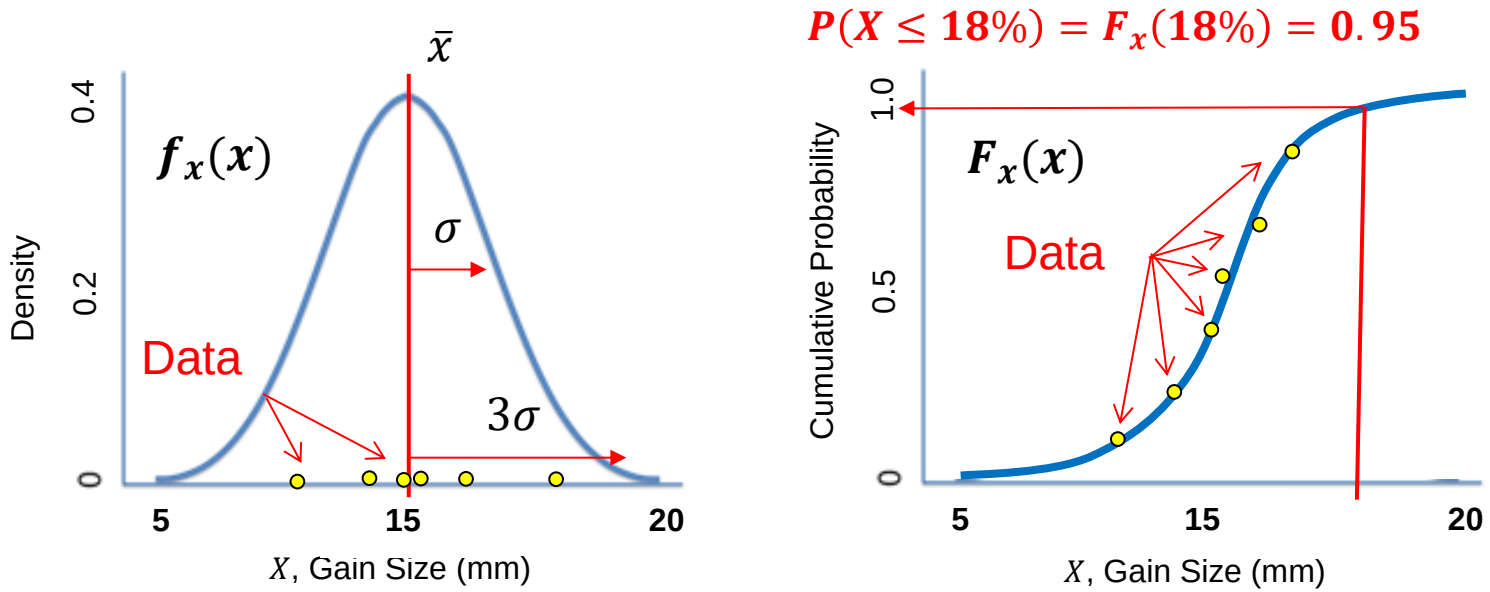



Illustration of use of parametric distribution model to predict an extreme value.

# Parametric Distributions

**Parametric Models May Be Relate to an Underlying Theory**

- Gaussian / Normal Distribution:
  - the normal distribution is the limit distribution for the central limit theorem
- Chi-squared Distribution:
  - for square of Gaussian distributed random variables
- Weibull (w-ā-bull) distribution:
  - in reliability theory, $f_x(x)$ probability of failure over time, $x$

Damage from a steam train boiler explosion.

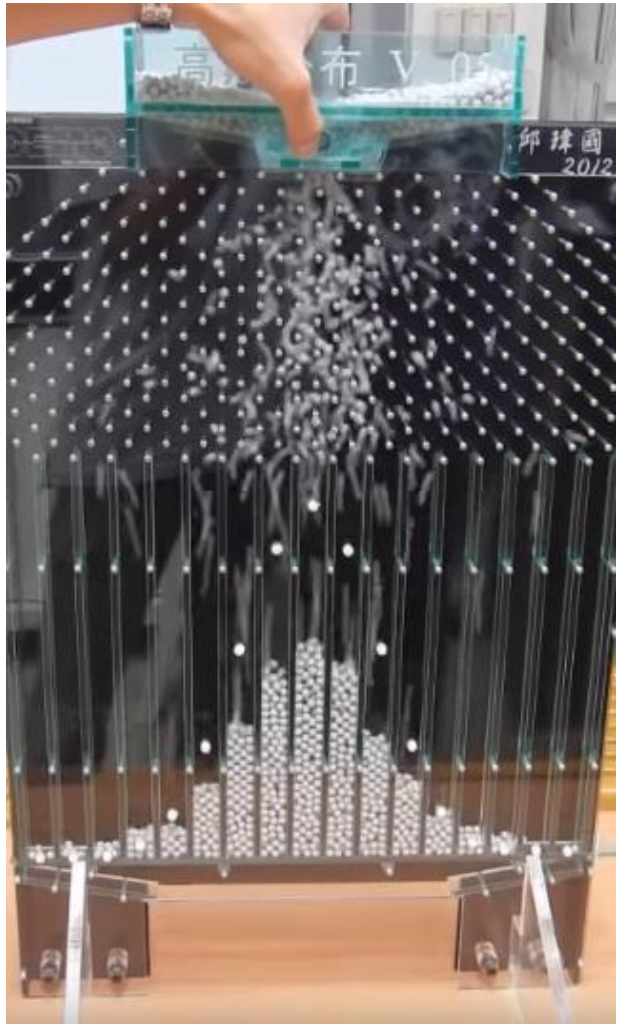
If we know how samples should be distributed, then we can use that distribution in confidence intervals and hypothesis testing (next Lecture).

Image from http://industrialscenery.blogspot.com/2015/11/boiler-explosion-of-c-t-1-3020-on-may.html

# Parametric Distributions

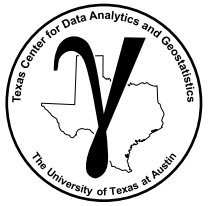
## Other Comments About Parametric Distributions

- Known, useful statistical properties
  - If Gaussian distributed all marginal and conditionals are Gaussian
- Commonly occur in nature
  - E.g., central limit theorem (to be discussed)
- Encompass a range of assumptions
  - E.g., uniform distribution is the maximum uncertainty distribution



Pachinko with BBs results in a Gaussian distribution.

Image captured from https://www.youtube.com/watch?v=AjI_LcQOOs4.




# Parametric Distributions

## We Will Cover the Following Distributions

1. Uniform Distribution
2. Binomial or Bernoulli Distribution
3. Poisson Distribution
4. Normal or Gaussian Distribution
5. Logarithmic Normal or Log–Normal Distribution
6. Student's t Distribution
7. $\chi 2$ or Chi-Squared Distribution
8. Fisher's $F$ Distribution

**All of these will be useful for common data analytics workflows.**

- There are many others that we could cover.

# Uniform Distribution

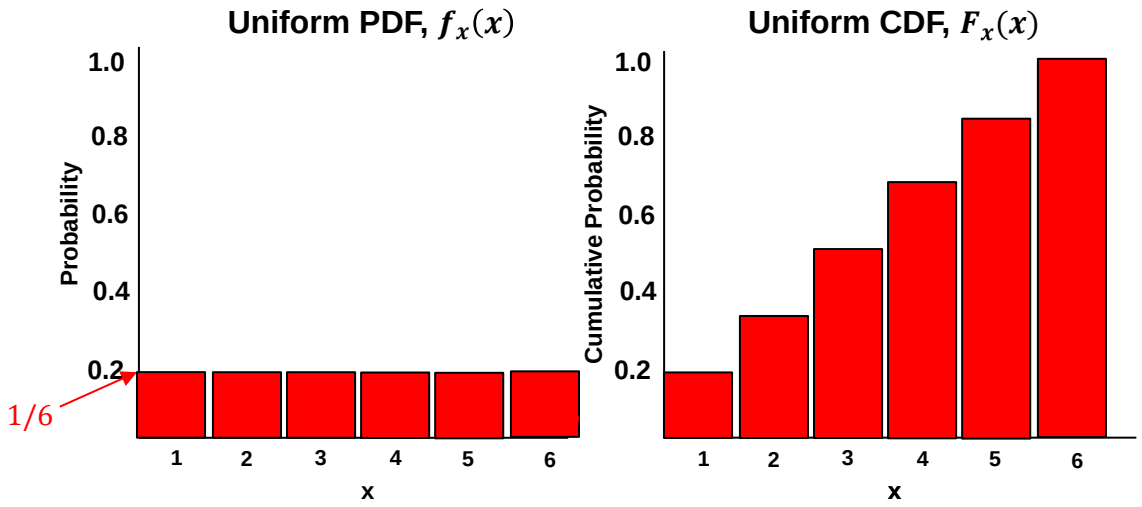
**Multiple outcomes, all equally likely.**

# Uniform Distribution

## Discrete Uniform Distribution

PDF: $f_x(x) = \frac{1}{n} = constant$ CDF: $F_x(x) = \frac{1}{n}x$

Example: a die with 6 faces, N = 6, x takes discrete values 1, 2,…,6.




Discrete uniform PDF and CDF for a 6-sided die.

Comments:

- Maximum uncertainty distribution. Used when very little information available.
- Distribution is discrete, can only be evaluated at integer values of $x$

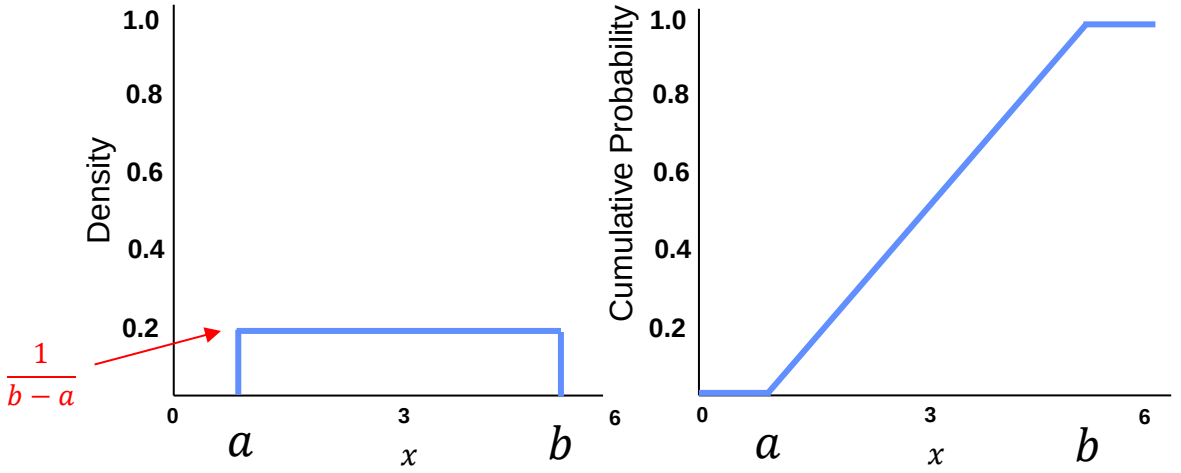
# Uniform Distribution

## Continuous Uniform Distribution

PDF: $$f_x(x) = \begin{cases} \frac{1}{b-a} & if\ a \le x \le b \\ 0 & otherwise \end{cases}$$

CDF: $$F_x(x) = \begin{cases} 0 & if\ x \le a \\ \frac{x-a}{b-a} & if\ a \le x \le b \\ 1 & if\ x \ge b \end{cases}$$

Parameterized by the minimum ($a$) and maximum ($b$) values.




Continuous uniform PDF, $f_x(x)$ (left), and CDF, $F_x(x)$ (right).

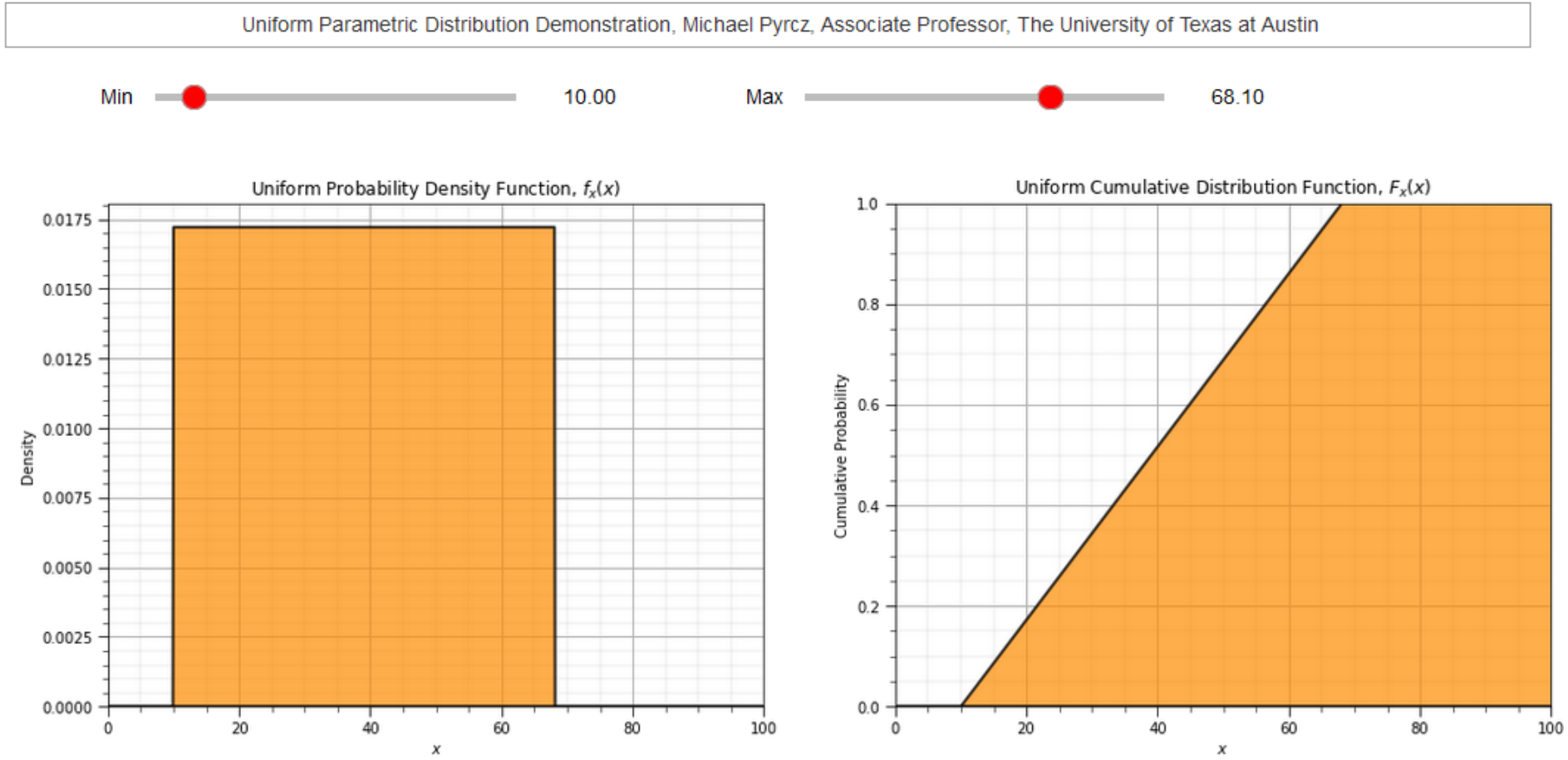
| | |
|---|---|
| Mean: | $\frac{1}{2}(a+b)$ |
| Variance: | $\frac{1}{12}(b-a)^2$ |
| Skewness: | $0$ |
| Excess Kurtosis: | $-\frac{6}{5}$ |

Comments:

- Maximum uncertainty distribution. Used when very little information available.
- Random cumulative probability values for Monte Carlo simulation are uniform.

# Uniform Distribution Demonstration in Python

Try out the uniform parametric distribution by changing the parameters and observing the PDF and CDF.




Continuous uniform parametric distribution PDF and CDF in file Interactive_ParametricDistributions.ipynb.

# Binomial Distribution

**Multiple trials with 2 possible outcomes?**

**Hockey game shot, score or not.**

**Exploration drilling, discovery or not.**

# Binomial Distribution

## Binomial Distribution

Use for multiple trials, $N$, with binary $(0,1)$ outcomes. Discrete distribution for probability of $\boldsymbol{x}$ successes (and $N - x$ failures), where $p$ is the probability of success, $(1 - p)$ is the probability of failure.

PDF:

$N$ choose $x$ combinatorial

probability $x$ successes

probability $N - x$ failures

$$f_x(x) = \binom{N}{x} p^x (1-p)^{N-x}$$

Probability of $x$ successes over $N$ trials

CDF:

sum over cases $1, \ldots, x$

$$F_x(x) = \sum_{i=1}^{x} \binom{N}{x} p^x (1-p)^{N-x}$$

Probability of $x$ or less successes over $N$ trials

| | |
|---|---|
| Mean: | $Np$ |
| Variance: | $Np(1-p)$ |
| Skewness: | $\frac{(1-p)-p}{\sqrt{Np(1-p)}}$ |
| Excess Kurtosis: | $\frac{1-6p(1-p)}{Np(1-p)}$ |

- Recall for independent events:

$$P(A, B, C) = P(A) \times P(B) \times P(C)$$

The $\binom{N}{x}$ operator:

- Accounts for the multiple possible combinations:

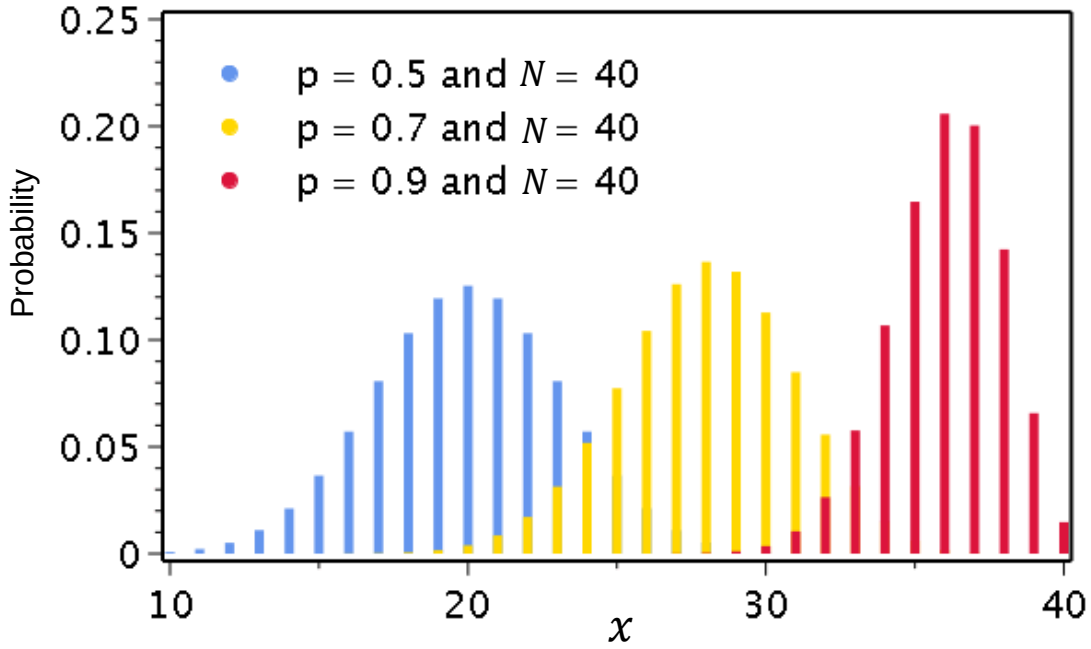
**E.g.** $\binom{3}{2} = \frac{3 \cdot 2 \cdot 1}{2 \cdot 1\ (1)} = 3,$

**E.g., for coin toss:** $HHT, HTH, THH$

# Binomial Distribution

## Example Binomial Distributions

- 40 trails with $P(success) = \{0.5, 0.7, 0.9\}$



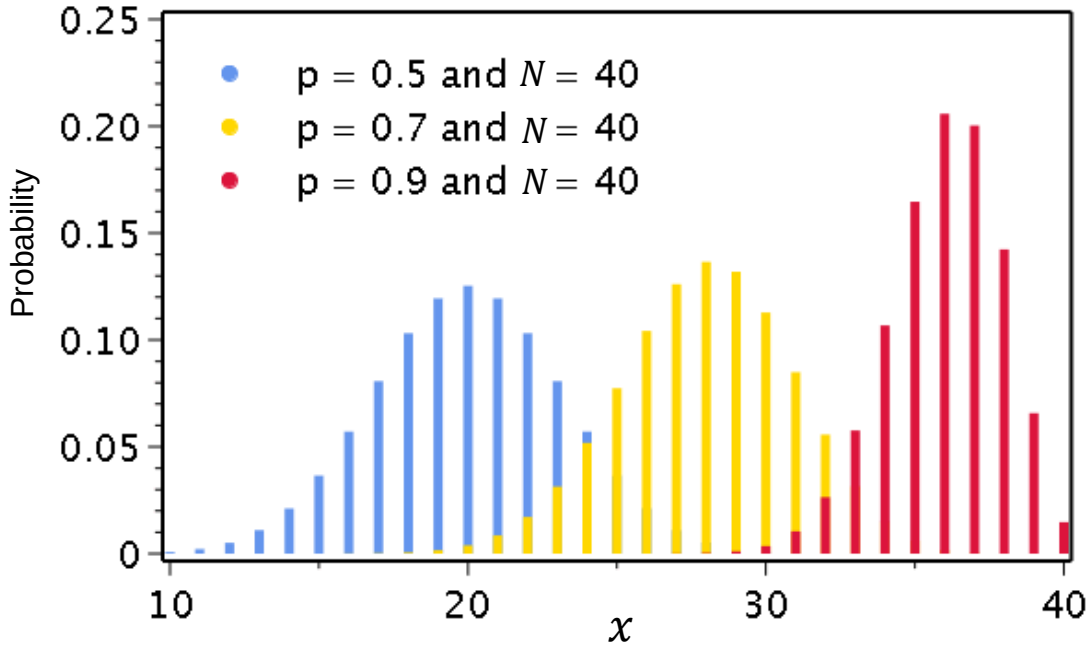

Example binomial PDFs with variable probability of success.

What is the expected value?

# Binomial Distribution

## Example Binomial Distributions

- 40 trails with $P(success) = \{0.5, 0.7, 0.9\}$




Example binomial PDFs with variable probability of success.

What is the expected value? $E\{X\} = N \times p$

Also, the variance is $\sigma^2 = N \times p \times (1 - p)$

Image from https://de.maplesoft.com/support/help/maple/view.aspx?path=MathApps/NormalApproximationOfBinomialDistribution

# Binomial Distribution

**Calculating the binomial distribution example.**

1. Assess success criteria (e.g., 2 successful wells of 3 drilled, $P(S) = 0.2$)

2. Calculate all possible outcomes and identify cases matching criteria.
   - SSS, SSF, SFS, SFF, FSS, FSF, FFS, FFF

3. Combinations are equiprobable, so 3 x probability of one case.

4. Probability of one case, $P(S,S,F) = P(S)P(S)P(F) = P(S)^{n_s}\left(1-P(S)\right)^{(n-n_s)}$

5. This is the binomial distribution PDF:

$$f(x) = \binom{N}{x} p^x (1-p)^{N-x} = \binom{3}{2} \cdot (0.5)^2 \cdot (0.5)^1$$

COMBIN in Excel
Math.comb($N$, $x$) in Python

$$= 3\ cases \cdot 0.125\ probability\ for\ each\ case$$

$$\boldsymbol{P(A,B,C) = P(A)P(B)P(C)}$$
assuming independence.

# Binomial Coefficient, $\binom{N}{x}$

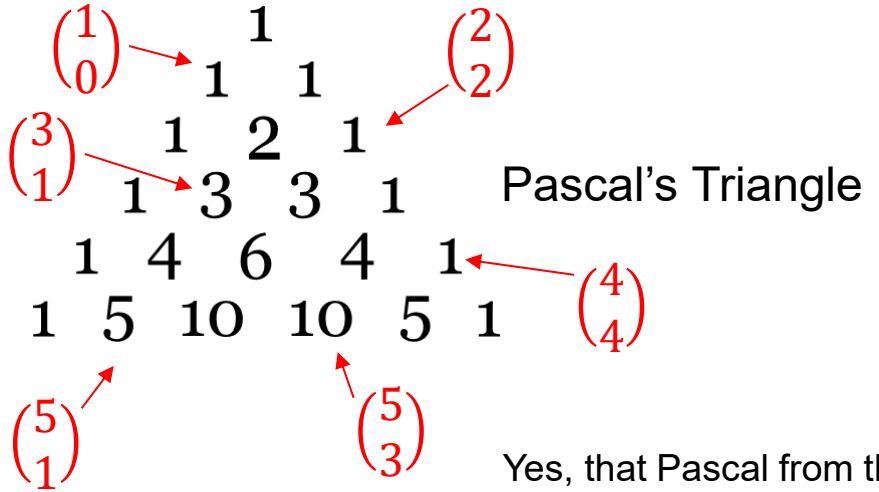
The choose operator, $\binom{N}{x}$, read as “N choose x”

The binary coefficient represents the number of combinations for $N$ trials with $\boldsymbol{x}$ success (S).

For $\binom{3}{1}$, {SFF, FSF, FFS}, = 3

$$\binom{N}{x} = \frac{\boldsymbol{N!}}{\boldsymbol{x!(N-x)!}}$$

Note the factorial operator, $N!$ , $N! = \prod_{k=1}^{N} k$, e.g., $5! = 5 \times 4 \times 3 \times 2 \times 1$




Pascal’s Triangle

Yes, that Pascal from the gambler’s dispute with Fermat in 1654.

# Binomial Distribution Example

**Exploration drilling failure rate is 70%. Your company is drilling up 10 prospects.**

What is the probability of only 3 successful discoveries? Recall:

COMBIN in Excel
Math.comb($N$, $x$) in Python

What is the probability of no success for your exploration program?

Now try out the binomial distribution function in:

# Binomial Distribution Example

**Exploration drilling failure rate is 70%. Your company is drilling up 10 prospects.**

What is the probability of only 3 successful discoveries?

COMBIN(10,3) = 120, 120 x 0.3^3 x (1.0 – 0.3)^7 = 0.267
BINOM.DIST(3,10,0.3,FALSE) = 0.267

What is the probability of no success for your exploration program?

COMBIN(10,0) = 1, 1 x 0.3^0 x (1.0 – 0.3)^10 = 0.028
BINOM.DIST(0,10,0.3,FALSE) = 0.028

Now try out the binomial distribution function in:

Excel - BINOM.DIST($x, N, p$,cumulative = False)

Python - scipy.stats.binom.pmf($x, N, p$)

# Binomial Distribution Example

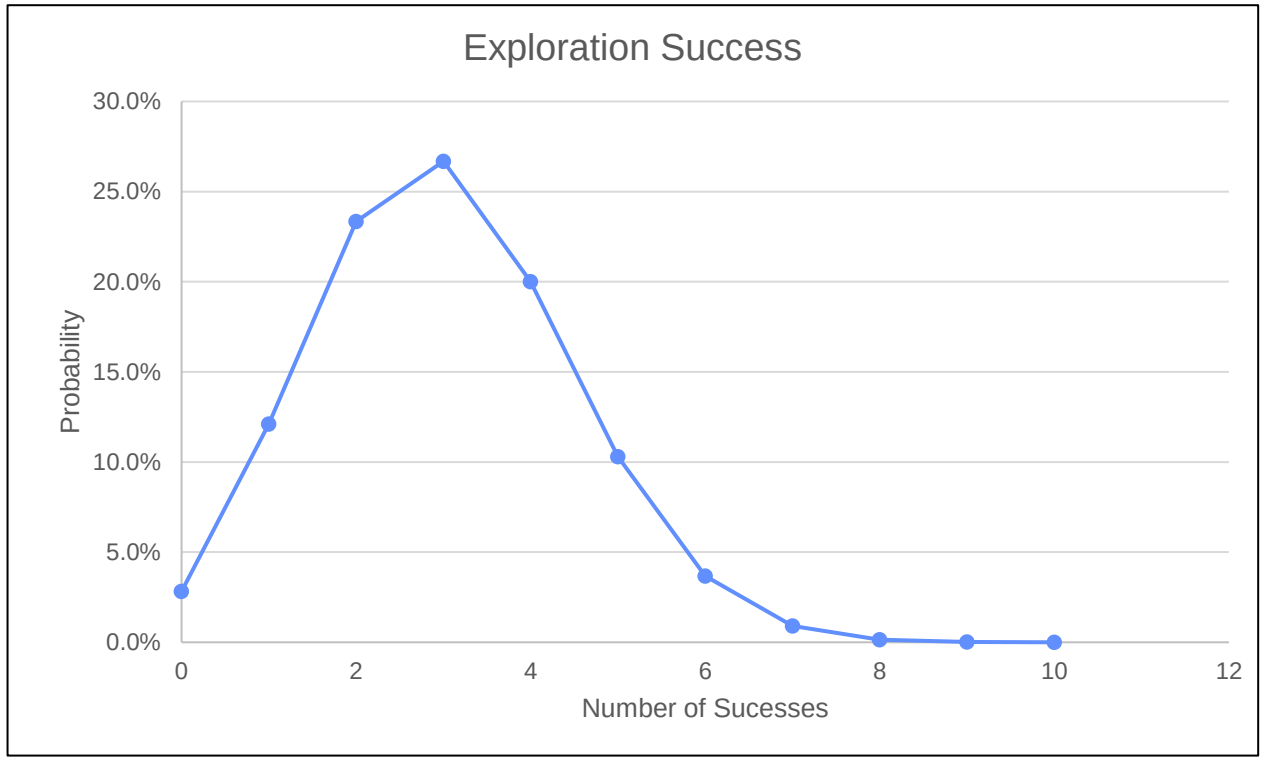
**Exploration drilling failure rate is 70%. Your company is drilling up 10 prospects.**

**Plotting Binomial Distribution Excel**

1. Make column of outcomes [0,1,…,n]

2. Use binomial.dist(outcome,n,probability,cumulative) to calculate the probability of each discrete outcome.

3. Select X column and then Y column and insert a plot.




Binomial PDF for exploration drilling program. Note, lines shown for visualization of shape, but the Binomial distribution is discrete only, valid for 0 and positive integers.

**Plotting Binomial Distribution Python**

1. Make 1D ndarray of outcomes [0,1,…,n] with np.arange(0,n+1,1)

2. Use scipy.stats.binom.pmf(cumul,n=1, p=0.2) to calculate the probability of each outcome.

3. Plot with plt.plot(outcomes,binomial probabilities)

# Binomial Distribution Demonstration in Excel

Try out the binomial parametric distribution by changing the parameters and observing the PDF and probability of a string of failures.

**Working with Discrete Outcomes? Binomial Distribution Demo, Michael Pyrcz, University of Texas at Austin, @GeostatsGuy**

For discrete outcomes with indepence between trials and stationary probability we can appy the binomial distribution. Here's an example for a 20 well exploration program with 20% (red) and 40% (blue) probability of success.
The resuls show (1) the binomial probability density functions with the probabilities for number of successful wells and (2) the probabilities of streaks of consecutive failures

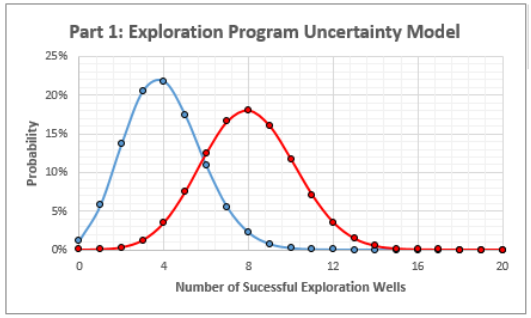
| Number of Wells | Case 1: Probability of Success | Case 2: Probability of Success |
|---|---|---|
| 20 | 20% | 40% |

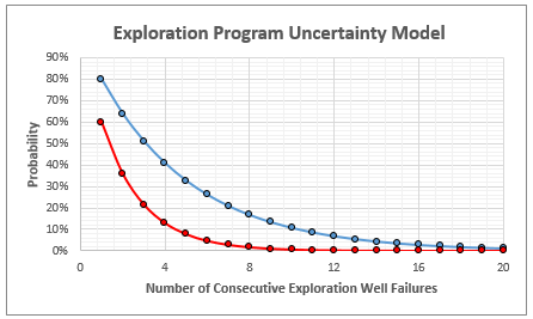
Part 1: Binomial PDF

| Number of Success | Case 1: Probability | Case 2: Probability |
|---|---|---|
| 0 | 1% | 0% |
| 1 | 6% | 0% |
| 2 | 14% | 0% |
| 3 | 21% | 1% |
| 4 | 22% | 3% |
| 5 | 17% | 7% |
| 6 | 11% | 12% |
| 7 | 5% | 17% |
| 8 | 2% | 18% |
| 9 | 1% | 16% |
| 10 | 0% | 12% |
| 11 | 0% | 7% |
| 12 | 0% | 4% |
| 13 | 0% | 1% |
| 14 | 0% | 0% |
| 15 | 0% | 0% |
| 16 | 0% | 0% |
| 17 | 0% | 0% |
| 18 | 0% | 0% |
| 19 | 0% | 0% |
| 20 | 0% | 0% |

Part 2: Probability of All Failures

| Number of Failures | Number of Trials | Case 1: Probability | Case 2: Probability |
|---|---|---|---|
| 1 | 1 | 80% | 60% |
| 2 | 2 | 64% | 36% |
| 3 | 3 | 51% | 22% |
| 4 | 4 | 41% | 13% |
| 5 | 5 | 33% | 8% |
| 6 | 6 | 26% | 5% |
| 7 | 7 | 21% | 3% |
| 8 | 8 | 17% | 2% |
| 9 | 9 | 13% | 1% |
| 10 | 10 | 11% | 1% |
| 11 | 11 | 9% | 0% |
| 12 | 12 | 7% | 0% |
| 13 | 13 | 5% | 0% |
| 14 | 14 | 4% | 0% |
| 15 | 15 | 4% | 0% |
| 16 | 16 | 3% | 0% |
| 17 | 17 | 2% | 0% |
| 18 | 18 | 2% | 0% |
| 19 | 19 | 1% | 0% |
| 20 | 20 | 1% | 0% |




**Demonstration Instructions**

Modify the case 1 and case 2 probablity of exploration success and observed the number of exploration success PDF and probability of n consecutive failures.

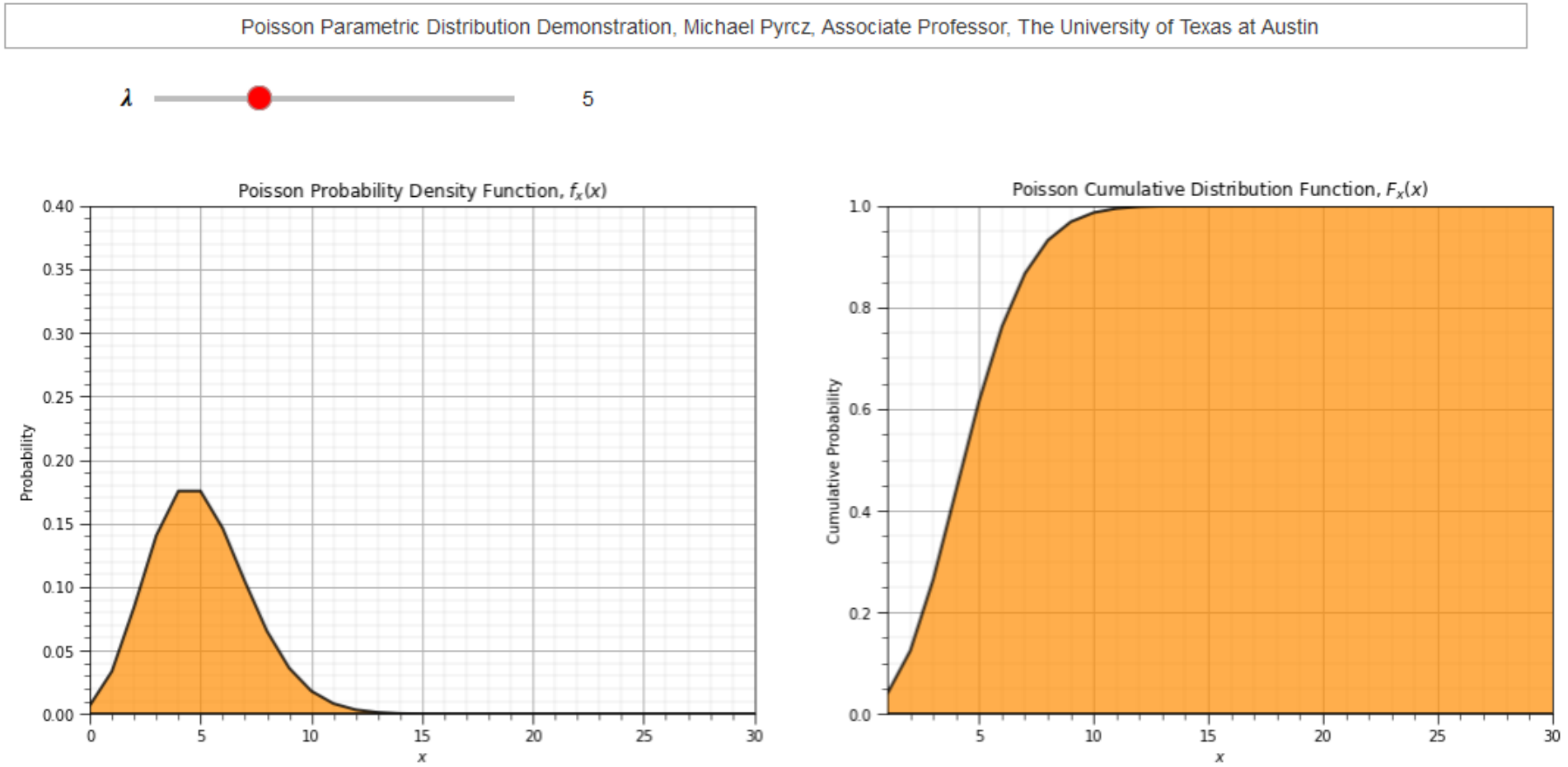
**What Should You Observe?**

The binomial PDF is centered on the expectation ( Prob(Success) x Number of Trials ) and the variance is:
the number of trials x probability of success x probability of failure.
Streaks of consecutive failures are quite possible when failure probability is high.

Binomial parametric distribution demonstration in Excel, file is Binomial_Demo.ipynb.

# Binomial Distribution Demonstration in Python

Try out the Poisson parametric distribution by changing the parameters and observing the PDF and CDF.



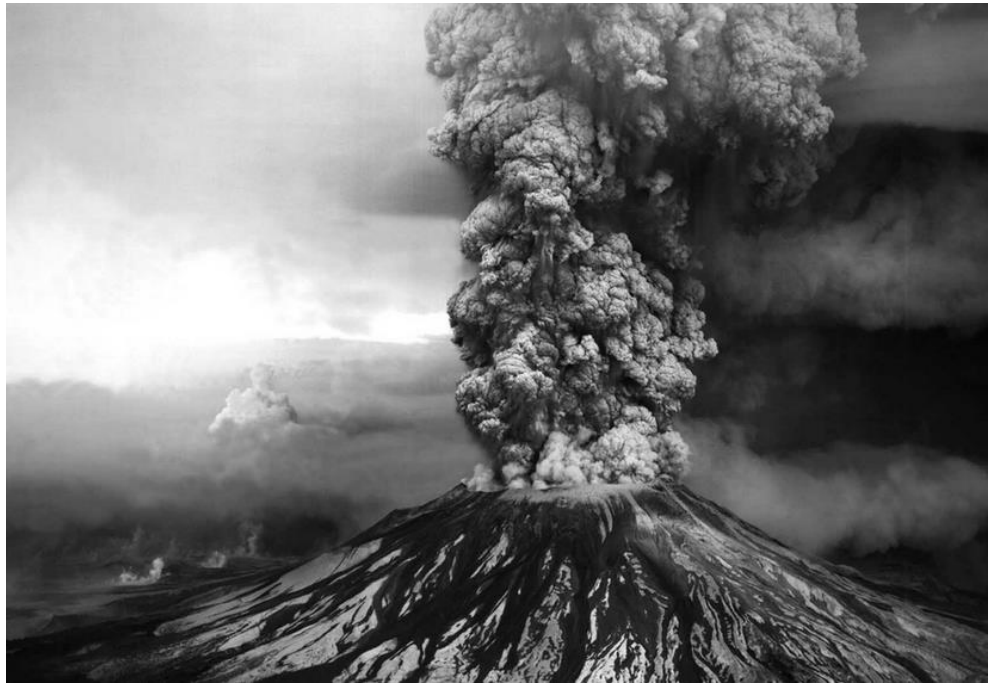

Poisson parametric distribution PDF and CDF in file Interactive_ParametricDistributions.ipynb.

# Poisson vs. Binomial Distribution

Both are for binary outcomes – success / failure. Both are discrete distributions, probability for integer frequencies.

- volcanoes are binary, erupting or not erupting
- yet, they don't make $N$ attempts

Mount St. Helens, May 17th 1980.



Mount St. Helens, May 18th 1980.

Images from https://rarehistoricalphotos.com/eruption-mount-st-helens-1980/

# Poisson vs. Binomial Distribution

## Poisson Distribution

- When given the **average number of successes** of an event happening **per interval over time or space**
- Assumptions: constant rate, independent events, binary outcomes.

## Compare with Binomial Distribution

- Use Binomial when you know the **exact probability of success for a single trial,** and you want successes **over a number of trials**.
- i.e., you could not use Binomial distribution for a volcano or machine failure.

Definition from http://www.statisticshowto.com/poisson-distribution/

# Poisson Distribution

**Poisson Distribution:**

PDF: $$f_x(x) = \frac{e^{-\lambda}\lambda^x}{x!}$$

CDF: $$F_x(x) = \sum_{i=1}^{x} \frac{e^{-\lambda}\lambda^x}{x!}$$

**Parameters:**

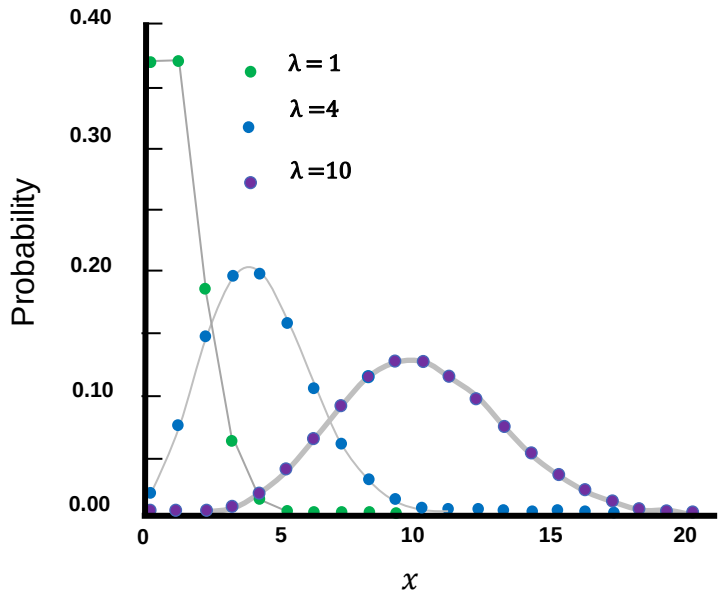
where $\lambda$ is the average number of occurrences over an interval (time period or space) and $x$ is the actual number of occurrences.

- Interval size is set, assumed.

**Expectation and Variance:**

$$E[X] = \lambda$$

$$Var[X] = \lambda$$




Example Poisson PDFs with variable probability of success.

| | |
|---|---|
| Mean: | $\lambda$ |
| Variance: | $\lambda$ |
| Skewness: | $\sqrt{\frac{1}{\lambda}}$ |
| Excess Kurtosis: | $\frac{1}{\lambda}$ |

Image from https://en.wikipedia.org/wiki/Poisson_distribution#/media/File:Poisson_pmf.svg

# Poisson Distribution Exercise

**Poisson Distribution Exercise**

When drilling through a specific formation there have been an average of 3 fluid loss incidents. What is the probability of 5 fluid loss incidents for the next well (Hint: use Excel FACT() and EXP() commands)?

What is the probability of no fluid loss incidence?

By hand and try out function:

Excel - POISSON.DIST($x, \lambda$,cumulative = False)

Python - scipy.stats.binom.pmf($x, mu = \lambda$)

# Poisson Distribution Exercise

**Poisson Distribution Exercise**

When drilling through a specific formation there have been an average of 3 fluid loss incidents. What is the probability of 5 fluid loss incidents for the next well (Hint: use Excel FACT() and EXP() commands)?

$$f(x) = \frac{e^{-\lambda}\lambda^{x}}{x!} \qquad f(5) = \frac{e^{-3}3^{5}}{5!} = 0.10$$

POISSON.DIST(5,3,FALSE) = 0.101

What is the probability of no fluid loss incidence?

$$f(0) = \frac{e^{-3}3^{0}}{0!} = 0.05$$

note that 0! = 1.

POISSON.DIST(0,3,FALSE) = 0.050

By hand and try out function:

Excel - POISSON.DIST($x, \lambda$,cumulative = False)

Python - scipy.stats.poisson.pmf($x, mu = \lambda$)

# Poisson Distribution Exercise

## Poisson Distribution Exercise

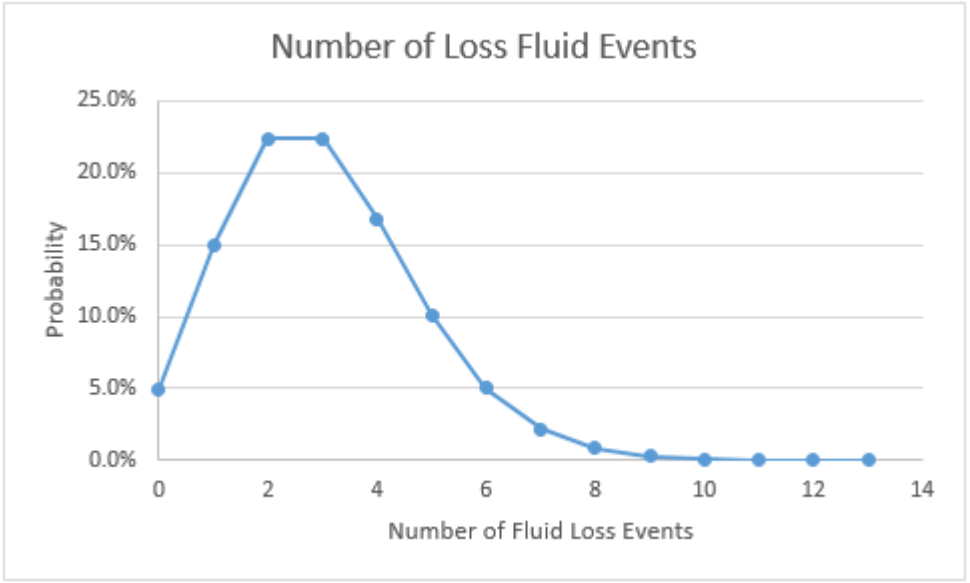
When drilling through a specific formation there have been an average of 3 fluid loss incidents. What is the probability of 5 fluid loss incidents for the next well (Hint: use Excel FACT() and EXP() commands)?

**Plotting Poisson Distribution Excel**

1. Make column of outcomes [0,1,…,n]

2. Use poisson.dist(outcome,lambda,cumulative) to calculate the probability of each discrete outcome.

3. Select X column and then Y column and insert a plot.




Poisson PDF for fluid loss incidents. Note, lines shown for visualization of shape, but the Binomial distribution is discrete only, valid for 0 and positive integers.

**Plotting Poisson Distribution Python**

1. Make 1D ndarray of outcomes [0,1,…,n] with np.arange(0,n+1,1)

2. Use scipy.stats.poisson.pmf(cumul,mu=3) to calculate the probability of each outcome.

3. Plot with plt.plot(outcomes, Poisson probabilities)

# Poisson Distribution Demonstration in Excel

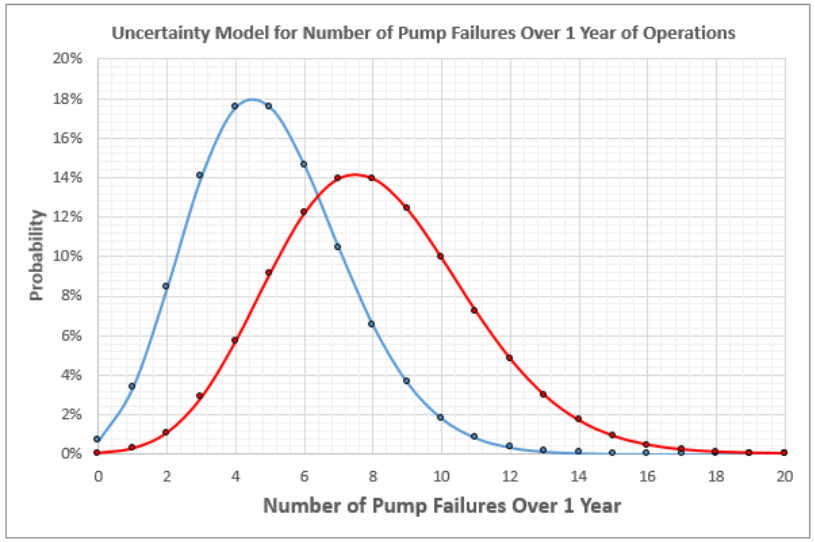
**How Many Events in a Fixed Interval of Time and Space? Poisson Distribution Demo, Michael Pyrcz, University of Texas at Austin, @GeostatsGuy**

Do you have a problem setting with events that occur independent of each other and with a constant rate over a fixed time or space. If you have the average number of events over that time or space interval use the Poisson distribution to predict the complete discrete probability density of number of events over that interval of time or space.

Comments: the intervals may specified in time, volume, area or length. Each subunit of the interval at the scale of the event is binary, labelled as the event or not event, more than one event may not occur over the same time or space. Ensure that the event are not correlated before use. For example, if **pump failures** in your project are independent from each other and occur at a constant rate then one may apply the Poisson distribution to model the discrete PDF for failures over an interval of time.

Consider case 1 (blue) with a low rate and Case 2 (red) with a higher failure rate.

| | Case 1: Average Number of Failures in 1 Year | Case 2: Average Number of Failures in 1 Year |
|---|---|---|
| λ | 5 | 8 |




**Poisson PDFs for Both Cases**

| Number of Pump Failures in 1 Year (X) | Case 1: Probability P(X) | Case 2: Probability P(X) |
|---|---|---|
| 0 | 1% | 0% |
| 1 | 3% | 0% |
| 2 | 8% | 1% |
| 3 | 14% | 3% |
| 4 | 18% | 6% |
| 5 | 18% | 9% |
| 6 | 15% | 12% |
| 7 | 10% | 14% |
| 8 | 7% | 14% |
| 9 | 4% | 12% |
| 10 | 2% | 10% |
| 11 | 1% | 7% |
| 12 | 0% | 5% |
| 13 | 0% | 3% |
| 14 | 0% | 2% |
| 15 | 0% | 1% |
| 16 | 0% | 0% |
| 17 | 0% | 0% |
| 18 | 0% | 0% |
| 19 | 0% | 0% |
| 20 | 0% | 0% |

**Demonstration Instructions**

Modify the average number of pump failures for Case 1 and Case 2 (yellow boxes).

**What Should You Observe?**

The expectation, **E[X]**, of the PDF is the average number of events (**λ**), the only parameter for the Poisso
Also, the variance of the PDF is equal to **λ**, resulting in this heteroscedastic relationship. $\frac{\sigma_Y^2}{E[X]} = 1$

**Expectation Calculation**

| $p_i \times x_i$ Case 1 | $p_i \times x_i$ Case 2 |
|---|---|
| 0.00 | 0.00 |
| 0.03 | 0.00 |
| 0.17 | 0.02 |
| 0.42 | 0.09 |
| 0.70 | 0.23 |
| 0.88 | 0.46 |
| 0.88 | 0.73 |
| 0.73 | 0.98 |
| 0.52 | 1.12 |
| 0.33 | 1.12 |
| 0.18 | 0.99 |
| 0.09 | 0.79 |
| 0.04 | 0.58 |
| 0.02 | 0.39 |
| 0.01 | 0.24 |
| 0.00 | 0.14 |
| 0.00 | 0.07 |
| 0.00 | 0.04 |
| 0.00 | 0.02 |
| 0.00 | 0.01 |
| 0.00 | 0.00 |
| Expectated Number of Failures | |
| 5.00 | 8.00 |
| $\Sigma p_i \times x_i$ | |

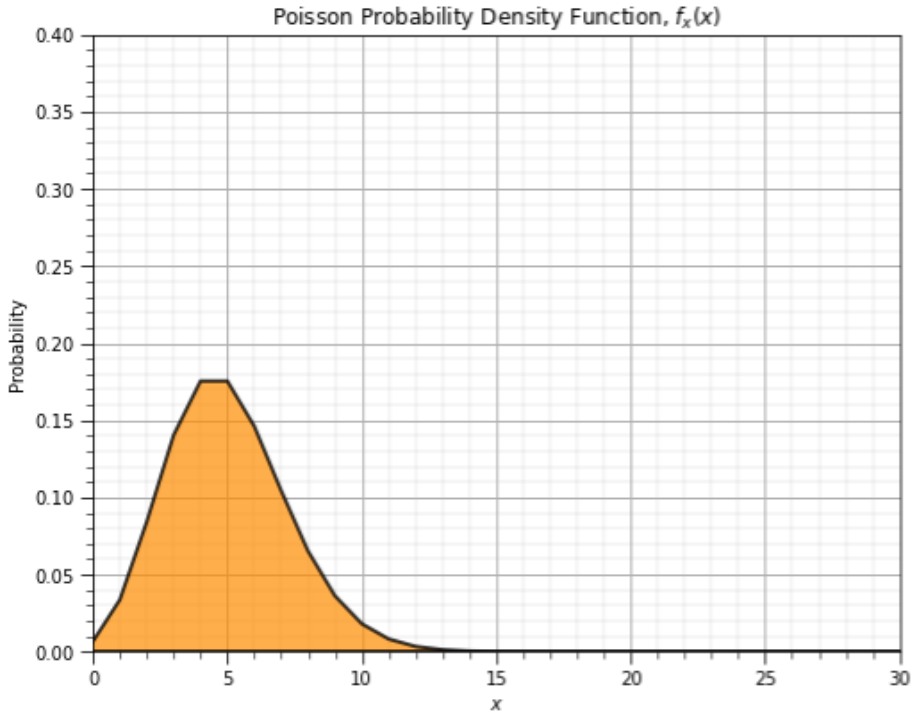
Binomial parametric distribution demonstration in Excel, file is Poisson_Demo.ipynb.

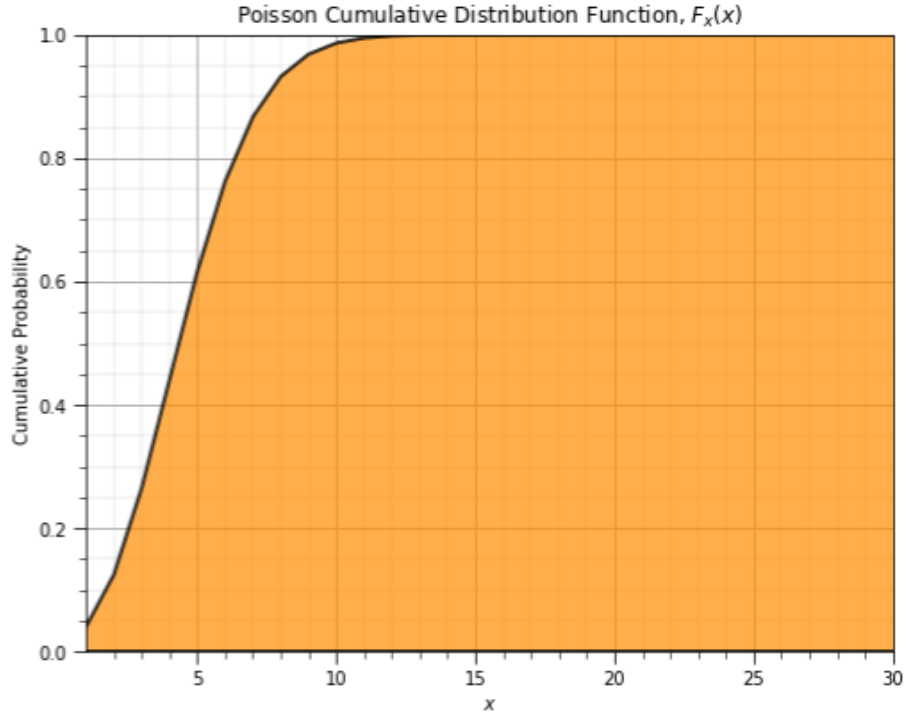
# Poisson Distribution Demonstration in Python

Try out the Poisson parametric distribution by changing the parameters and observing the PDF and CDF.

Poisson Parametric Distribution Demonstration, Michael Pyrcz, Associate Professor, The University of Texas at Austin

$\lambda$ 5



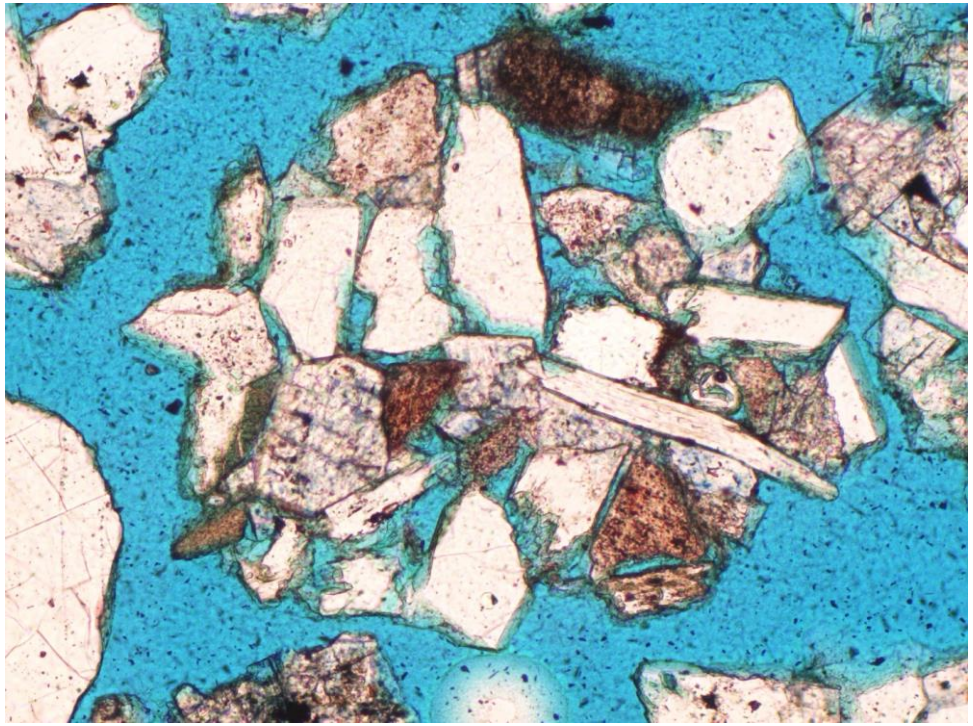

Poisson parametric distribution PDF and CDF in file Interactive_ParametricDistributions.ipynb.

# Gaussian / Normal Distribution

## Gaussian / Normal Distribution

- Commonly observed in natural phenomenon

Distribution of Error



Summation or Averaging

# Gaussian / Normal Distribution

## Gaussian Distribution:

PDF: $f_x(x) = \frac{1}{\sigma\sqrt{2\pi}} exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$

mean

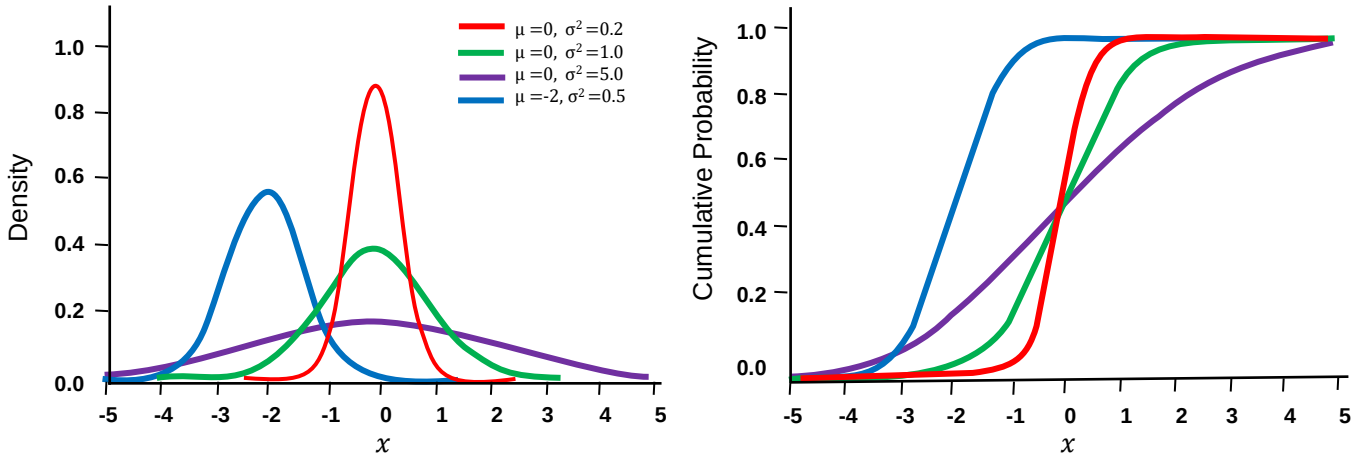
standard deviation

CDF: $F_x(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} exp\left[-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2\right] dy$

$-\infty < \boldsymbol{x} < +\infty$

## Parameters:

where $\mu$, is the mean and $\sigma$, is the standard deviation.



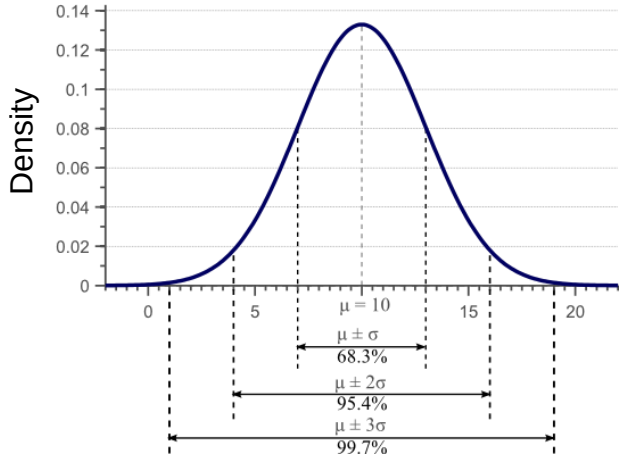

Example Gaussian PDFs and CDFs with variable mean and variance.

| | |
|---|---|
| Mean: | $\boldsymbol{\mu}$ |
| Variance: | $\boldsymbol{\sigma^2}$ |
| Skewness: | 0 |
| Excess Kurtosis: | 0 |

# Gaussian / Normal Distribution

**Gaussian Distribution:**

- Shorthand for a Normal Distribution is $N[mean, st.dev]$, $N(\mu, \sigma^2)$.
- Distribution is unbounded, no min nor max $-\infty < \boldsymbol{x} < +\infty$, extremes are very unlikely, some type of truncation is often applied.
- e.g., max is $3\,\sigma$ and min is - $3\,\sigma$, $3 \times$ standard deviation




**Gaussian PDF**

$\mu \pm \sigma = 68.3\%$

$\mu \pm 2\sigma = 95.4\%$

$\mu \pm 3\sigma = 99.7\%$

Gaussian PDF and probability for 1, 2 and 3 standard deviation intervals.

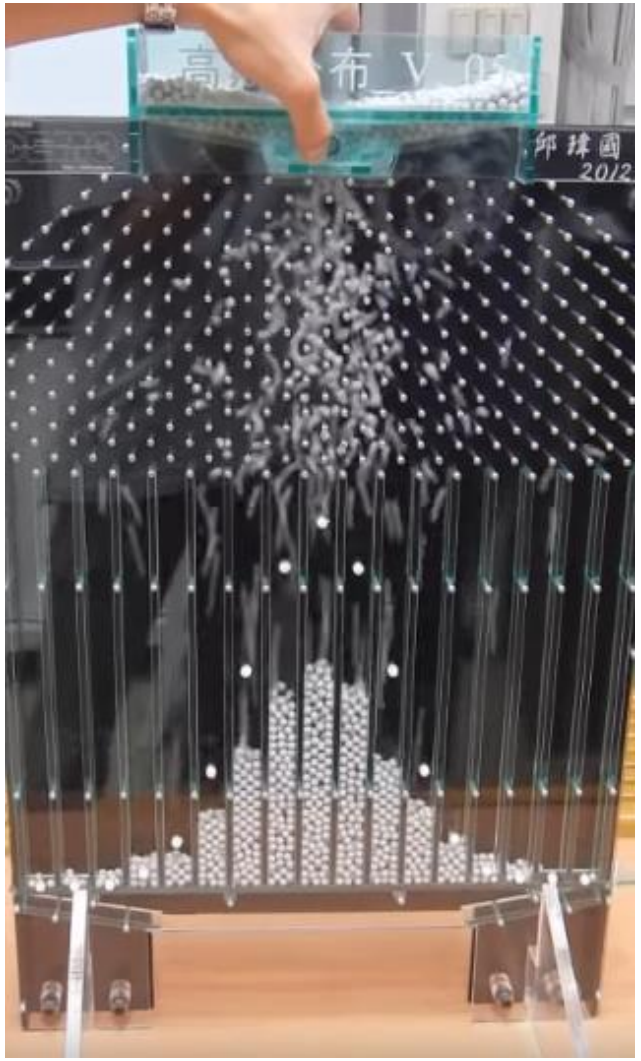
# Central Limit Theorem

**Central Limit Theorem (CLT)**

- the summation / average of multiple random variables tends towards a Gaussian distributed

$$\sum_{i=1}^{m} X_i$$ is distributed as Gaussian as $m$ is large.

- this occurs practically with 3-4 independent variables ($m \geq 3$)
- some reservoir properties may tend to Gaussian distributed due to CLT (e.g., porosity is the average of pore space vs. grains over smaller volumes).
- this may be disrupted by combining multiple populations and trends.



Pachinko with BBs results in a Gaussian distribution.

# Central Limit Theorem Demonstration in Excel

**Demonstration:**

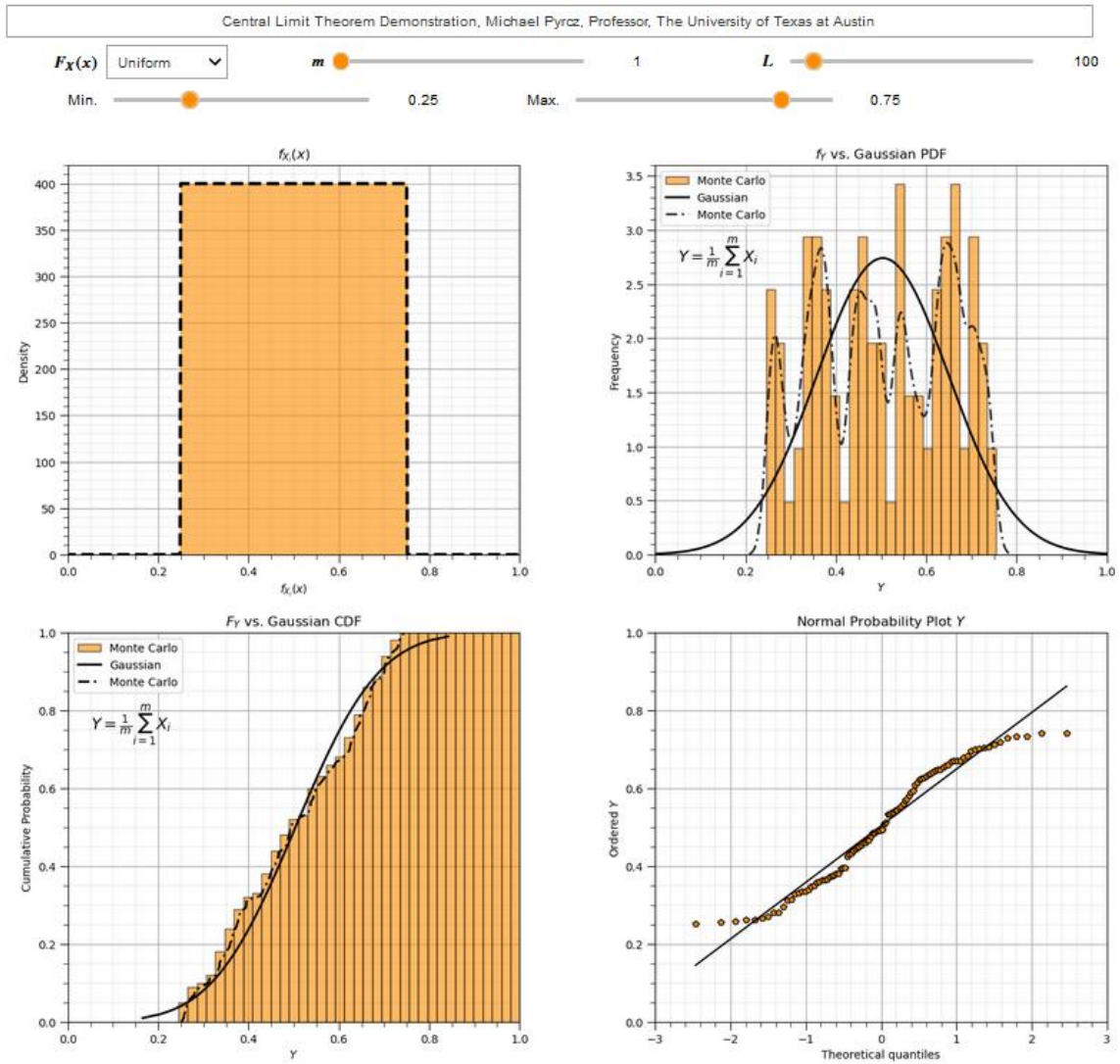
Take $X_1, X_2, \ldots, X_{20}$ random variables

$$y = \sum_{i=1}^{m} X_i$$

average them to make a new random variable (RV)

Plot their CDF's:

- $X_1$ is uniformly distributed ($m = 1$)
- How is $X_1 + X_2$ distributed ($m = 2$)?
- How is $X_1 + X_2 + X_3$ distributed ($m = 3$)?
- and so on?




Demonstration of the central limit theorem in file Interactive_Central_Limit_Theorem.ipynb.

# Central Limit Theorem Demonstration in Excel

**Demonstration:**

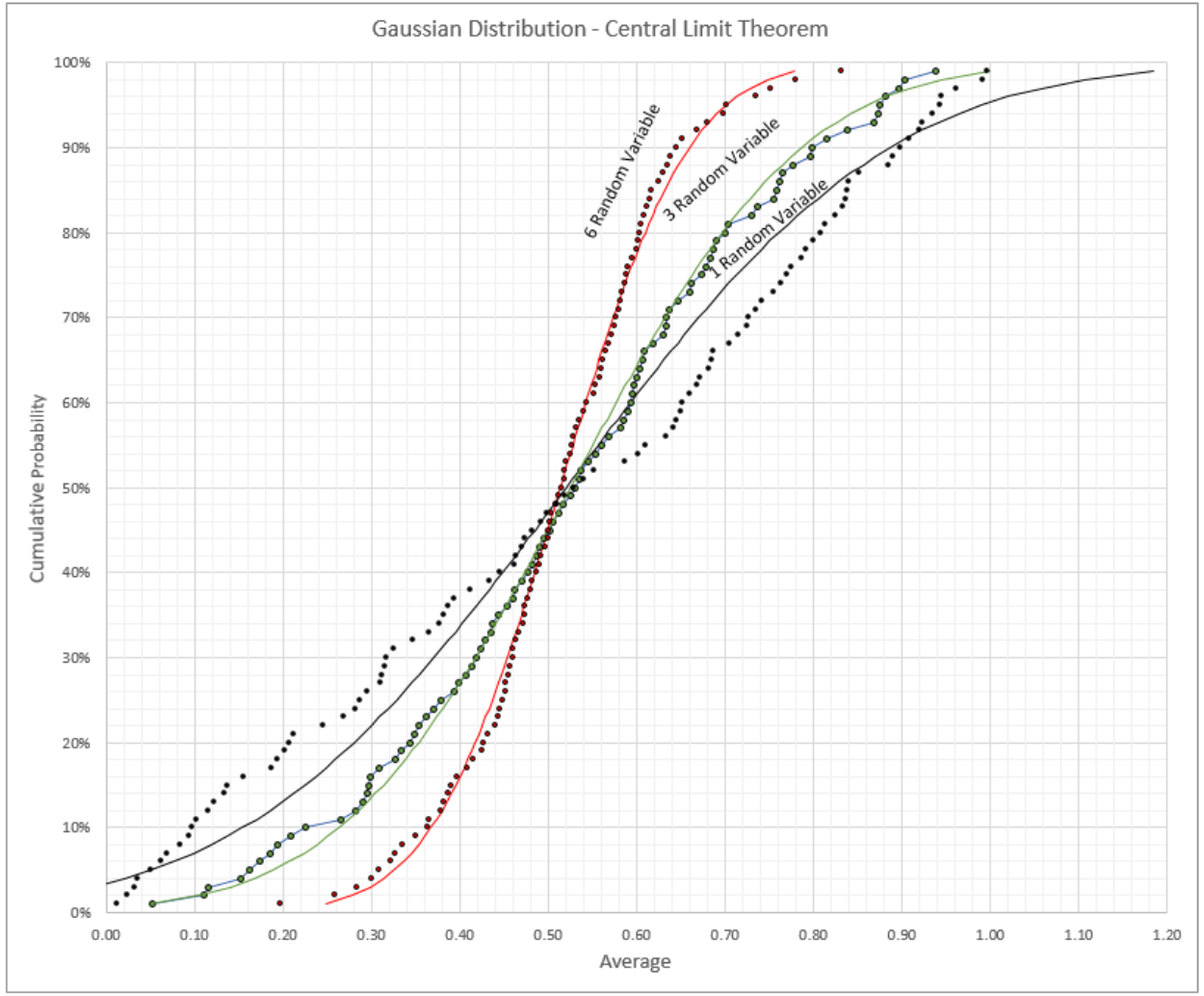
Take $X_1, X_2, \ldots, X_{20}$ random variables

$$y = \sum_{i=1}^{m} X_i$$

average them to make a new random variable (RV)

Plot their CDF's:

- $X_1$ is uniformly distributed ($m = 1$)
- How is $X_1 + X_2$ distributed ($m = 2$)?
- How is $X_1 + X_2 + X_3$ distributed ($m = 3$)?
- and so on?




How to make distributions from scratch? MakingDistributions.xlsx.

# Fitting Parametric Distributions

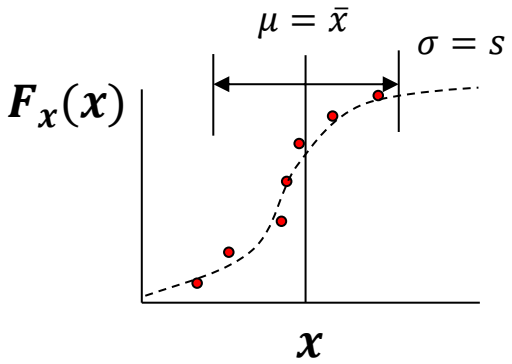
## Using data to infer the parameter distribution parameters

1. **Calculate the parameters from the data**: e.g., mean and standard deviation for a Gaussian distribution from the data.

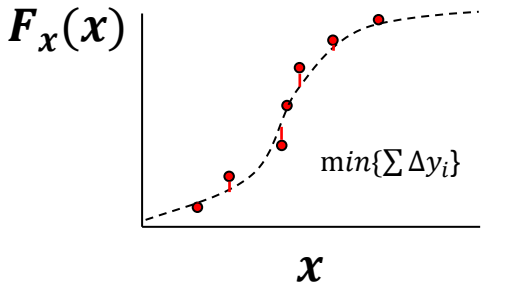
2. **Least squares fitting**: calculate the distribution parameters such that error is minimized.

3. **Maximum likelihood estimation**: calculate the parameters such that the probability of observing the data, $P(data \mid model)$, is maximized. Assumes independent, identically distributed (i.i.d.) data.

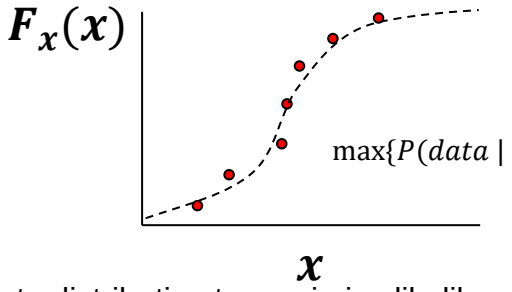
For this course we will just calculate the parameters from the data.




Calculate distribution parameters from data.




Calculate distribution to minimize error at data.




Calculate distribution to maximize likelihood of the data.

**For Reference Only – Not on Exam**

# Gaussian Distribution Exercise

## Fit and Predict with a Gaussian Distribution

Fit a Gaussian distribution to the following porosity samples

- 5%, 7%, 10%, 12%, 14%, 14%, 15%, 19%, 20%, 23%

What is the sample mean and standard deviation?

What is the probability of a porosity less than 18%, hint: use Excel NORM.DIST(x,mean,stdev,cumulative) with cumulative = True.

**Hint: calculate the mean and standard deviation to fit. We could also use more robust fitting methods (e.g., maximum likelihood).**

# Gaussian Distribution Exercise

## Fit and Predict with a Gaussian Distribution

Fit a Gaussian distribution to the following porosity samples

- 5%, 7%, 10%, 12%, 14%, 14%, 15%, 19%, 20%, 23%

What is the sample mean and standard deviation?

$$\bar{x} = 13.9, \sigma = 5.7$$

What is the probability of a porosity less than 18%, hint: use Excel NORM.DIST(x,mean,stdev,cumulative) with cumulative = True.

$$P(x \leq 18) = 76.4\%$$

**Hint: calculate the mean and standard deviation to fit. We could also use more robust fitting methods (e.g., maximum likelihood).**

# Gaussian Distribution Example

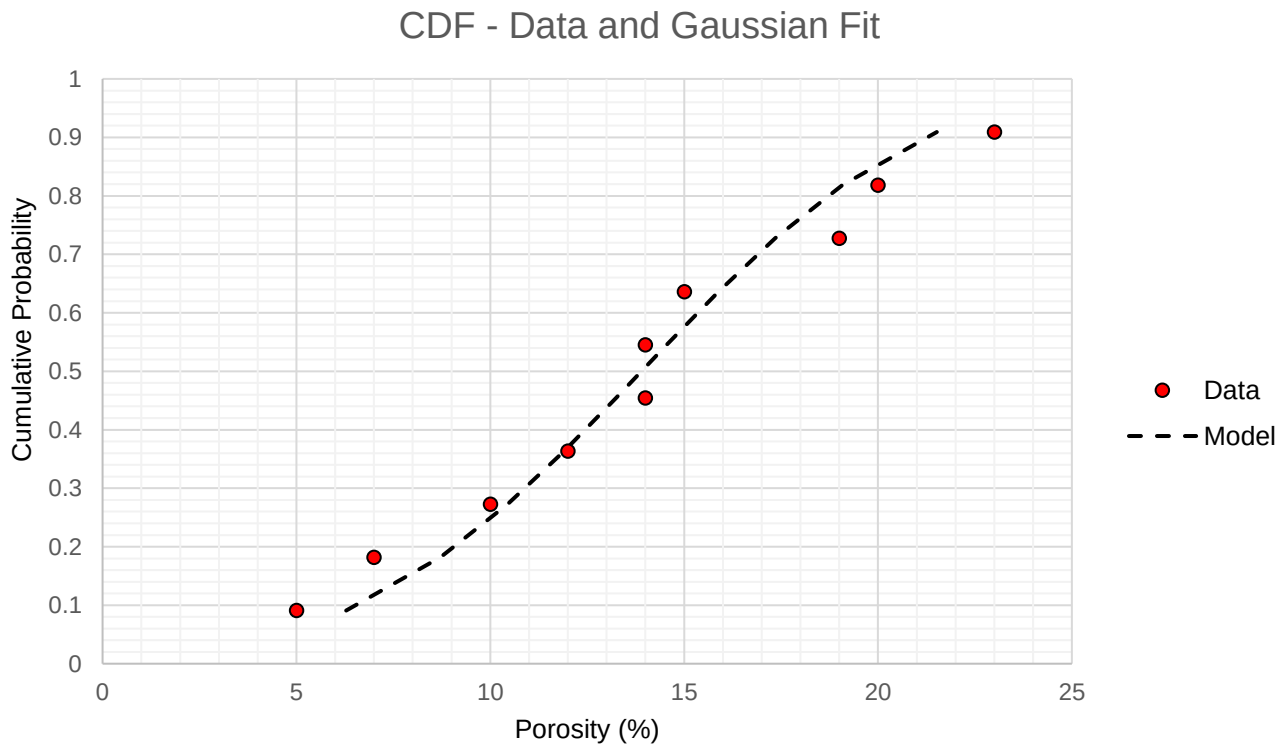
Here's what you get if you calculate the data CDF ($F_i = {}^{i}/_{n+1}$) and display the data and 'fit' Gaussian distribution CDF.

**Plotting Normal Distribution Excel**

1. Make column of cumulative probability values [0.01,0.02,…,0.99]

2. Use norm.inv(cumul,mean,stdev) to calculate the percentile values for each cumulative probability.

3. Select X then Y column and insert a plot.




Data and Gaussian fit parametric distribution CDFs.

**Plotting Lognormal Distribution Python**

1. Make 1D ndarray of cumulative probability values [0.01,0.02,…,0.99] with np.linspace(0.01,0.99,100)

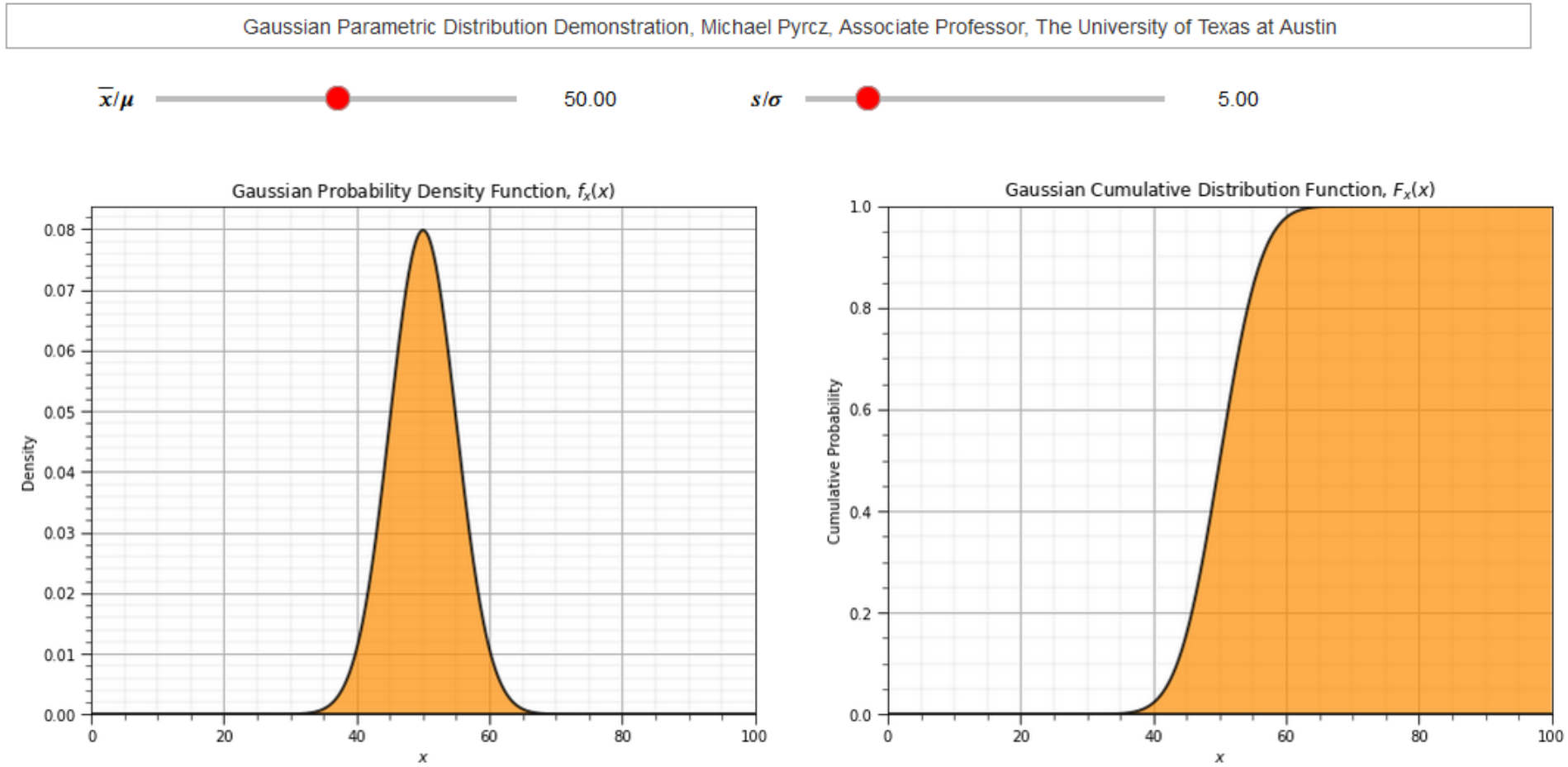
2. Use scipy.stats.norm.ppf(cumul,loc=mean,scale=stdev) to calculate the percentile values for each cumulative probability.

3. Plot with plt.plot(norm inverse values in x, cumululative probabilities)

Note: Fit by just calculating the mean and standard deviation from the data. $N[13.9\%, 5.7\%]$

# Gaussian Distribution Demonstration in Python

Try out the Gaussian parametric distribution by changing the parameters and observing the PDF and CDF.



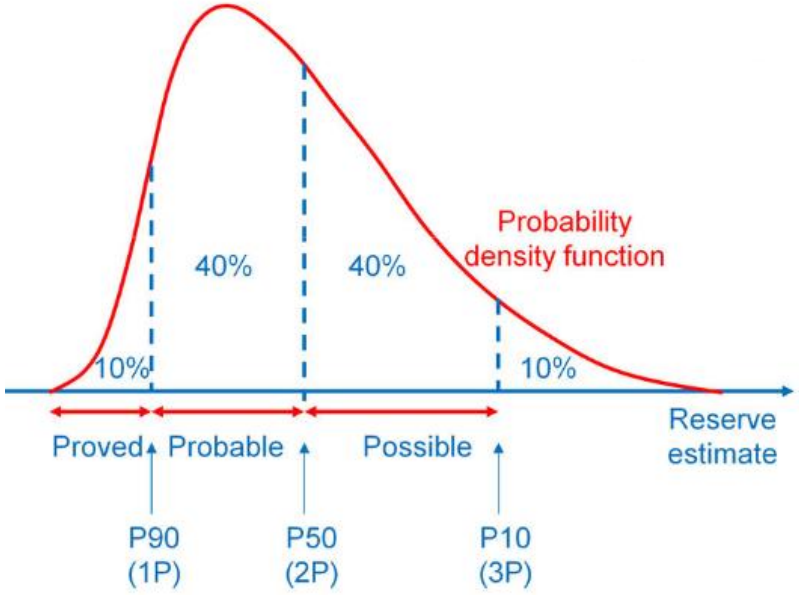

Gaussian parametric distribution PDF and CDF in file Interactive_ParametricDistributions.ipynb.

# Lognormal Distribution

## Log Normal Distribution

- Continuous probability distribution for phenomenon that are normally distributed after a natural log transform, log-normally distributed.

Milk production from cows, there are very high productivity cows.

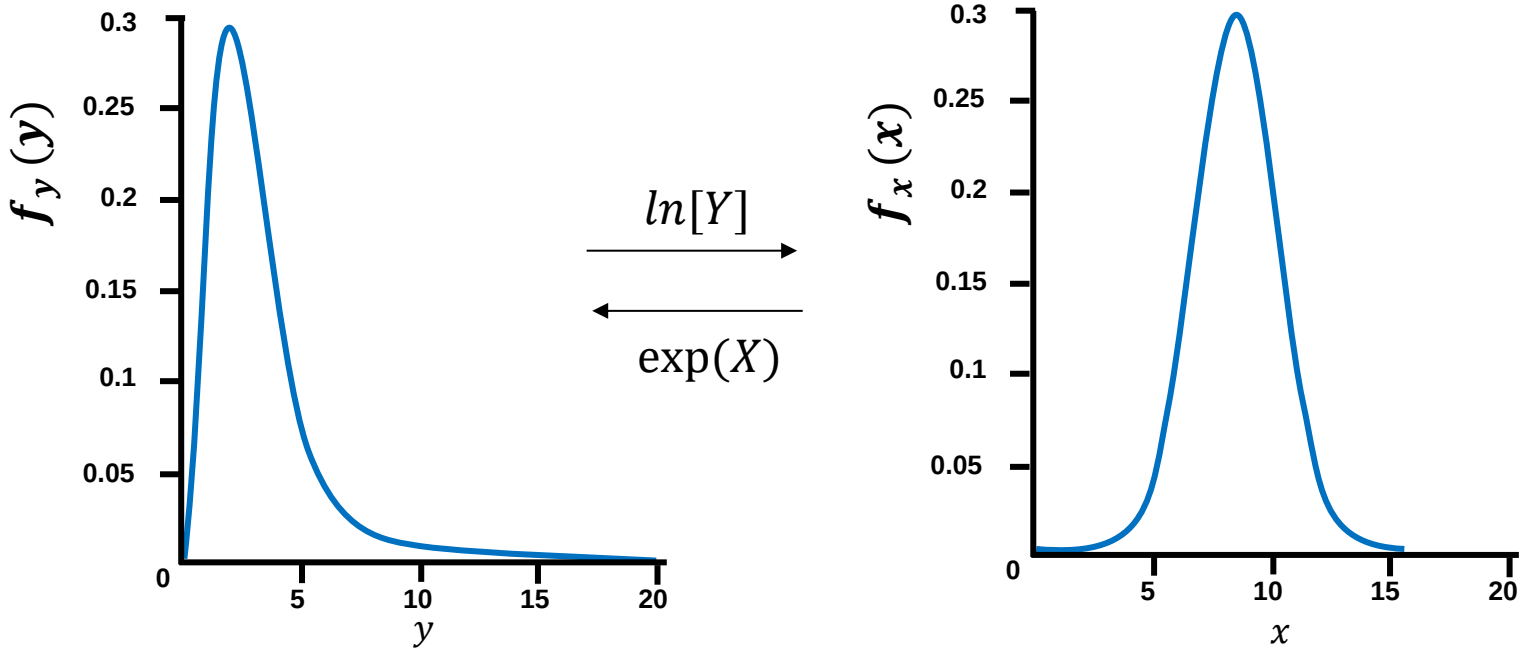



Oil Reserves Uncertainty Distribution

# Lognormal vs. Normal Distribution

## Lognormal vs. Normal Distribution

- Take a Gaussian distributed random variable, $X$, and apply $Y = \exp(X)$ transform
- Random variable $Y$ is lognormal distributed
- Apply $ln[Y] = ln[\exp(X)] = X$, back to normal distributed original variable.




Log normal PDF (left) and normal (Gaussian) distribution (right).

# Lognormal Distribution

## Lognormal Distribution:

mu = mean of $ln(X)$

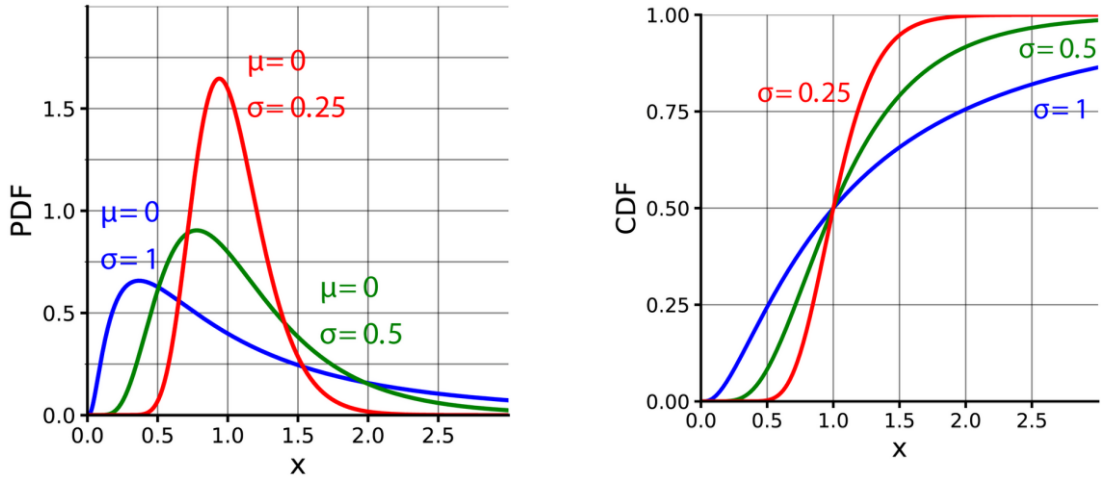
PDF: $$f_x(x) = \frac{1}{\sigma\sqrt{2\pi}x} exp\left[-\frac{1}{2}\left(\frac{ln(x)-\mu}{\sigma}\right)^2\right]$$

CDF: $$F_x(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} \frac{1}{y} exp\left[-\frac{1}{2}\left(\frac{ln(x)-\mu}{\sigma}\right)^2\right] dy$$

Note: we use an approximation for this integration.

sigma = standard deviation of $\ln(X)$

**Parameters:** mu, $\mu$, and sigma, $\sigma$, are the mean and standard deviation of the associated Gaussian distribution, $ln(X)$, not $Y$!




Example lognormal PDFs and CDFs with variable mean and variance.

Images from https://en.wikipedia.org/wiki/Log-normal_distribution

# Lognormal Distribution

**Calculating summary statistics and fitting parameters.**

mu = mean of Ln(X)

$$f_x(x) = \frac{1}{\sigma\sqrt{2\pi}x} exp\left[-\frac{1}{2}\left(\frac{ln(x)-\mu}{\sigma}\right)^2\right]$$

sigma = standard deviation of Ln(X)

Calculating the mean/expectation and variance given mu and sigma:

$$E[X] = exp\left[\mu + \frac{\sigma^2}{2}\right] \qquad var[X] = (E[X])^2\left[e^{\sigma^2} - 1\right]$$

- important if you want to know the central tendency and dispersion.

Calculating the mu and sigma given the expectation and variance:

**mu,** $\mu = ln\left(\frac{E[X]^2}{\sqrt{var[X] + (E[X])^2}}\right)$ **sigma,** $\sigma = \sqrt{ln\left(\frac{var[X]}{(E[X])^2} + 1\right)}$

- important if you want to fit a lognormal based on summary statistics of a set of values.

# Lognormal Distribution Exercise

The permeability (in mD) of a reservoir is lognormal distributed with mu, $\mu$ = 2.0 and sigma, $\sigma$ = 1.8.

What is the mean and the standard deviation of the distribution?

**Recall:**

$$E[X] = exp\left[\mu + \frac{\sigma^2}{2}\right] \qquad var[X] = (E[X])^2\left[e^{\sigma^2} - 1\right]$$

What is the probability of permeability less than 100 mD?

Excel LOGNORM.DIST(mean=mu,std=sigma,) with cumulative set to TRUE)

Python scipy.stats.lognorm.cdf(100,s=sigma,scale=math.exp(mu))

**Warning:** Excel says mean and stdev, but the inputs are actually mu and sigma!, SciPy implementation is not straightforward.

# Lognormal Distribution Exercise

The permeability (in mD) of a reservoir is lognormal distributed with mu, $\boldsymbol{\mu}$ = 2.0 and sigma, $\boldsymbol{\sigma}$ = 1.8.

What is the mean and the standard deviation of the distribution?

$$E[X] = exp\left[\mu + \frac{\sigma^2}{2}\right] = exp\left[2.0 + \frac{1.8^2}{2}\right] = 37.3 \text{ mD}$$

$$\text{StDev}[X] = E[X]\sqrt{\left[e^{\sigma^2} - 1\right]} = (37.3)\sqrt{\left[e^{(1.8)^2} - 1\right]} = 184.9 \text{ mD}$$

What is the probability of permeability less than 100 mD?

Excel - LOGNORM.DIST(100,2.0,1.8,TRUE)

$$P(x \leq 100\ mD) = 0.93$$

**Warning:** Excel says mean and stdev, but the inputs are actually mu and sigma, SciPy stats implementation is overly complicated.

# Lognormal Distribution Exercise

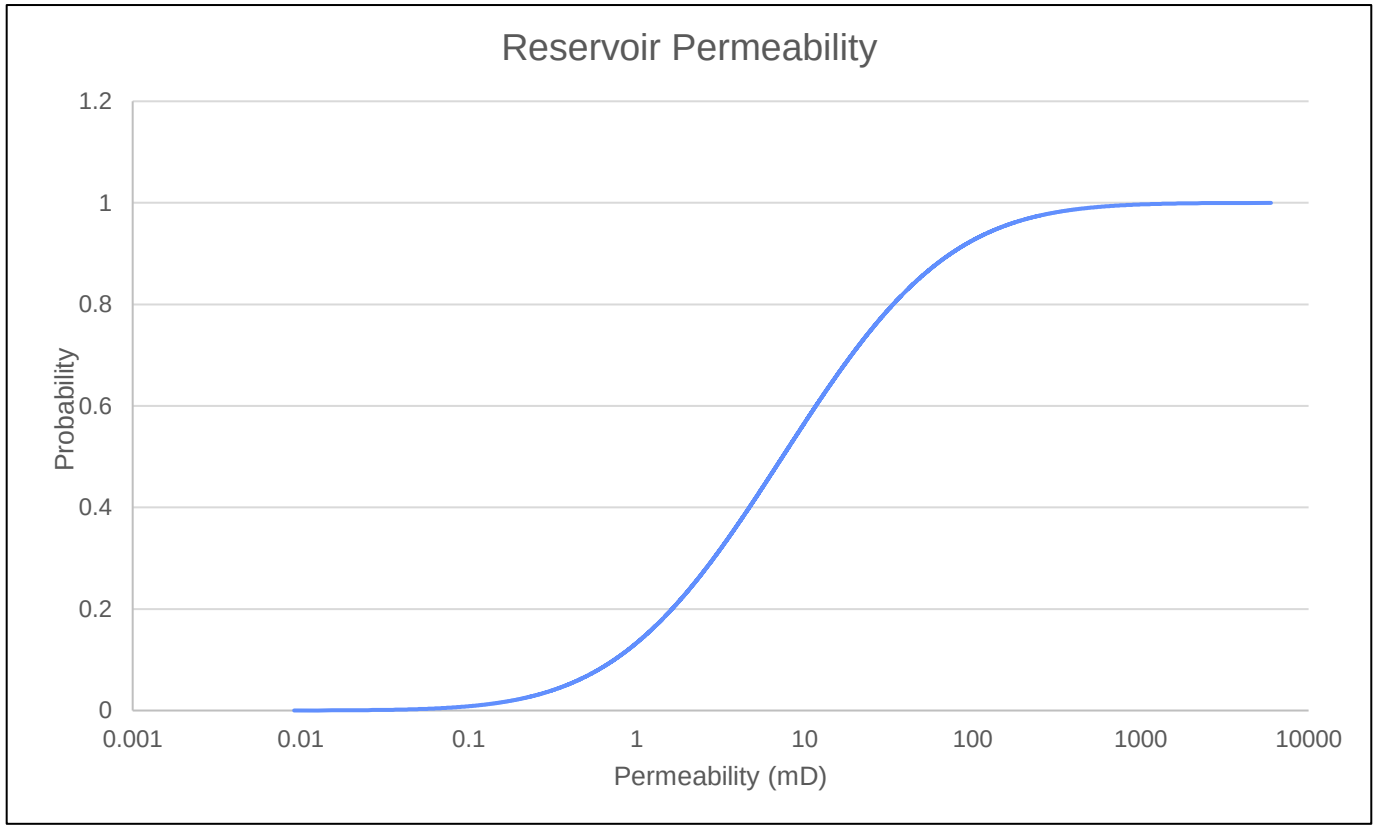
The permeability (in mD) of a reservoir is lognormal distributed with mu, $\mu$ = 2.0 and sigma, $\sigma$ = 1.8.

**Plotting Lognormal Distribution Excel**

1. Make column of cumulative probability values [0.01,0.02,…,0.99]

2. Use lognorm.inv(cumul,mu,sigma) to calculate the percentile values for each cumulative probability.

3. Select X then Y column and insert a plot.




Calculate cumulative probability for a range of outcomes and plot.

**Plotting Lognormal Distribution Python**

1. Make 1D ndarray of cumulative probability values [0.01,0.02,…,0.99] with np.linspace(0.01,0.99,100)

2. Use scipy.stats.lognorm.ppf(cumul,s=sigma,scale=math.exp(mu)) to calculate the percentile values for each cumulative probability.

3. Plot with plt.plot(inverse lognorm values in y,cumulative probabilities)

# Lognormal vs. Normal Distribution Demonstration in Excel

**Workflow:**

1. Start with Gaussian standard normal, N{0,1}
2. Rescale $\bar{x} = mu, s = sigma$ (affine correction)
3. Apply exponentiation, EXP[ ]
4. Fit with Log Normal distribution
5. Compare predicted and actual mean and standard deviation
6. Apply natural log Ln[ ] to return to the Gaussian distribution.

**The Lognormal Distribution, Interpretation of Its Parameters and Relationship to the Normal Distribution.**

Michael Pyrcz, the University of Texas at Austin, Geostatistical Reservoir Modeling Class

| 1. Lognormal Distribution Parameters | |
|---|---|
| mu | 1 |
| sigma | 0.4 |

| Percentile | 2. Standard Normal Distribution | 3. Normal Distribution [mu,sigma] | 4. Our Lognormal Distribution | 5. Check with Excel Function Lognormal | 6. Normal Distribution [mu, sigma] | 7.Standard Normal Distribution N[0,1] |
|---|---|---|---|---|---|---|
| | N[0,1] | N[0,1] * sigma + mu = N[mu,sigma] | EXP(N[mu,sigma]) = LogN[mu,sigma] | Excel Command LogNorm.Inv(p,mu,sigma) | LN(EXP(N[mu,sigma])) = N[mu,sigma] | (LN(EXP(N[mu,sigma] - mu))) / sigma = N[0,1] |
| 0.005 | -2.576 | -0.030 | 0.970 | 0.970 | -0.030 | -2.576 |
| 0.015 | -2.170 | 0.132 | 1.141 | 1.141 | 0.132 | -2.170 |
| 0.025 | -1.960 | 0.216 | 1.241 | 1.241 | 0.216 | -1.960 |
| 0.035 | -1.812 | 0.275 | 1.317 | 1.317 | 0.275 | -1.812 |
| 0.045 | -1.695 | 0.322 | 1.380 | 1.380 | 0.322 | -1.695 |
| 0.985 | 2.170 | 1.868 | 6.476 | 6.476 | 1.868 | 2.170 |
| 0.995 | 2.576 | 2.030 | 7.617 | 7.617 | 2.030 | 2.576 |
| Average | 0.000 | 1.000 | 2.941 | 2.941 | 1.000 | 0.000 |
| Standard Deviation | 0.999 | 0.399 | 1.212 | 1.212 | 0.399 | 0.999 |

**Instructions and Workflow of this Lognormal Distribution Excel Demo**

1. Set the mu and sigma parameters.
2. A standard normal distribution (mean = 0.0 and standard deviation = 1.0), N[0,1], is calculated from the percentile list in the table (note most rows are hidden).
3. Standard normal distribution is adjusted to have a mean = mu and a standard deviation = sigma, N[mu,sigm].
4. Exponentiation of the N[mu,sigma] distribution converts it to a log normal distribution parameterized by mu and sigma, LogN[mu,sigma].
   The lognormal distribution mean and standard deviation may be calculated directly from mu and sigma parameters as shown below.

| | | | | | |
|---|---|---|---|---|---|
| Derived Mean | 2.945 | mean = Exp(mu + sigma^2/2) | Actual Mean | 2.941 |
| Derived St. Dev. | 1.227 | st. dev. = mu x sqrt(EXP(sigma^2)-1) | Actual St. Dev. | 1.212 |

5. Check our log normal distribution with the Excel LogNorm.inv command and we see that we have the same percentiles and summary statistics given the parameters mu and sigma. Our distirbution is the same as the Excel built in function for LogN[mu,sigma].
6. Working in reverse now, take the natural log (LN) of the lognormal distribution and we are back to the N[mu,sigma] distribution.
7. Subtract mu and divide by sigma and we return to the standard normal distribution, N[0,1].

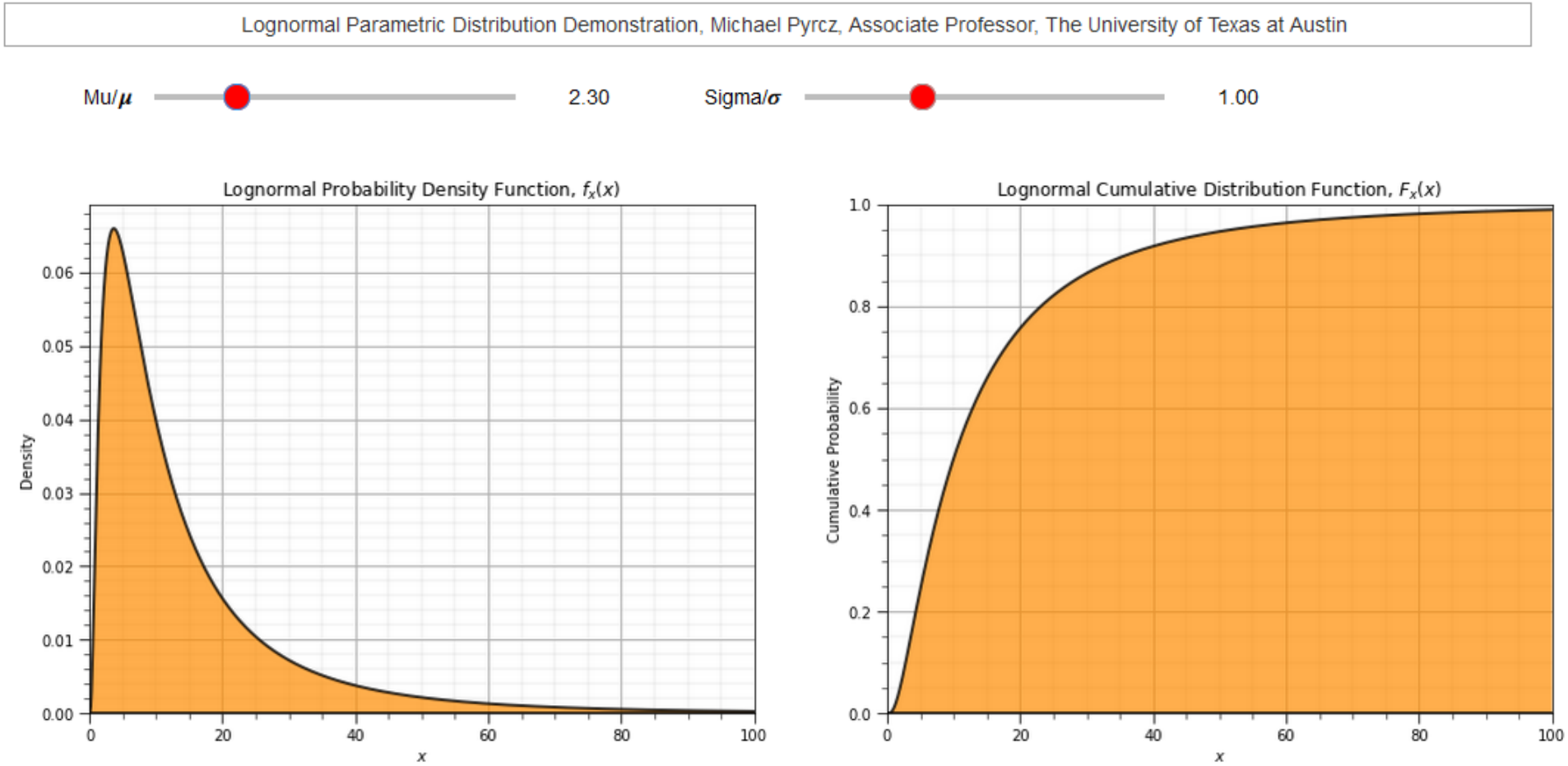
**What did we learn?**

1. We can build a lognormal distribution by exponentiation of random variable X, if X is normally distributed.
2. We can build a normal distribution by taking the natural log of Y if Y is lognormally distributed.
3. mu and sigma lognormal parameters are the mean and standard deviation for the normal distribution exponentiated to become lognormal.
   If X is N[mu,sigma] then Y = EXP(X), Y is LogNormal[mu,sigma], and it follows that X = LN(Y), X is once again N[mu,sigma]
4. It is possible to calculate the mean and standard deviation of a lognormal distribution given its parameters mu and sigma.

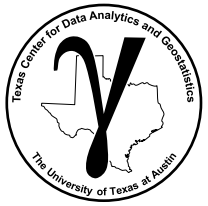
Going from Normal to Lognormal and back, calculating mean, variance, mu and sigma. Example at Files/Excel/Lognormal_Distribution_Demo.xlsx GitHub/GeostatsGuy https://git.io/fNgBB

# Gaussian Distribution Demonstration in Python

Try out the lognormal parametric distribution by changing the parameters and observing the PDF and CDF.




Lognormal parametric distribution PDF and CDF in file Interactive_ParametricDistributions.ipynb.




# Other Parametric Statistical Distributions

**We Will Use These for Confidence Intervals and Hypothesis Testing**

- Student's t
- Chi-square
- Fisher's F

# Student's t Distribution

**Student's t Distribution:**

PDF:

$$f(x) = \frac{\Gamma\left(\frac{\Phi+1}{2}\right)}{\Gamma\left(\frac{\Phi}{2}\right)} \frac{1}{\sqrt{\Phi\pi}} \frac{1}{\left(1+\frac{x^2}{\Phi}\right)^{\frac{\Phi+1}{2}}}$$

**Φ** = degrees of freedom (univariate) = n-1

Gamma Function

$$\Gamma(x) = \lim_{n\to\infty} \frac{n!\, n^{x-1}}{x(x+1)(x+2)\ldots(X+n-1)}$$

William Gosset's (1876-1937) pseudonym was Student

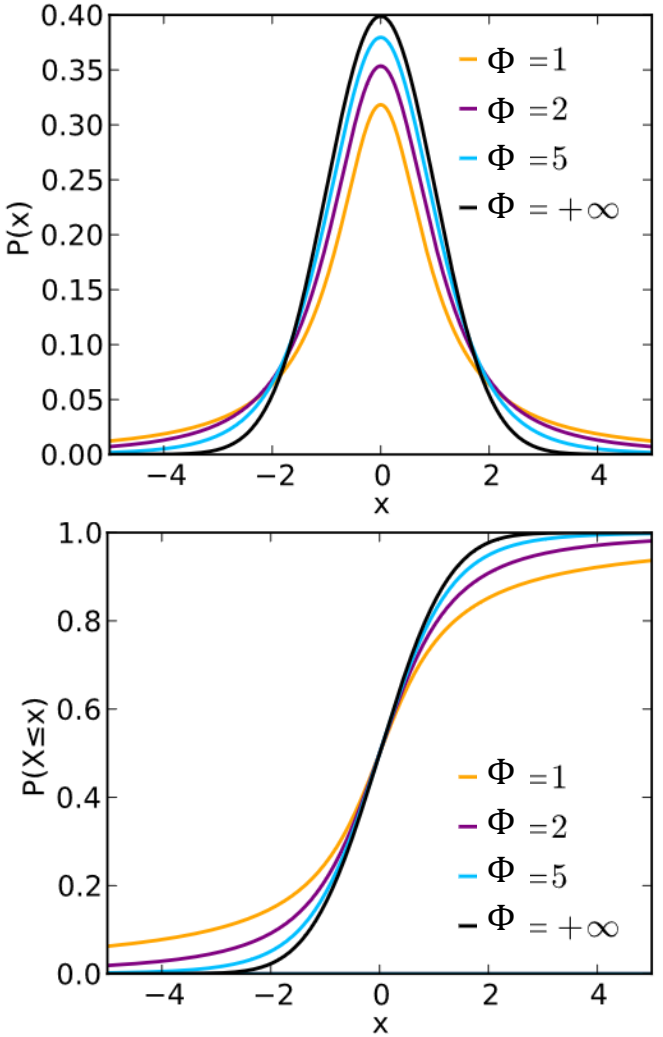
Mean of $X_1, X_1, \ldots, X_n$ independent, identically distribution as N(μ,?).

"Take $n$ samples from a Gaussian distribution with unknown standard deviation. Calculate the average. Repeat. The means have a Student's t distribution."

Note as $\Phi \to \infty$ , as the sample size is large, student's t distribution approaches Gaussian distribution.

We use student's t when we have too few samples and population standard deviation is unknown.

Used in testing difference between means from 2 sample sets.




Example student's t PDFs and CDFs with variable degrees of freedom.

Image from https://en.wikipedia.org/wiki/Student%27s_t-distribution

# $\chi^2$ or Chi-Square Distribution

## $\chi^2$ Distribution:

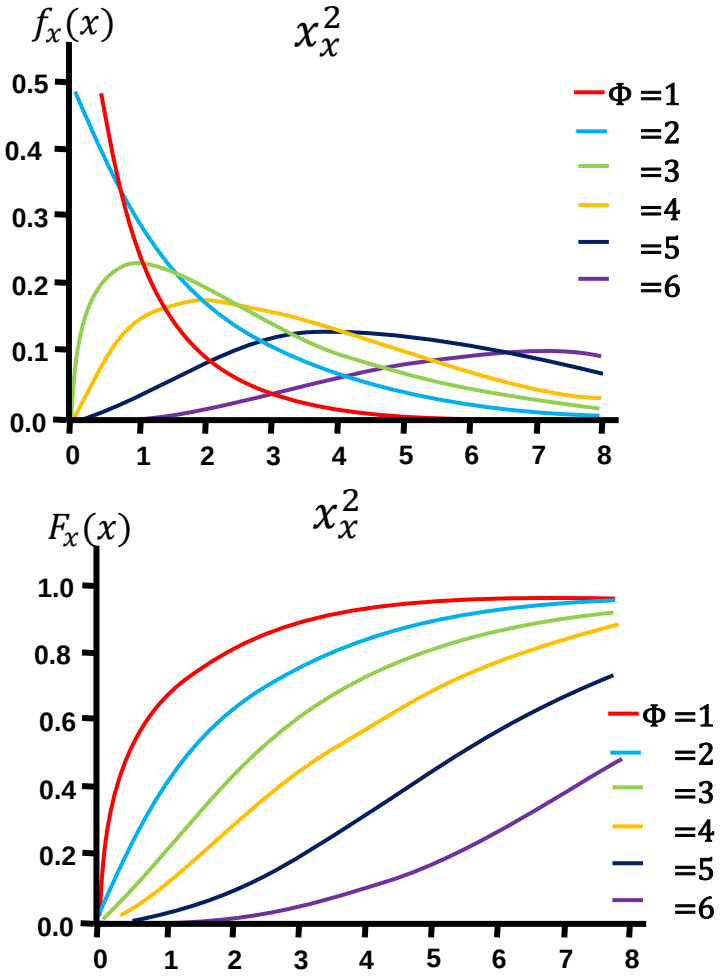
PDF: $$f(x) = \frac{1}{2^{\frac{\Phi}{2}}\Gamma\left(\frac{\Phi}{2}\right)} x^{\frac{\Phi-2}{2}} e^{-\frac{x}{2}}, \quad x > 0$$

degrees of freedom (univariate) = n-1

Let $X_1, X_2, \ldots, X_n$ be independent identically distributed as N(μ,$\sigma^2$). The distribution of the sum of squares with have a chi-square distribution.

$$\sum_{\alpha=1}^{n} (X_\alpha)^2 \rightarrow \chi^2$$

Applied in a lot of hypothesis testing (next Lecture). E.g., to compare, test difference between 2 CDFs.




Example $\chi^2$ PDFs and CDFs with variable degrees of freedom.

Image from https://en.wikipedia.org/wiki/Chi-squared_distribution

# Fisher's F Distribution

**F Distribution:**

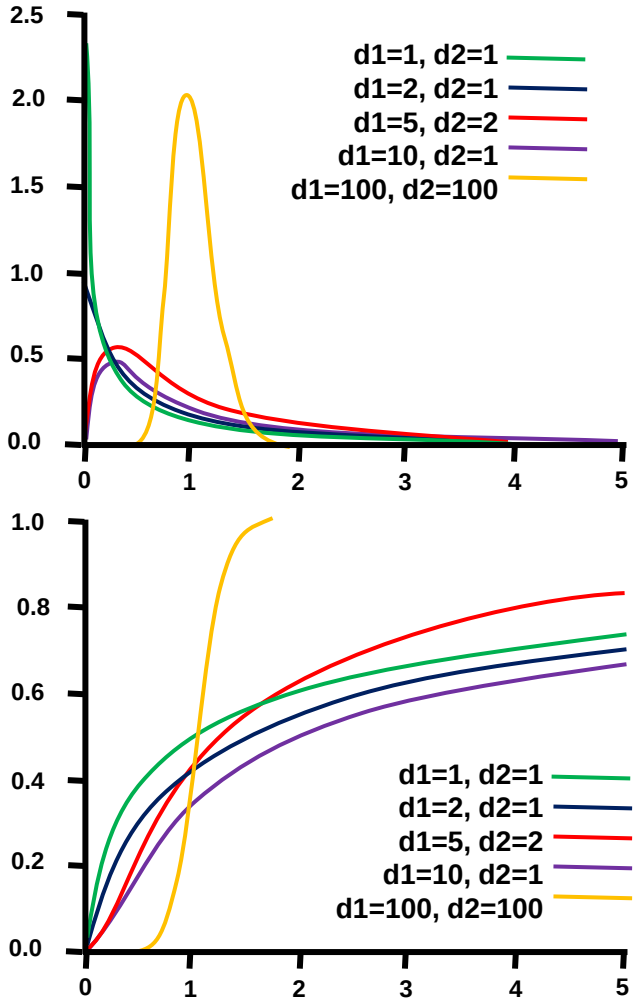
PDF: $$f(x) = \frac{\Gamma\left(\frac{\Phi_1 + \Phi_2}{2}\right)\left(\frac{\Phi_1}{\Phi_2}\right)^{\frac{\Phi_1}{2}}}{\Gamma\left(\frac{\Phi_1}{2}\right)\Gamma\left(\frac{\Phi_2}{2}\right)} x^{\frac{\Phi_1 - 2}{2}} \left(1 + \frac{\Phi_1}{\Phi_2} x\right)^{-\left(\frac{\Phi_1 + \Phi_2}{2}\right)}$$

$\boldsymbol{\Phi_1}$, $\boldsymbol{\Phi_2}$ are degrees of freedom (univariate) = n-1

$\boldsymbol{\Phi_1}$ and $\boldsymbol{\Phi_2}$ are the degrees of freedom of the two distributions being compared.

From ratio of two scaled chi-square distributed random variables.

Distribution used in analysis of variance, test difference in variance between 2 sample sets.




Example F PDFs and CDFs with variable degrees of freedom.

Image from https://en.wikipedia.org/wiki/Chi-squared_distribution

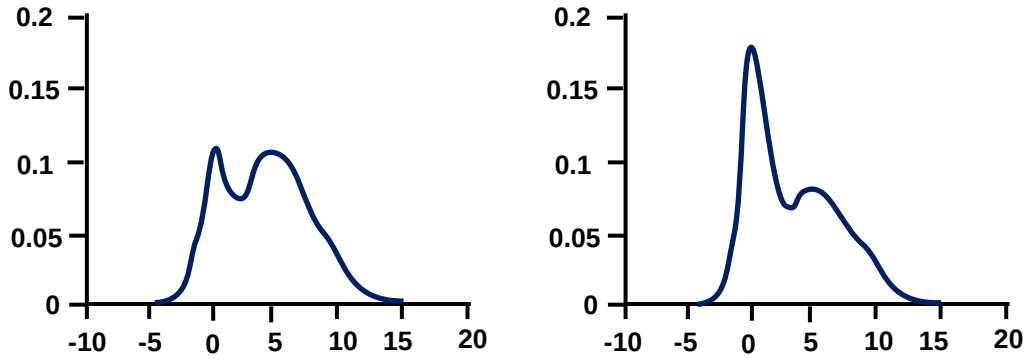
# Linear Family of Distribution Models

$$F(x) = \sum_k \lambda_k F_k(x)$$

- A positive linear combination of licit distributions is a licit distribution.

$$\sum_k \lambda_k = 1, \quad \text{and} \quad \lambda_k \geq 0, \forall k$$

- For example, $F(x) = p\Delta_0(x) + (1-p)F_1(x)$ is a mixture of a spike at zero and a positive distribution



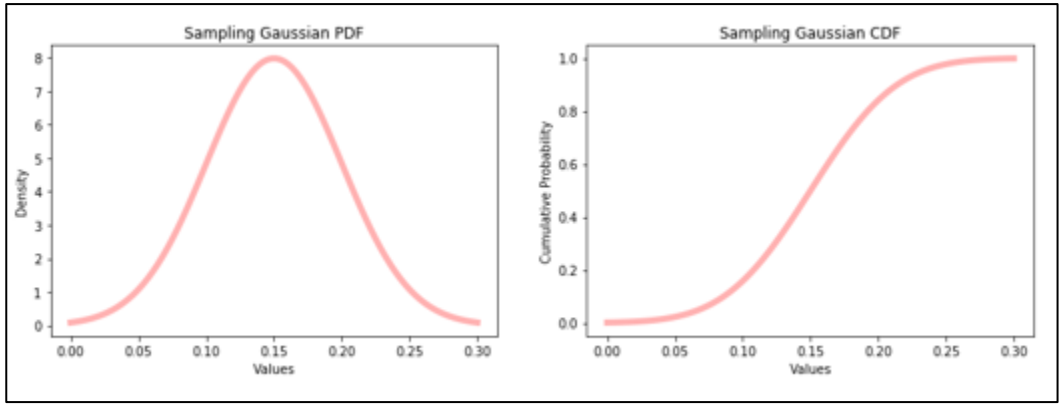

Example linear combinations of PDFs.

- Commonly observed with the mixing of populations in sample sets.

# Parametric Distributions in Python

## Walk Through in Python

Parametric distributions in Python demo




### Data Analytics

#### Parametric Distributions in Python

**Michael Pyrcz, Associate Professor, University of Texas at Austin**

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

#### Data Analytics: Parametric Distributions

Here's a demonstration of making and general use of parametric distributions in Python. This demonstration is part of the resources that I include for my courses in Spatial / Subsurface Data Analytics at the Cockrell School of Engineering at the University of Texas at Austin.

**Parametric Distributions**

We will cover the following distributions:

- Uniform
- Triangular
- Gaussian
- Log Normal

We will demonstrate:

- distribution parameters
- forward and inverse operators
- summary statistics

I have a lecture on these parametric distributions available on YouTube.

**Getting Started**

Here's the steps to get setup in Python with the GeostatsPy package:

1. Install Anaconda 3 on your machine (https://www.anaconda.com/download/).
2. From Anaconda Navigator (within Anaconda3 group), go to the environment tab, click on base (root) green arrow and open a terminal.
3. In the terminal type: pip install geostatspy.
4. Open Jupyter and in the top block get started by copy and pasting the code block below from this Jupyter Notebook to start using the geostatspy functionality.

You will need to copy the data file to your working directory. They are available here:

Parametric distribution demonstration in Python file PythonDataBasics_ParametricDistributions.ipynb.

# PGE 338 Data Analytics and Geostatistics

## Lecture 5: Univariate Distributions

**Lecture outline . . .**

- **Nonparametric Distributions**

**Michael Pyrcz, The University of Texas at Austin**


- Introduction
- General Concepts
- Univariate
  - PDF / CDF
  - Statistics
  - Distributions
  - Heterogeneity
  - Hypothesis
- Bivariate
- Time Series Analysis
- Spatial Analysis
- Machine Learning
- Uncertainty Analysis

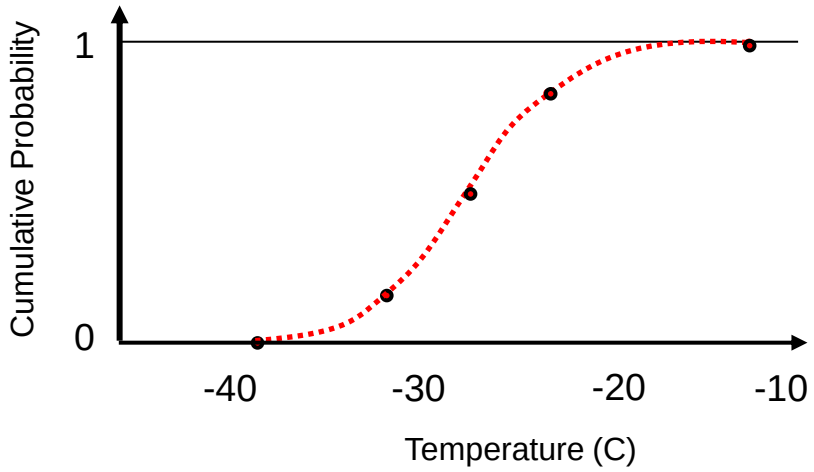
# General Nonparametric

## Working with Nonparametric Distributions

- We can calculate the distribution directly from the data, as demonstrated in:

**Topic 03 – Univariate PDF and CDF.**

- Later we will demonstration distribution transforms, transforming any distribution to any other distribution (see later)




$$F_x(x) = Prob\{X \leq x\}$$

Jan 16th – daytime high, Edmonton, Ab, Canada (degrees Celsius)

# PGE 338 Data Analytics and Geostatistics

## Lecture 5: Univariate Distributions

**Lecture outline . . .**

- **Monte Carlo Simulation**

**Michael Pyrcz, The University of Texas at Austin**


- Introduction
- General Concepts
- Univariate
  - PDF / CDF
  - Statistics
  - Distributions
  - Heterogeneity
  - Hypothesis
- Bivariate
- Time Series Analysis
- Spatial Analysis
- Machine Learning
- Uncertainty Analysis

# Monte Carlo Simulation

**Random sampling from a distribution**

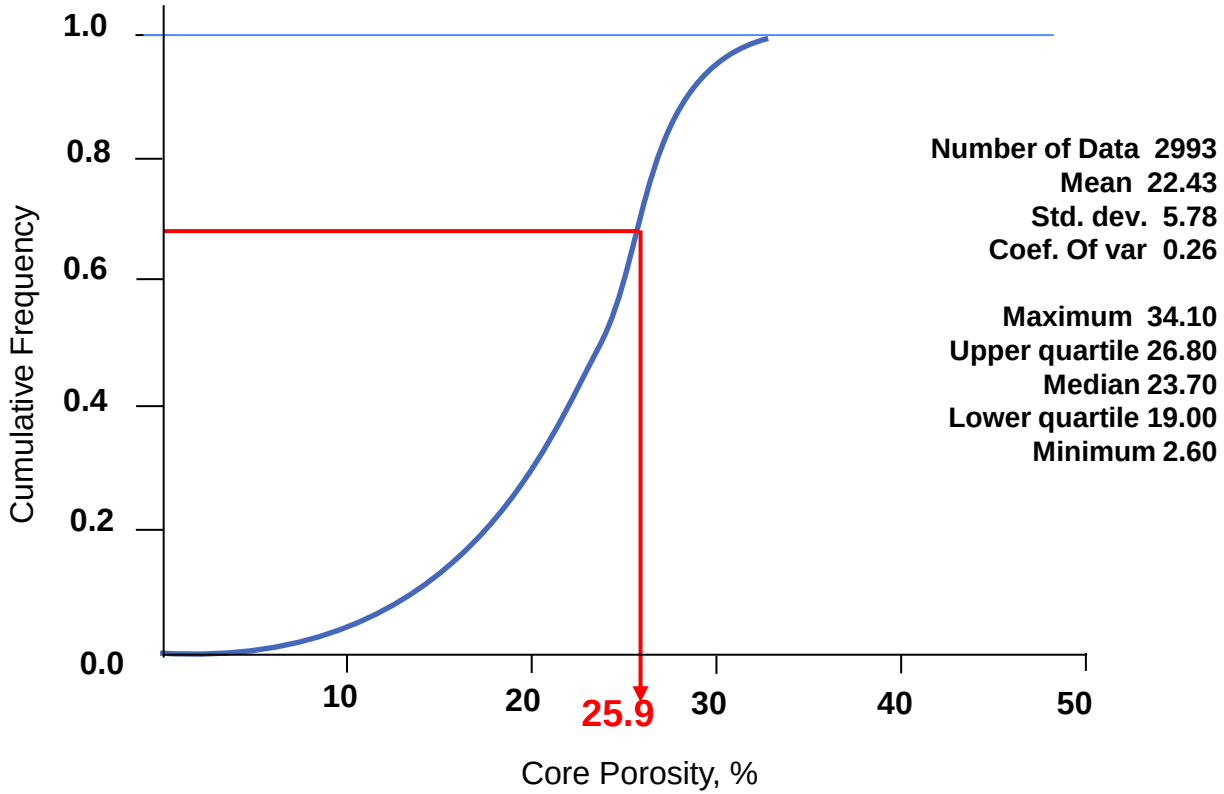
Workflow Steps:

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

$$x^l = F_x^{-1}(p^l)$$

4. Repeat to calculate enough realizations for analysis.

The method is very powerful. You can simulate distributions that could not be calculated analytically.




CDF of 2,993 porosity samples with a single Monte Carlo Simulation.

# Monte Carlo Simulation Workflow

**Propagating uncertainty through a transfer function**

Workflow Steps:

1. Model all the inputs distributions

$$f_{x_1}(x_1) \quad f_{x_2}(x_2) \quad f_{x_3}(x_3)$$

2. Monte Carlo simulate a realizations for all the inputs

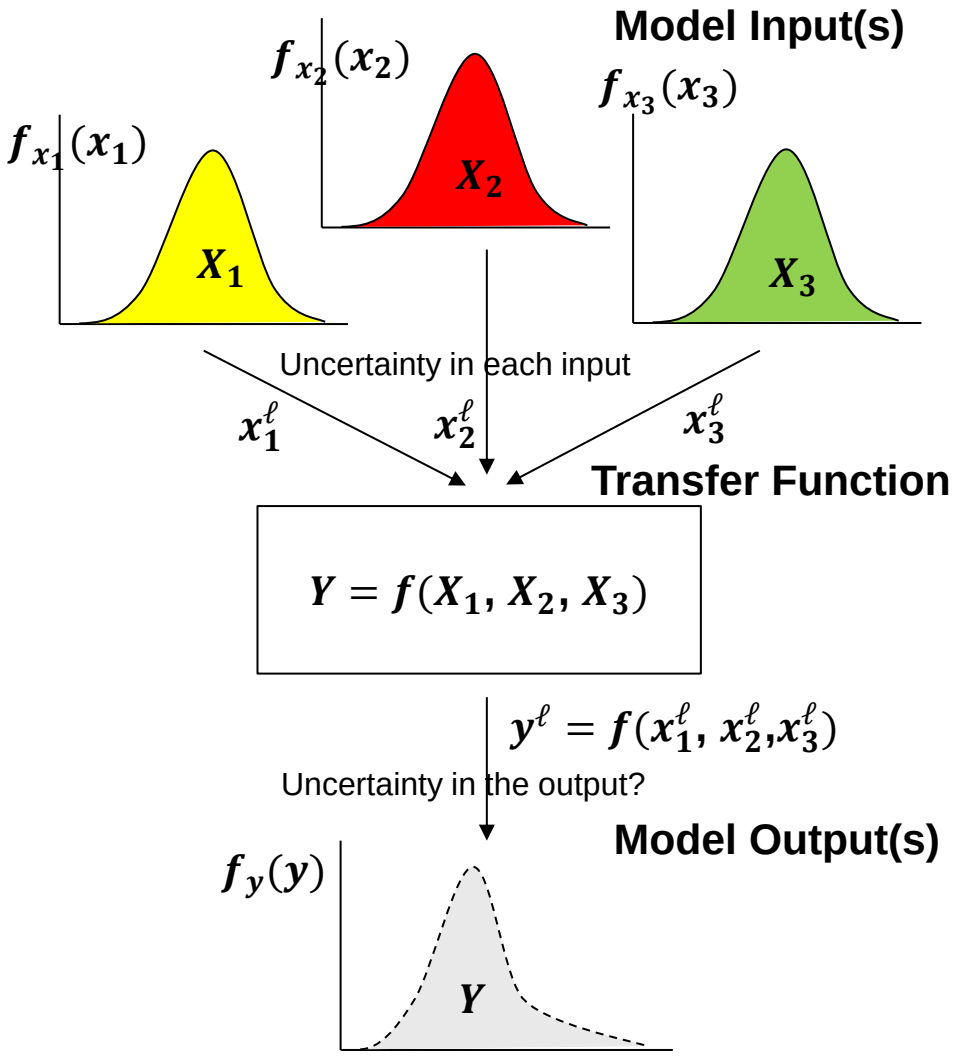
$$x_1^\ell,\ x_2^\ell, x_3^\ell$$

3. Apply to the transfer function to get a realization of the model output

$$y^\ell = f(x_1^\ell,\ x_2^\ell, x_3^\ell)$$

4. Repeat to calculate enough realizations to model the response feature distribution.

$$f_y(y)$$

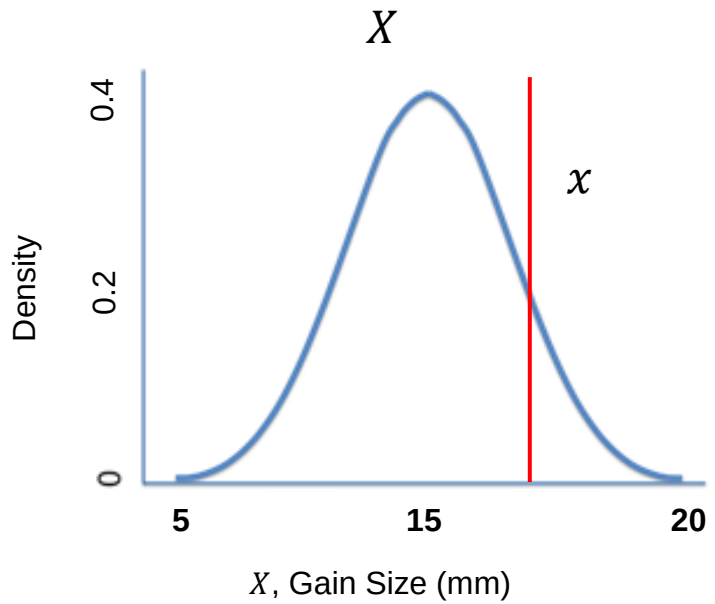



Monte Carlo simulation workflow to propagate uncertainty.

# Recall: Random Variable (RV) Definition

**Recall: Random Variable**

- we do not know the value at a location / time, **it can take on a range of possible values**, fully described with a PDF.
- represented as an **upper-case variable, e.g., $X$, while possible outcomes or data measures are represented with lower case, e.g., $x$.**
- in spatial context common to use a location vector, $\mathbf{u}$, to describe a location, e.g., $x(\mathbf{u})$, $X(\mathbf{u})$

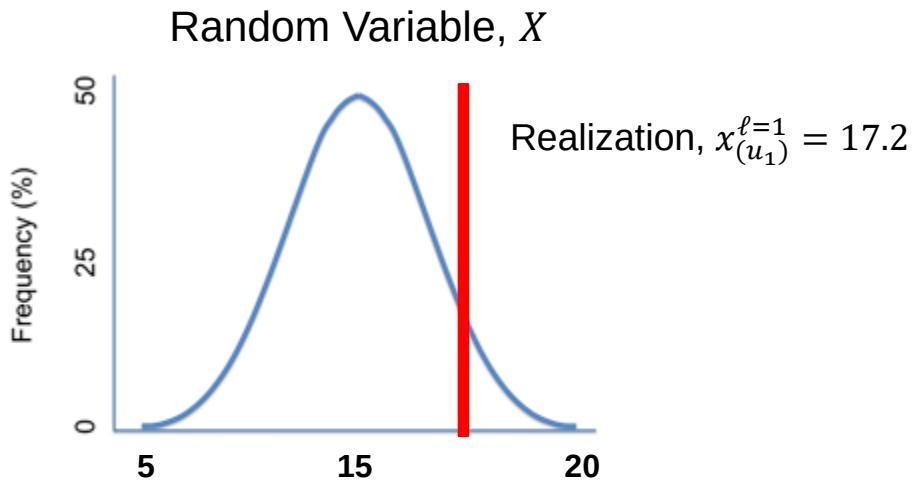



Random variable, grain size uncertain at a location represented with a PDF.

# Realization Definition

## Realization

- an outcome from a random variable (or joint set of outcomes from a random function – we will cover this later)
- represented with lower case, e.g., $x$.
- in spatial context common to use a location vector, $\mathbf{u}$, to describe a location, e.g. $x(\mathbf{u})$, $X(\mathbf{u})$
- resulting from simulation, e.g., Monte Carlo simulation, sequential Gaussian simulation ← a method to sample (jointly) from the RV (RF)
- each realization is considered equiprobable




Random variable, grain size uncertain at a location represented with a PDF and a single realization.

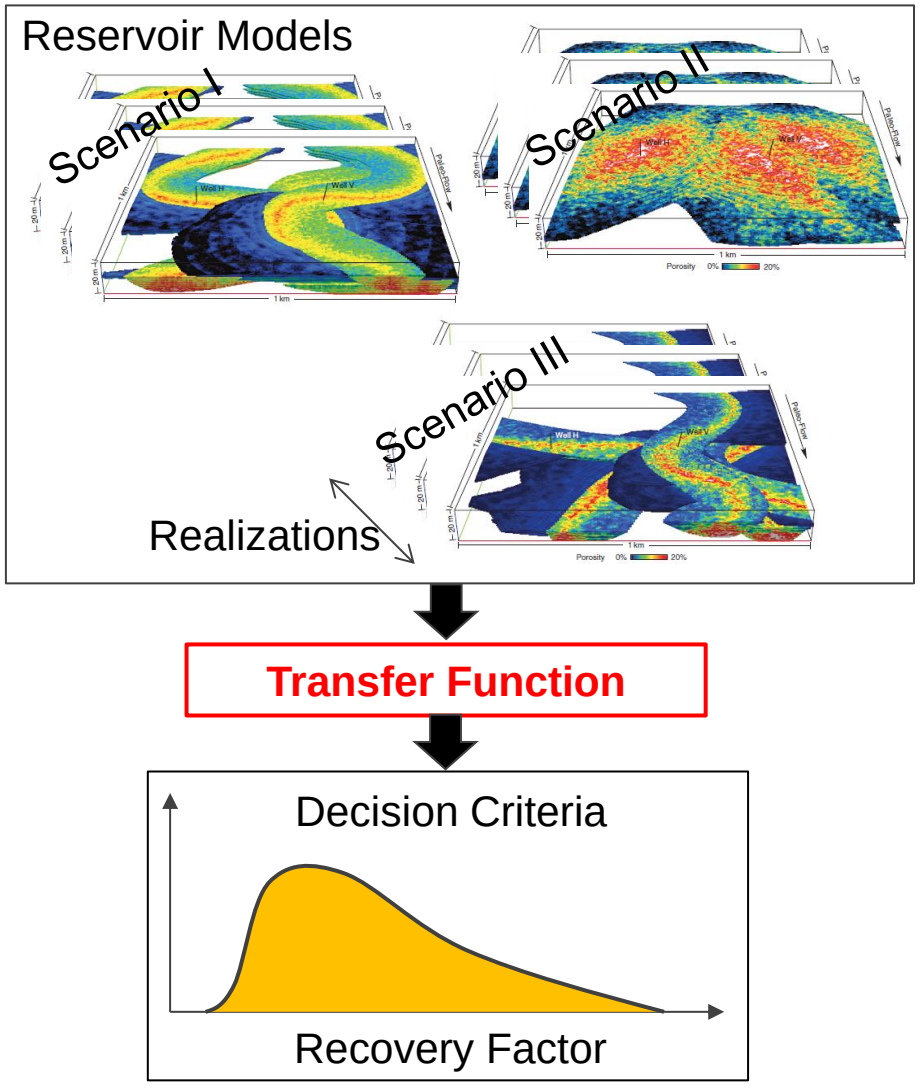
# Recall: Transfer Function

**Transfer Function**

Calculation applied to the model (deterministic or statistical) to calculate a decision criterion

Examples:

- transport and bioattenuation of a soil contaminant
- volumetric calculation for oil-in-place
- heterogeneity metric for estimation of recovery factor
- flow simulation for production forecast
- mine plan and rock homogenization for mineral resources




The standard reservoir modeling workflow.

# Monte Carlo Simulation Motivation

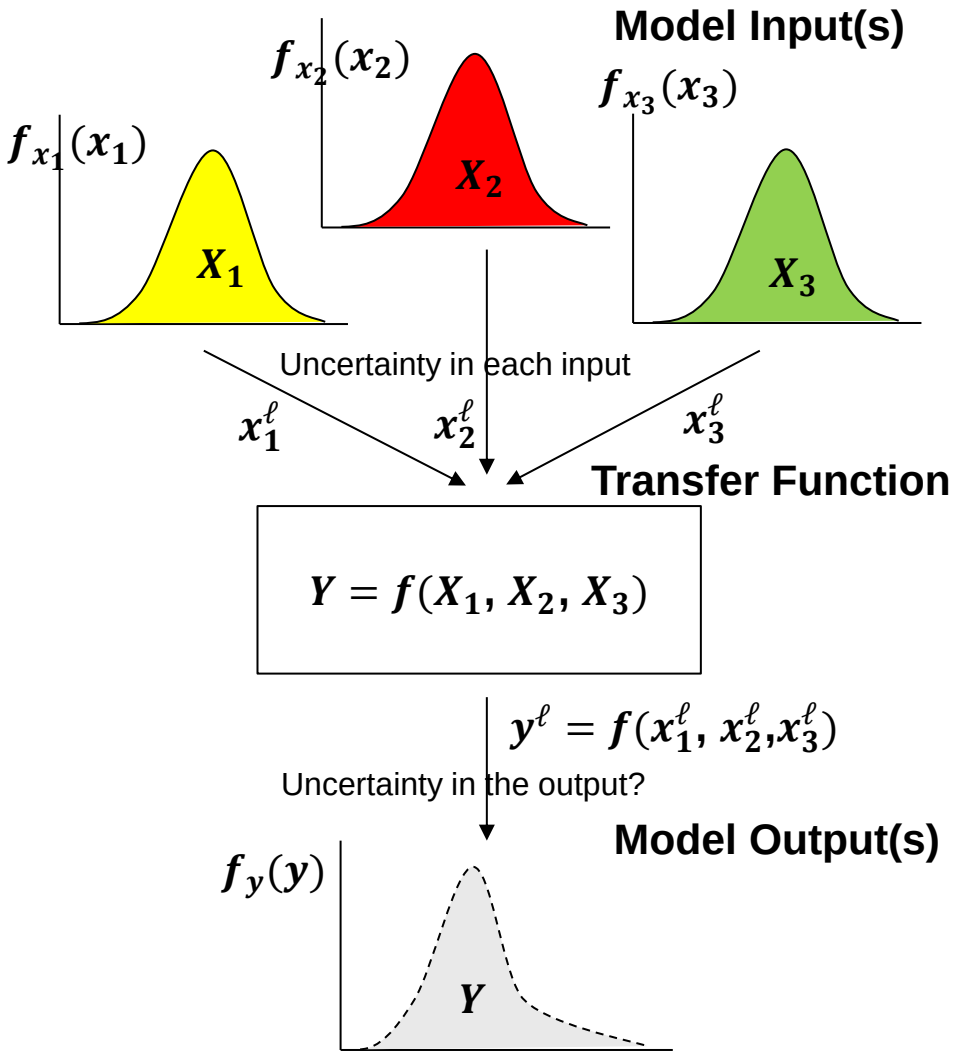
**Very Simple and Flexible**

- Method for propagating the distribution of uncertainty through a calculation.

- Volumetrics
- Flow Forecasts
- Economics

- We can only do this analytically for simple cases.

**We need Monte Carlo Simulation to build practical uncertainty models**




Monte Carlo simulation workflow to propagate uncertainty.

# Monte Carlo Simulation Motivation

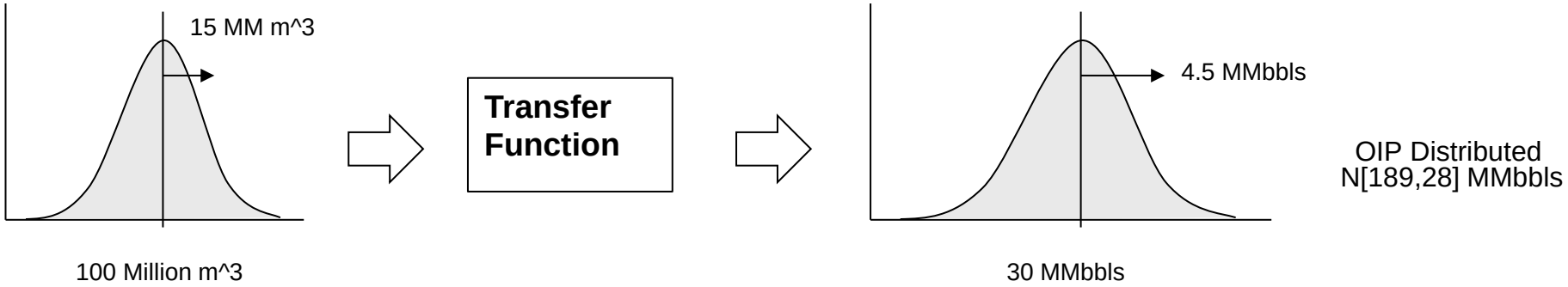
## Problem 1:

- The reservoir oil in place ($OIP$) uncertainty is Gaussian distributed with:
  - mean of 100 million barrels and standard deviation of 15 million barrels.

The recovery factor ($RF$) is estimated at 0.3, 30% of the oil can technically be extracted.

- **Calculate the uncertainty in the recoverable oil ($RO$):**

$$\boldsymbol{RO = RF \times OIP}$$




With statistical expectation we can calculate the $\mu_{cX}$ and $\sigma_{cX}$.
Scaling by a constant won't change the distribution shape.

$$c \cdot X, c = 0.3$$
$$\mu_{cX} = E[c \cdot X] = c \cdot E[X]$$
$$\sigma_{cX} = St.Dev.[c \cdot X] = c \cdot SD[X]$$

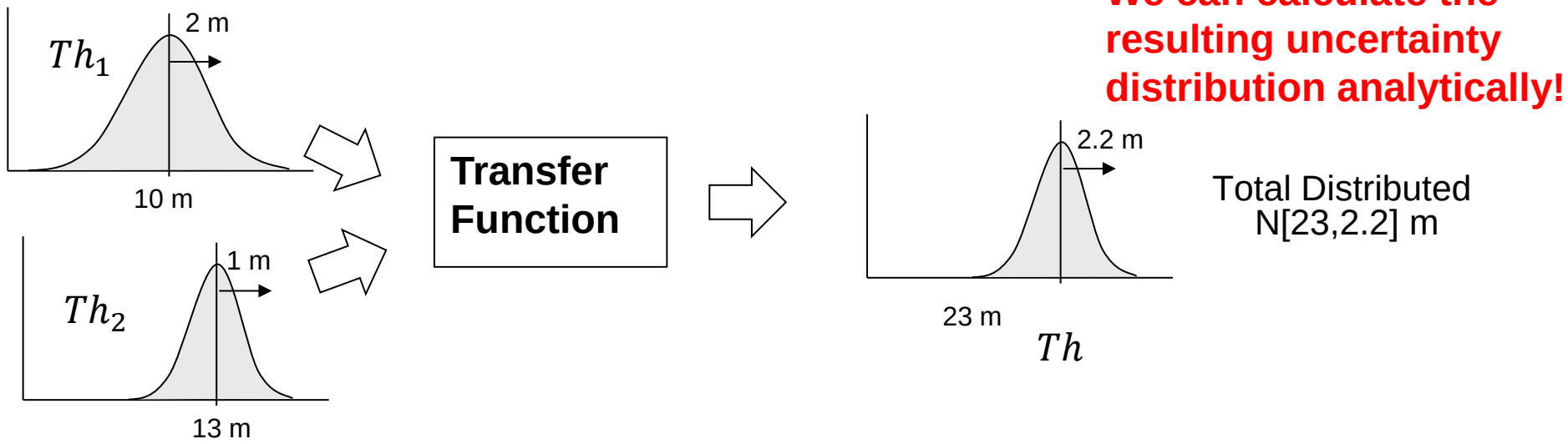
**We can calculate the resulting uncertainty distribution analytically!**

# Monte Carlo Simulation Motivation

## Problem 2:

- The reservoir is comprised of 2 units with thickness uncertainty for each:
  - Unit 1 N[10,2] m (recall, Gaussian with mean = 10m and standard deviation = 2)
  - Unit 2 N[13,1] m
- **Calculate the uncertainty in the total thickness ($Th$)**

$$Th = Th_1 + Th_2$$




**We can calculate the resulting uncertainty distribution analytically!**

Total Distributed N[23,2.2] m

With expectation we can calculate $\mu_{X_1+X_2}$ and $\sigma_{X_1+X_2}$ if $X_1$ and $X_2$ are independent.
Adding two Gaussian distributions results in a Gaussian distribution.

$$E[X_1 + X_2] = E[X_1] + E[X_2]$$

$$Var[X_1 + X_2] = Var[X_1] + Var[X_2] \quad \textbf{if } X_1 \perp\!\!\!\perp X_2$$

(independent random variables)

# Monte Carlo Simulation Motivation

## Problem 3:

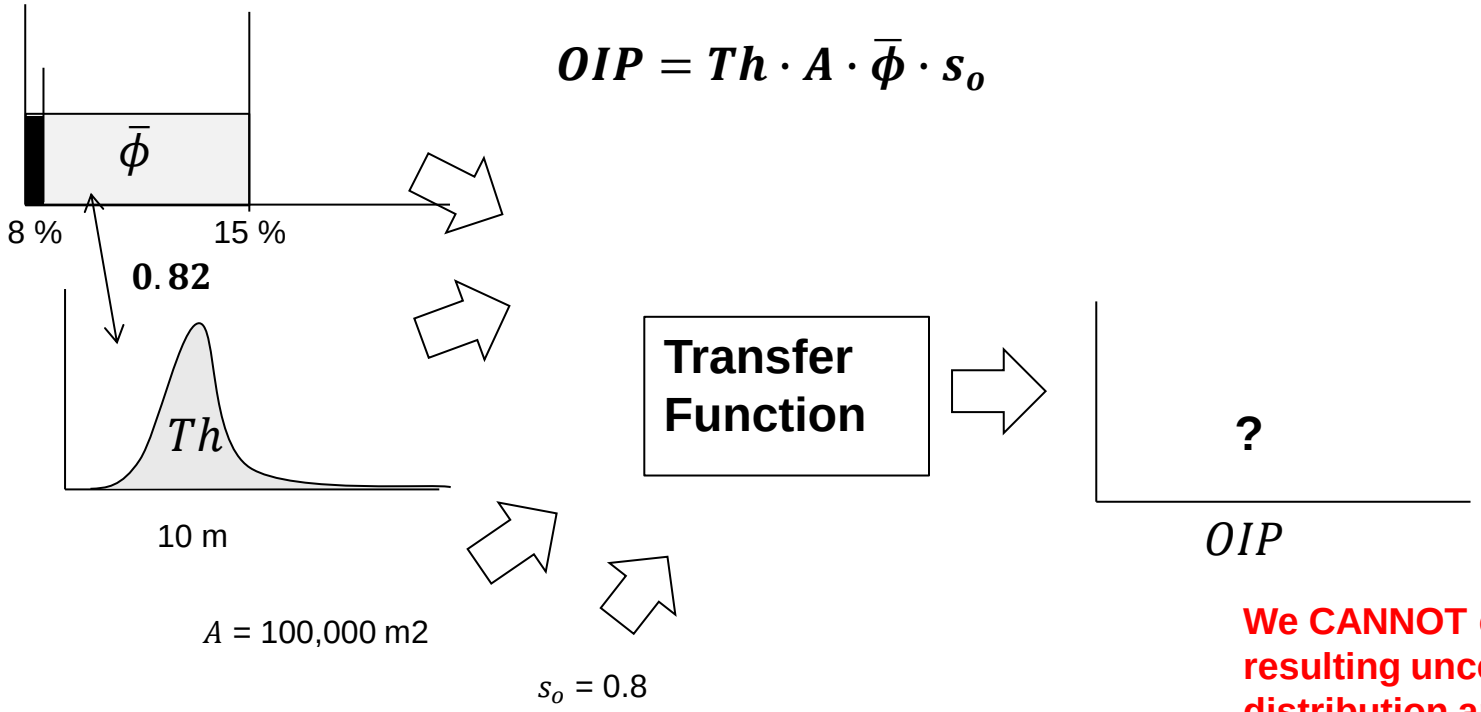
- The reservoir has the following features with uncertainty:
  - Average Porosity ($\bar{\phi}$) ~ U[8,15] %
  - Thickness ($Th$) ~ LogN[1,1] m
  - Area ($A$) = 100,000 m^2 and Oil Saturation ($s_o$) = 0.8
- **Calculate the uncertainty in the oil in place (OIP):**

**Correlation between features**

$$\rho_{\bar{\phi},thick} = 0.82$$

**Threshold on a feature**

$$if\ \bar{\phi} < 9\%\ \text{then}\ \bar{\phi} = 0$$




$$\boldsymbol{OIP = Th \cdot A \cdot \bar{\phi} \cdot s_o}$$

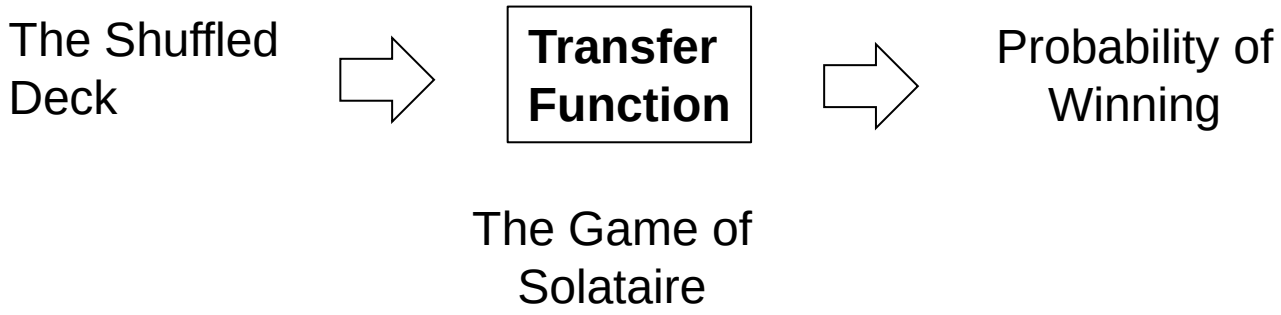
**We CANNOT calculate the resulting uncertainty distribution analytically!**

# Monte Carlo Simulation Motivation

## Problem 4

- Solitaire Game – probability of winning:
  - » Large combinatorial of possible outcomes with each hand and cards on the table
  - » Card ordering matters
  - » Correlation / constraints imposed on hands from cards played
  - » Strategy / Discrete choices

**Calculate the uncertainty distribution, probability of winning:**




This is not trivial, many practical problems have large combinatorials, complicated correlations, ordering, correlation, constraints, discrete choices.

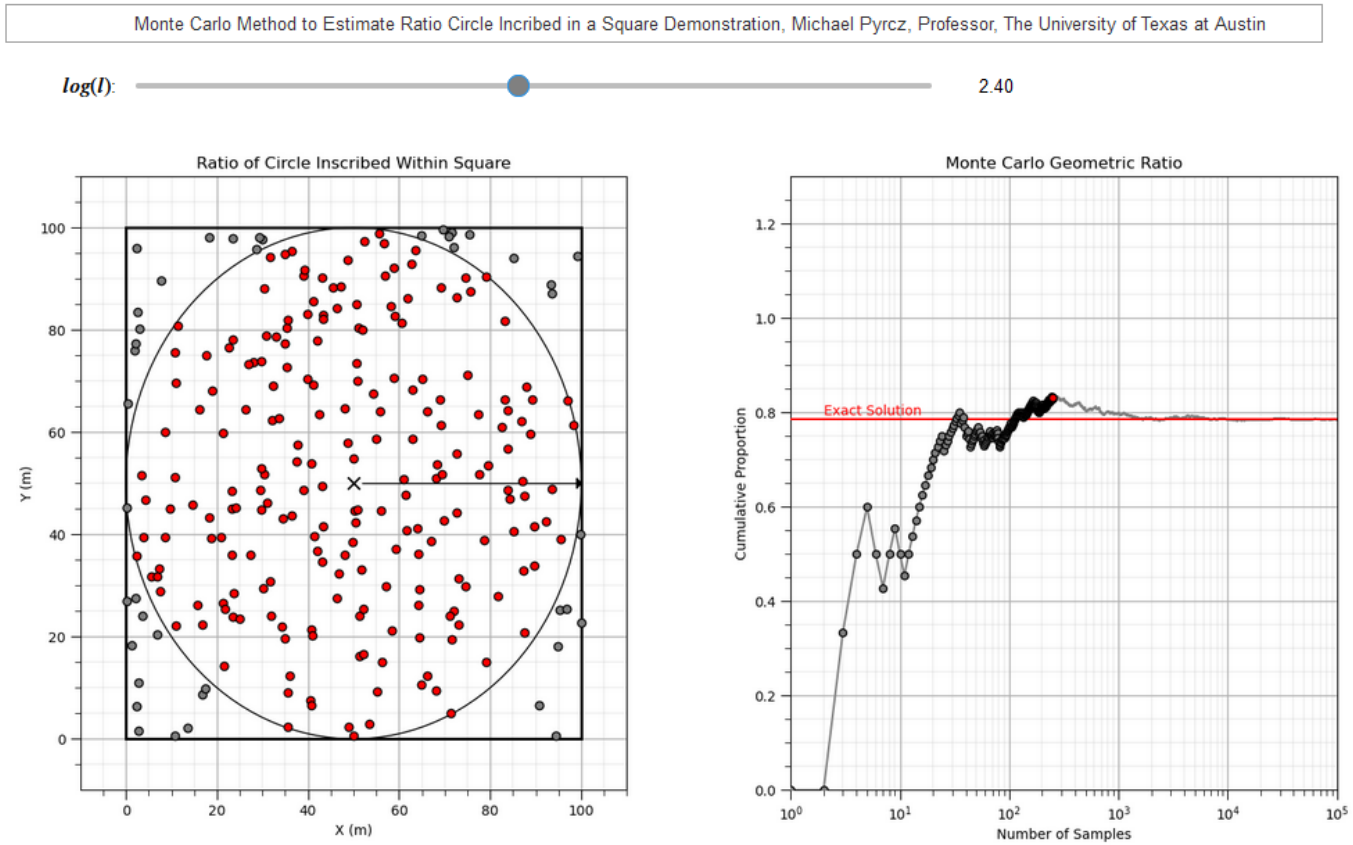
We CANNOT calculate this analytically. Why not just get the computer to play enough solitaire and observe / count the outcomes?

**This is the idea of Monte Carlo Simulation.**

# Monte Carlo Methods

Let's pause and define the general Monte Carlo method:

- solving otherwise intractable or computationally expensive problems, e.g., numerical integration or optimization, by random sampling (drawing realizations).




Interactive Monte Carlo Methods in Python demo in file: Interactive_Monte_Carlo_Methods.ipynb

# Monte Carlo Simulation Uncertainty Modeling Workflow

**The Common Steps for Monte Carlo Simulation:**

1. **Specify the model** / equation, $f$, and all required independent variables / predictor features, $X$, and dependent variable(s) / response feature(s), $Y$, $Y = f(X_1, \ldots, X_m)$

2. Specify the **uncertainty distribution for each predictor feature**, $X_1, \ldots, X_m$

3. Perform **Monte Carlo simulation from each distribution** to draw a realization from all $1, \ldots, m$ predictor feature distributions to get one realization of all predictor features, $x_1^\ell, \ldots, x_m^\ell$

4. Apply the realization of predictor features to the model to get a **realization of the output**, dependent variable, response feature, $Y^\ell = f(x_1^\ell, \ldots, x_m^\ell)$

5. **Repeat for $\ell = 1, \ldots, L$ realizations** to sufficiently sample the distribution of the response feature, this is your uncertainty model.

# Monte Carlo Simulation Exercise in Excel

**DIY Monte Carlo Simulation in Excel. How would you do it?**

- Each row is an independent Monte Carlo sample of each random variable.
- Each column is a random variable.

$F_x^{-1}(P) = x_p$

$F_x^{-1}(P) = x_p$

$F_x^{-1}(P)$ Gaussian Distribution:
NORM.INV(RAND(),mean,st.dev.)

**Transfer Function**

| Monte Carlo Simulation | | | |
|---|---|---|---|
| | A | B | A+B |
| 1 | 12.67 | 19.14 | 31.81 |
| 2 | 11.31 | 23.29 | 34.60 |
| 3 | 9.25 | 21.79 | 31.04 |
| 4 | 9.59 | 19.32 | 28.90 |
| 5 | 9.21 | 12.06 | 21.27 |
| 6 | 10.83 | 20.78 | 31.61 |
| 7 | 8.07 | 19.17 | 27.24 |
| 8 | 11.21 | 29.77 | 40.98 |
| 9 | 9.16 | 23.32 | 32.48 |
| 10 | 13.97 | 24.32 | 38.29 |
| 11 | 8.67 | 20.35 | 29.02 |
| 12 | 12.08 | 20.78 | 32.87 |
| 13 | 13.30 | 16.36 | 29.66 |
| 14 | 11.72 | 19.55 | 31.28 |
| 15 | 9.56 | 19.85 | 29.41 |
| 16 | 9.53 | 24.58 | 34.11 |

# Monte Carlo Simulation Exercise in Excel

You project is about to drill into two formations.

- The distribution of thickness is N[10.0,2.0] meters for unit A and N[24.0,4.0] meters for unit B.
- The thicknesses of the units are independent.  (Hint: use NORM.INV() and RAND() functions).

What is the mean and variance of the total thickness of reservoir units that your asset team should forecast for production planning / facilities?

What is the P10 low side and the P90 high side?

# Monte Carlo Simulation Exercise in Excel

You project is about to drill into two formations.

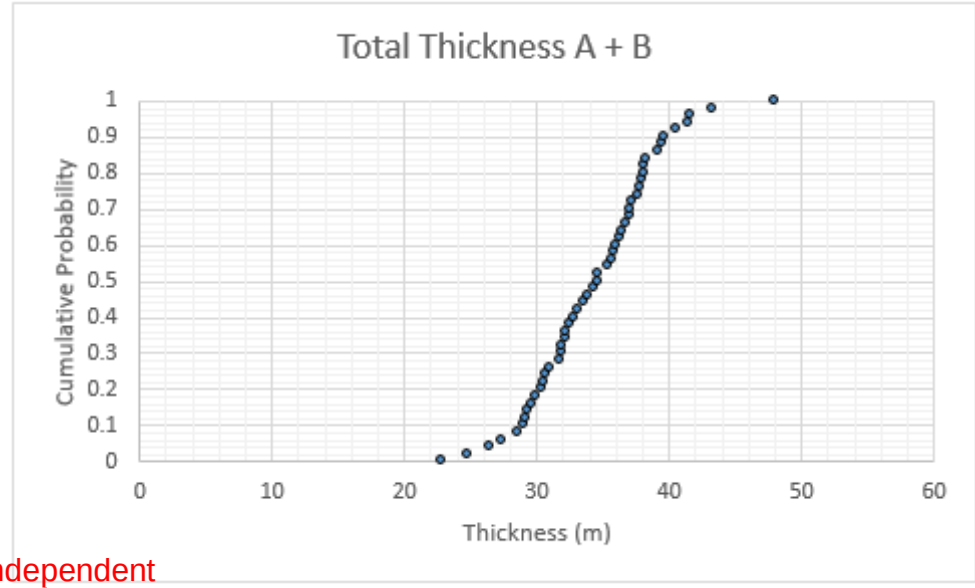
- The distribution of thickness is N[10.0,2.0] meters for unit A and N[24.0,4.0] meters for unit B.
- The thicknesses of the units are independent. (Hint: use NORM.INV() and RAND() functions).

What is the mean and variance of the total thickness of reservoir units that your asset team should forecast for production planning / facilities?

Mean ≈ 34
Variance ≈ 21

What is the P10 low side and the P90 high side?

P10 ≈ 28
P90 ≈ 40




CDF of Monte Carlo simulations of total thickness.

Note: E[A + B] = E[A] + E[B] and Var[A + B] = Var[A] + Var[B] + 2 x Covariance(A,B)

= 0 since independent

# Monte Carlo Simulation Exercise in Excel

**Monte Carlo Simulation Demo, Michael Pyrcz, The University of Texas at Austin, @GeostatsGuy**

Here's a very simple Monte Carlo simulation demonstration for the case of prediction of total reservoir thickness given 2 reservoir units A and B, each with uncertain thickness.
Note, both have thickness random variables represented as Gaussian distributions. The transfer function is: $th_{total} = th_A + th_B$

| | Unit A Thickness RV (m) | Unit A Thickness RV (m) |
|---|---|---|
| mean | 10.0 | 24.0 |
| st. dev. | 2.0 | 4.0 |

**Monte Carlo Simulation Statistics**

| | |
|---|---|
| Mean | 33.38 |
| St Dev | 4.13 |

**Summarize and Model**

**Unit A** $th_A$

| Realizations $\ell$ | Random Cumulative Probability U[0,1] $p^\ell$ | Realization Thickness of Unit A $th_A^\ell = F_{th_A}^{-1}(p^\ell)$ |
|---|---|---|
| 1 | 0.14 | 7.83 |
| 2 | 0.10 | 7.47 |
| 3 | 0.35 | 9.22 |
| 4 | 0.55 | 10.27 |
| 5 | 0.89 | 12.47 |
| 6 | 0.49 | 9.93 |
| 7 | 0.32 | 9.04 |
| 8 | 0.61 | 10.55 |
| 9 | 0.62 | 10.60 |
| 10 | 0.06 | 6.88 |
| 11 | 0.82 | 11.85 |
| 12 | 0.52 | 10.11 |

**Unit B** $th_B$

| Realizations $\ell$ | Random Cumulative Probability U[0,1] $p^\ell$ | Realization Thickness of Unit B (m) $th_B^\ell = F_{th_B}^{-1}(p^\ell)$ |
|---|---|---|
| 1 | 0.76 | 26.88 |
| 2 | 0.50 | 23.99 |
| 3 | 0.71 | 26.20 |
| 4 | 0.15 | 19.87 |
| 5 | 0.20 | 20.62 |
| 6 | 0.71 | 26.17 |
| 7 | 0.39 | 22.91 |
| 8 | 0.21 | 20.84 |
| 9 | 0.41 | 23.09 |
| 10 | 0.80 | 27.36 |
| 11 | 0.36 | 22.60 |
| 12 | 0.15 | 19.87 |

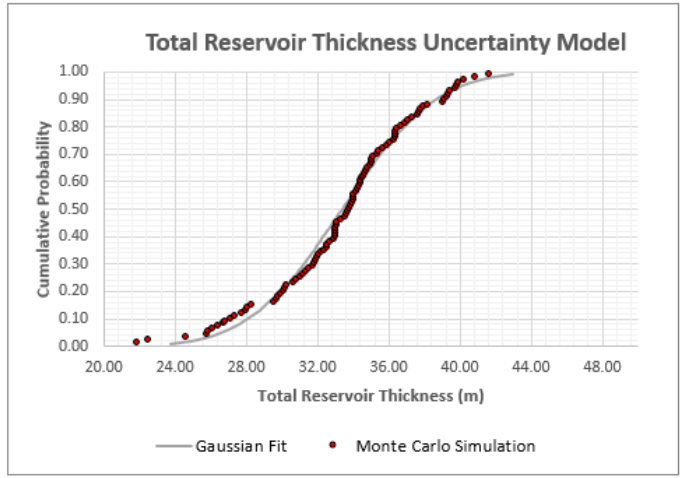
**Unit A + Unit B**

| Total Reservoir Thickness (m) $th_{total}^\ell$ |
|---|
| 34.71 |
| 31.45 |
| 35.42 |
| 30.14 |
| 33.09 |
| 36.10 |
| 31.95 |
| 31.39 |
| 33.69 |
| 34.24 |
| 34.45 |
| 29.98 |

**Calculate CDF**

| Cumulative Probability (fraction) | Percentile (m) |
|---|---|
| 0.01 | 21.93 |
| 0.02 | 22.50 |
| 0.03 | 24.66 |
| 0.04 | 25.85 |
| 0.05 | 25.88 |
| 0.06 | 26.11 |
| 0.07 | 26.41 |
| 0.08 | 26.73 |
| 0.09 | 26.84 |
| 0.10 | 27.14 |
| 0.11 | 27.42 |
| 0.12 | 27.77 |

| Fit Gaussian Distribution (m) |
|---|
| 23.77 |
| 24.89 |
| 25.61 |
| 26.14 |
| 26.58 |
| 26.95 |
| 27.28 |
| 27.57 |
| 27.84 |
| 28.08 |
| 28.31 |
| 28.52 |



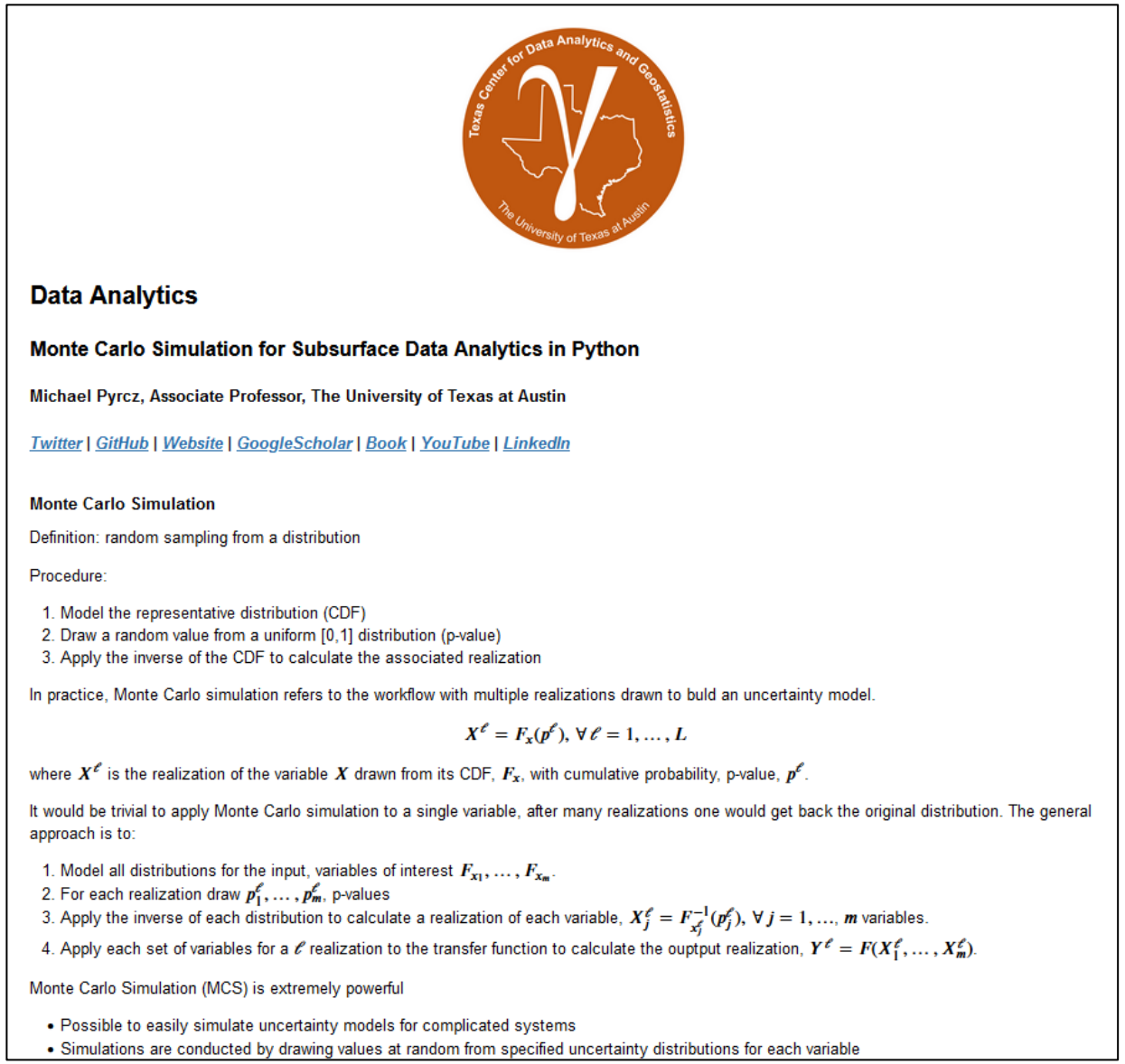

Monte Carlo simulation demonstration in Excel file MonteCarloSimulation.xlsx.

An opportunity to observe and review the steps of a simple Monte Carlo simulation workflow in Excel.

# Monte Carlo Simulation Exercise in Python

Ideas for things to try:

1. 50% increase in the average porosity
2. Set saturation max as 0.9
3. Set the small L size to 10!



## Data Analytics

### Monte Carlo Simulation for Subsurface Data Analytics in Python

**Michael Pyrcz, Associate Professor, The University of Texas at Austin**

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

**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 $X^{\ell}$ is the realization of the variable $X$ drawn from its CDF, $F_x$, with cumulative probability, p-value, $p^{\ell}$.

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:

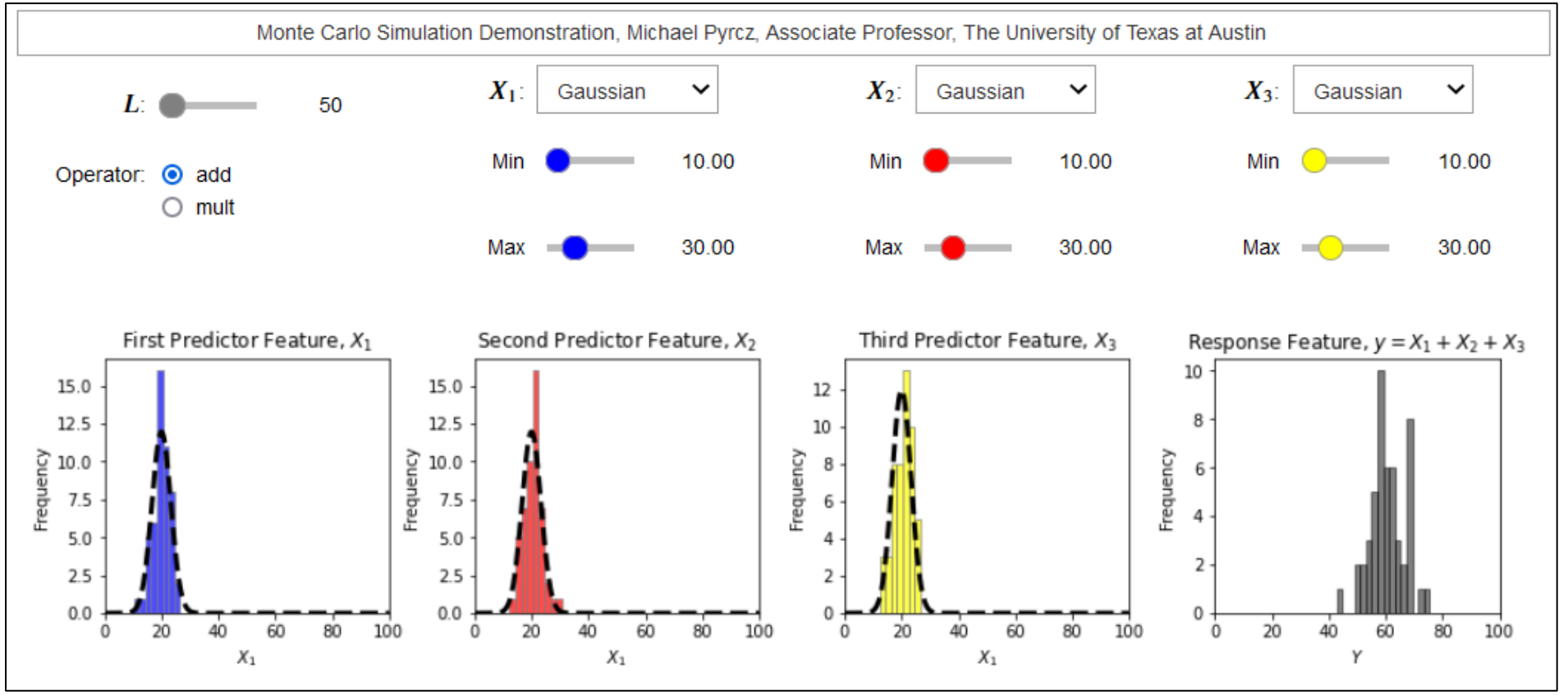
1. Model all distributions for the input, variables of interest $F_{x_1}, \dots, F_{x_m}$.
2. For each realization draw $p_1^{\ell}, \dots, p_m^{\ell}$, p-values
3. Apply the inverse of each distribution to calculate a realization of each variable, $X_j^{\ell} = F_{x_j}^{-1}(p_j^{\ell}), \forall j =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

Monte Carlo simulation demonstration in Python file is GeostatsPy_Monte_Carlo_simulation.ipynb.

# Interactive Monte Carlo Simulation Demonstration in Python




Interactive Monte Carlo Simulation in Python demo in file: Interactive_Monte_Carlo_simulation.ipynb

**Things to Try:**

1. Change the input distributions.
2. Change the number of realizations from only 1 to 10,000.
3. Change the transfer function from addition to multiplication.

# PGE 338 Data Analytics and Geostatistics

## Lecture 5: Univariate Distributions

**Lecture outline . . .**

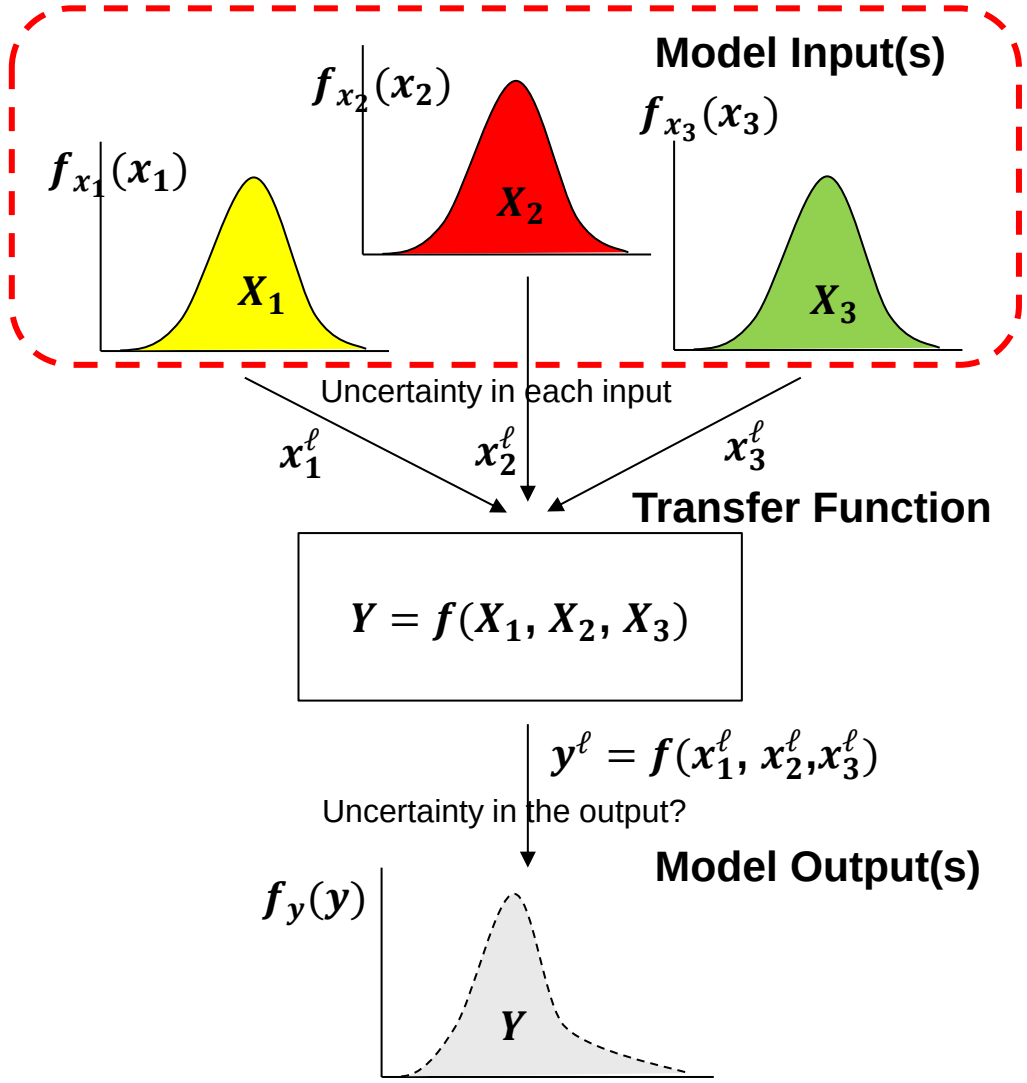
- **Bootstrap**

Michael Pyrcz, The University of Texas at Austin


- Introduction
- General Concepts
- Univariate
  - PDF / CDF
  - Statistics
  - Distributions
  - Heterogeneity
  - Hypothesis
- Bivariate
- Time Series Analysis
- Spatial Analysis
- Machine Learning
- Uncertainty Analysis

# Bootstrap Definition

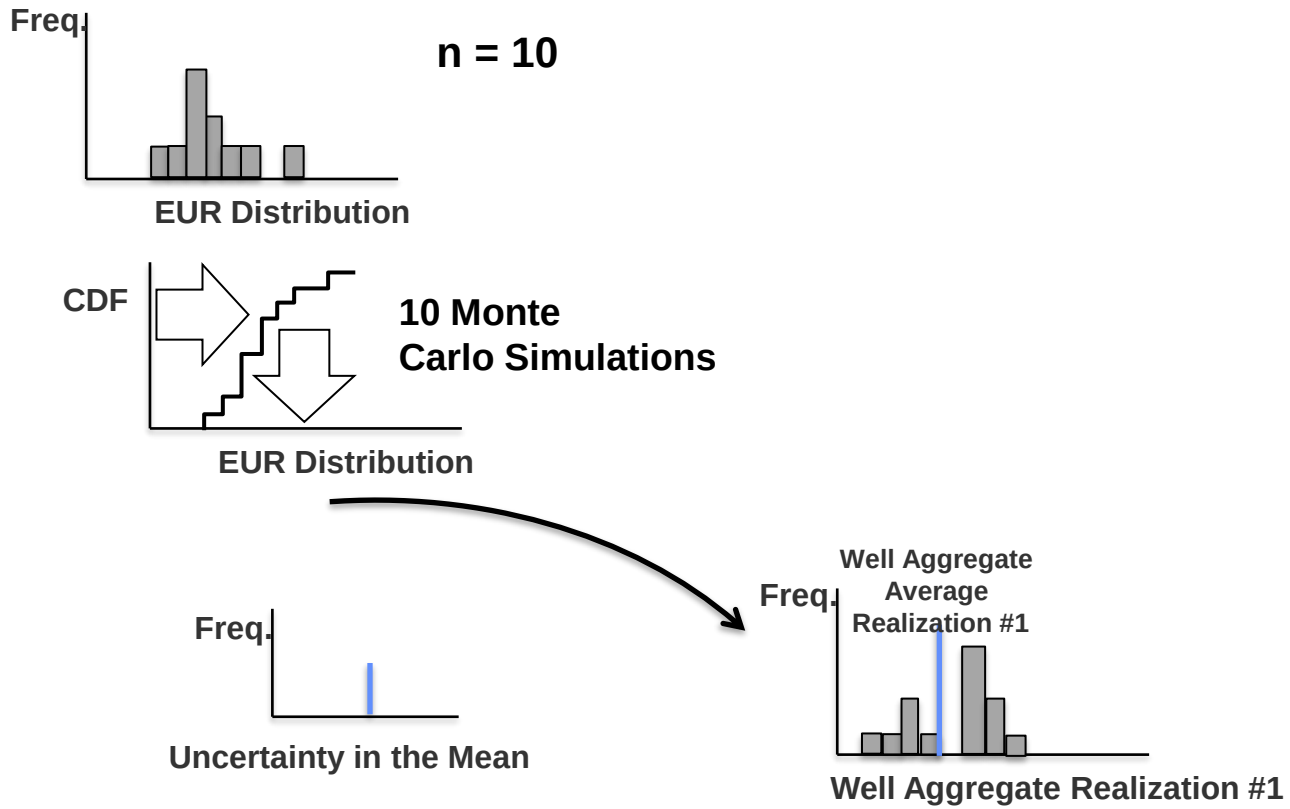
- Statistical resampling procedure to calculate uncertainty in a calculated statistic from the sample data itself.
- Uses repeated ($n$ times) Monte Carlo simulations from the dataset CDF, repeated ($n$ times) samples from the sample data with replacement.
- Builds the entire distribution for uncertainty in any statistic!
- We can use bootstrap to build the uncertainty distributions, **predictor feature RVs**, for a Monte Carlo Simulation workflow when that input is a statistic, e.g., average porosity.



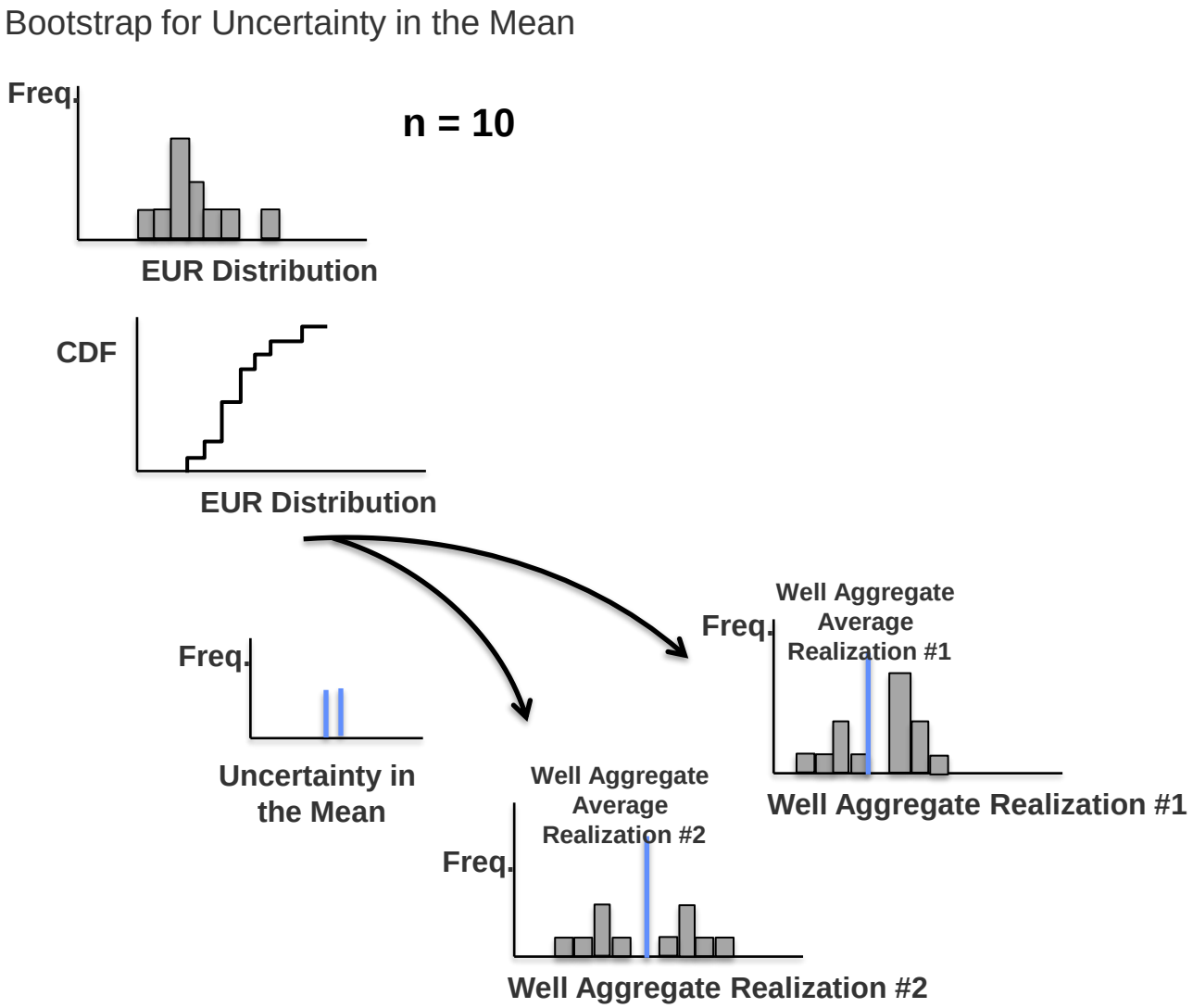

Monte Carlo simulation workflow with predictor feature RVs indicated by red, dashed line.

# Probability and Statistics
## Bootstrap

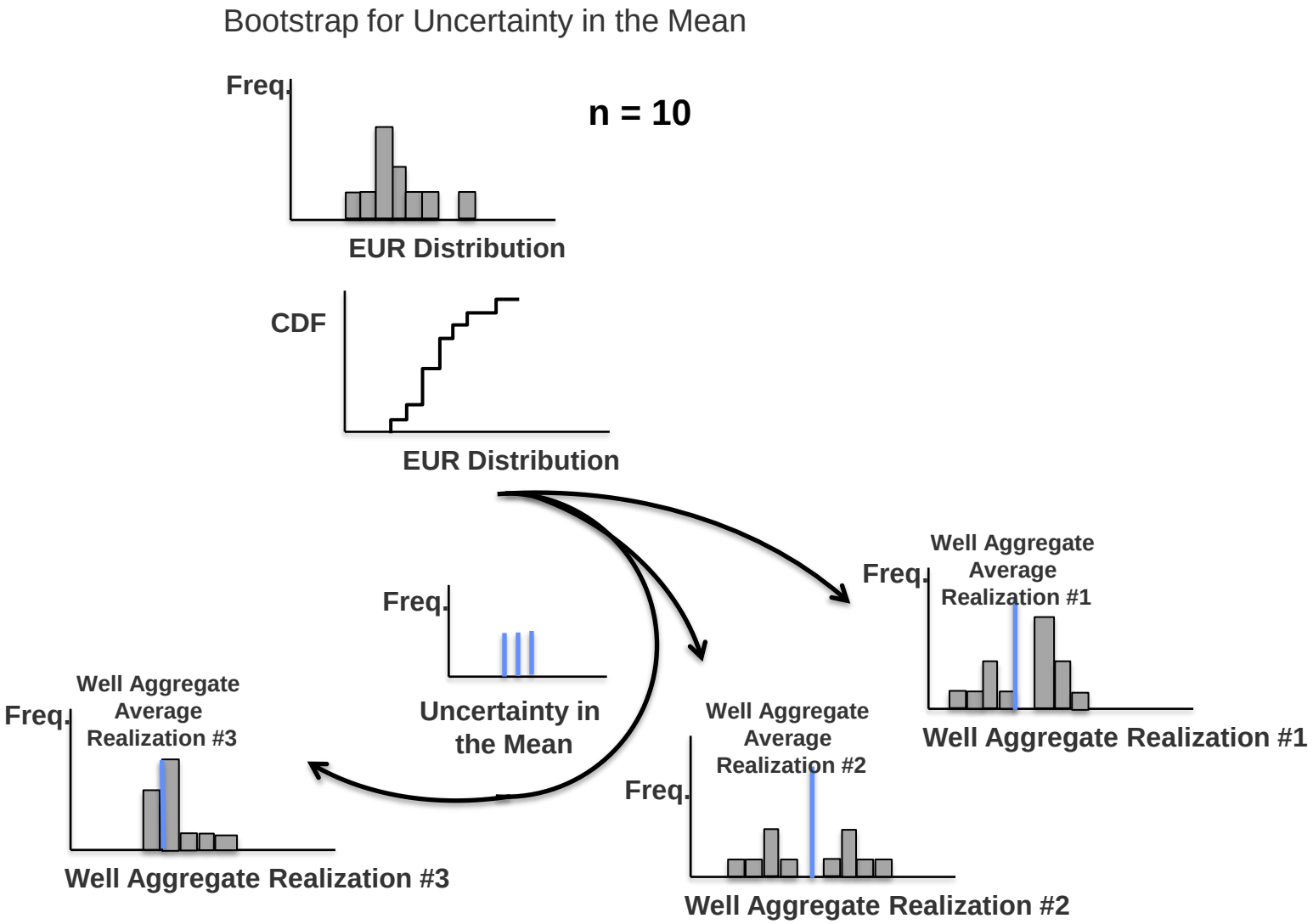
Bootstrap for Uncertainty in the Mean



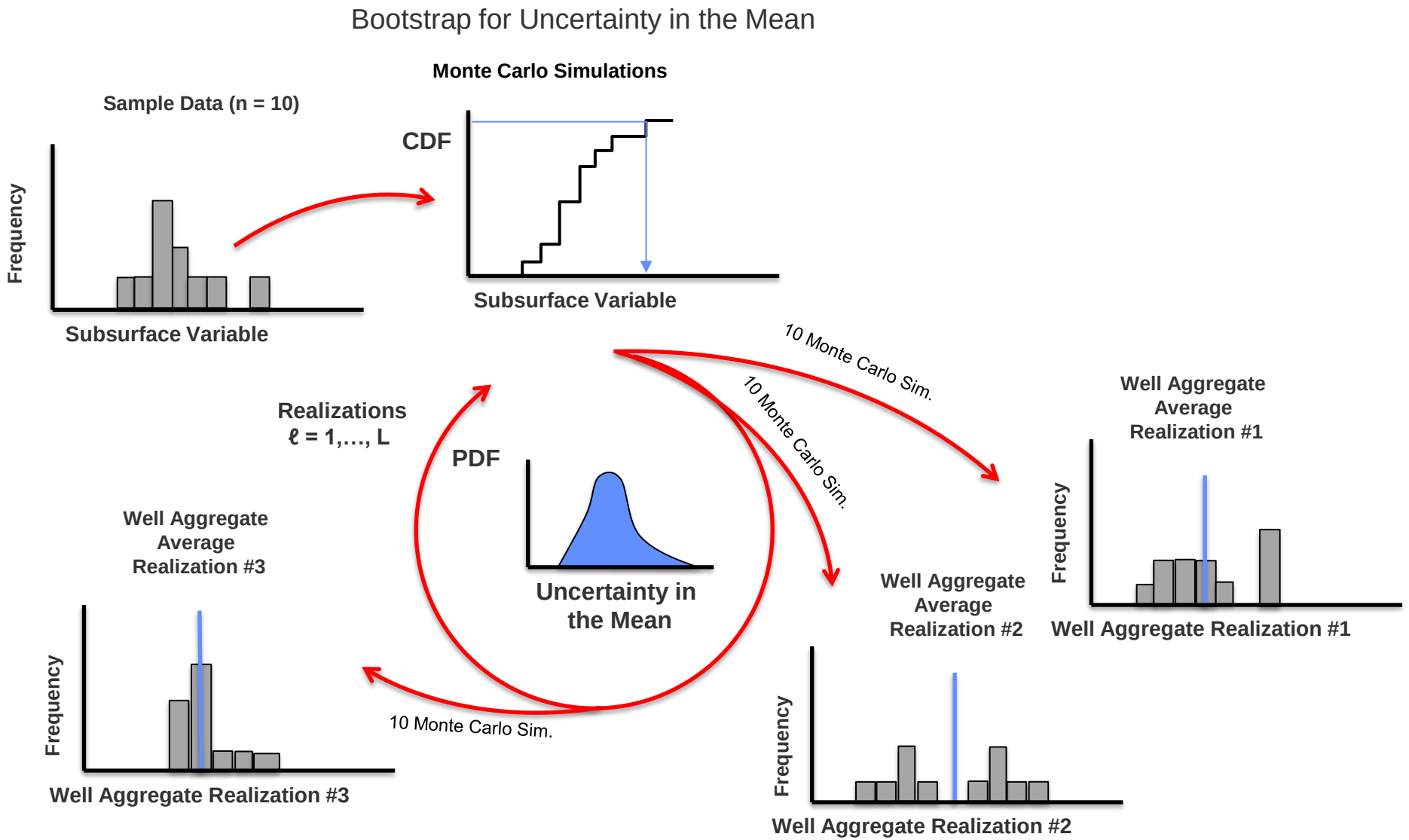
# Probability and Statistics
## Bootstrap



# Probability and Statistics
## Bootstrap

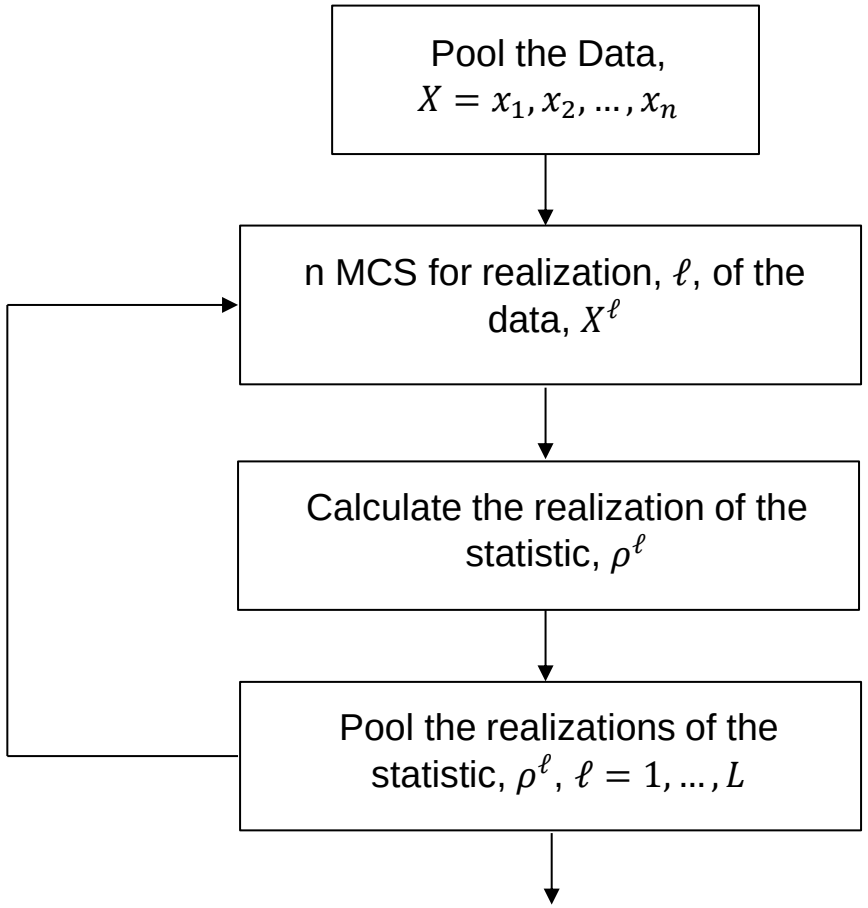


# Probability and Statistics
## Bootstrap



# More on Bootstrap

Let me reinforce, the bootstrap approach may be applied to calculate uncertainty in any statistic, realizations of any statistic from realizations of the data.




Bootstrap uncertainty in a statistic workflow.

**$\ell$, realization of the data distribution(s)**

**50th Percentile** $P_k(50)^\ell$

**Dykstra Parsons** $dp^\ell = \frac{P_k(50)^\ell - P_k(16)^\ell}{P_k(50)^\ell}$

**Welch's t-statistic**

2 paired distributions $\hat{t}^\ell = \frac{\bar{x}_1{}^\ell - \bar{x}_2{}^\ell}{\sqrt{\left(\frac{s_1^{2\,\ell}}{n_1} + \frac{s_2^{2\,\ell}}{n_2}\right)}}$

**$P_k(50)^\ell$, $dp^\ell$, $\hat{t}^\ell$ realization of the statistic**

# Probability and Statistics
## Bootstrap

Bootstrap Approach (Efron, 1982)

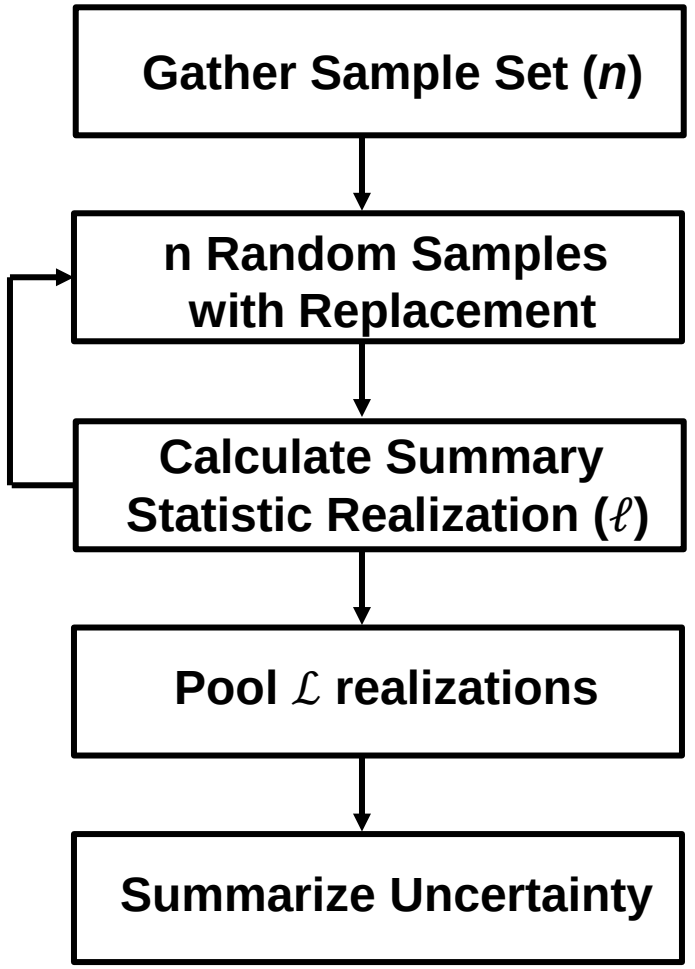
- Statistical resampling procedure to calculate uncertainty in a calculated statistic from the data itself.

- For uncertainty in the mean solution is standard error:

$$\sigma_{\bar{x}}^2 = \frac{\sigma_s^2}{n}$$

- Extremely powerful. Could get uncertainty in any statistic! For example, P13, skew, etc. not be possible without bootstrap.

- Advanced forms account for spatial information and strategy (game theory).



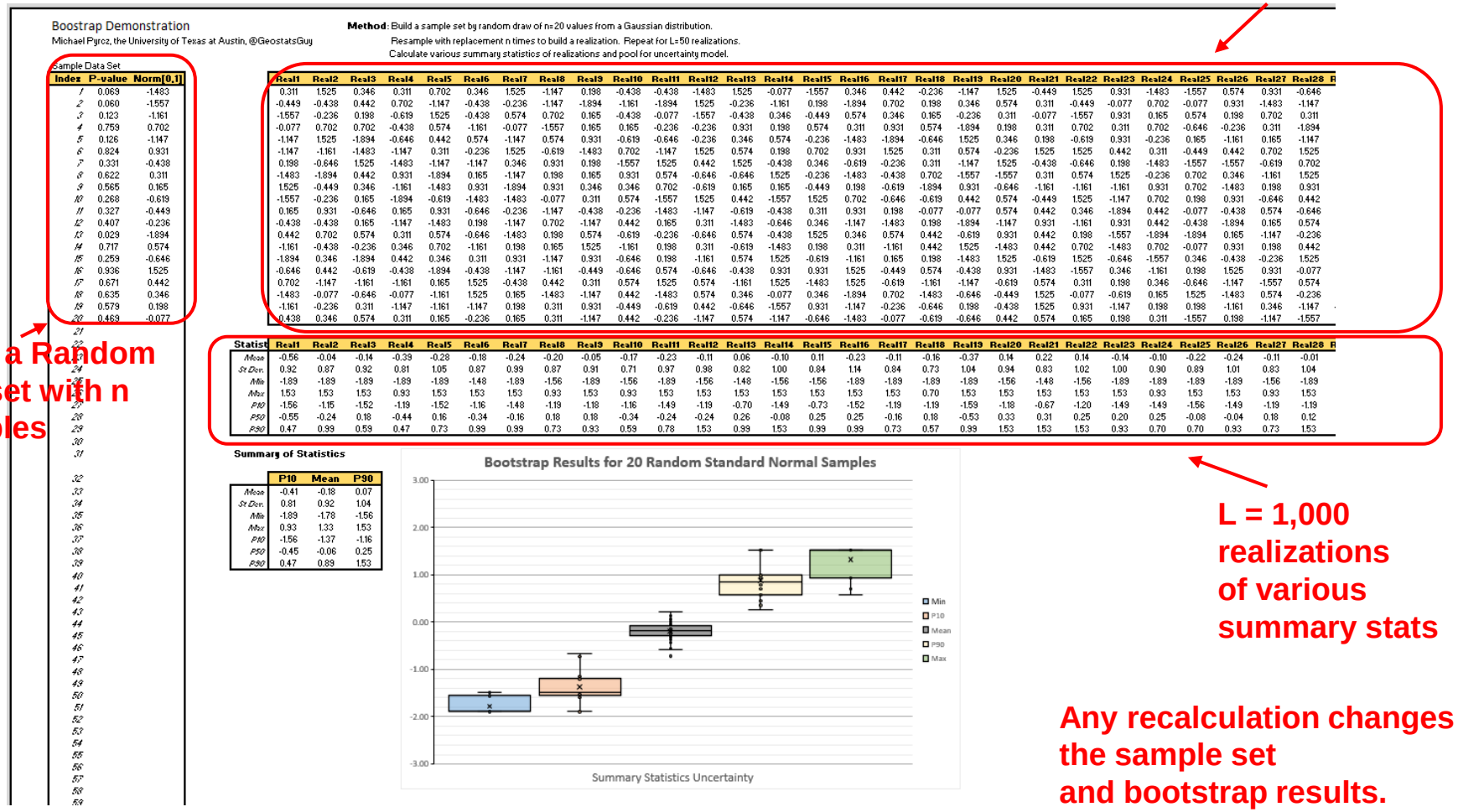

Bootstrap uncertainty in a statistic workflow.

# Bootstrap Demonstration in Excel

**You can bootstrap in Excel.**

- By using the 'vlookup' command with a random index we can randomly sample with replacements from a list of values.
- **This is random sampling with replacement, Monte Carlo simulation.**
- We just need to repeat this for L realizations and compile the L realizations of the statistic of interest.

**L = 1,000 realizations of n resample with replacement**




**L = 1,000 realizations of various summary stats**

**Any recalculation changes the sample set and bootstrap results.**

Excel bootstrap demonstration in file Bootstrap_Demo.xlsx

# Bootstrap Demonstration in Excel

**From the Bootstrap_Demo_Simple.xlsx calculate / demonstrate the following.**

1. That the bootstrap result for uncertainty in the mean is equivalent to standard error (hint: $\boldsymbol{SE} = \frac{\boldsymbol{s}}{\sqrt{\boldsymbol{n}}}$)?

2. Compare the uncertainty in the mean above with the case if the number of samples is cut in half? Does uncertainty go up or down?

# Bootstrap Demonstration in Excel

**From the Bootstrap_Demo_Simple.xlsx calculate / demonstrate the following.**

1. That the bootstrap result for uncertainty in the mean is equivalent to standard error (hint: $\boldsymbol{SE = \frac{s}{\sqrt{n}}}$)?

$$\boldsymbol{SE = \frac{s}{\sqrt{n}} = \frac{0.058}{\sqrt{20}} = 0.013,} \quad \boldsymbol{\sigma_{bootstrap} = 0.013}$$

2. Compare the uncertainty in the mean above with the case if the number of samples is cut in half? Does uncertainty go up or down?

$$\boldsymbol{SE = \frac{s}{\sqrt{n}} = \frac{0.058}{\sqrt{10}} = 0.018}$$

Uncertainty in the mean increases as the number of samples, $n$, decreases.

# Bootstrap Demonstration in Excel

## Bootstrap Practice

- What is the uncertainty in the average porosity for 2D_Porosity_Small.csv?

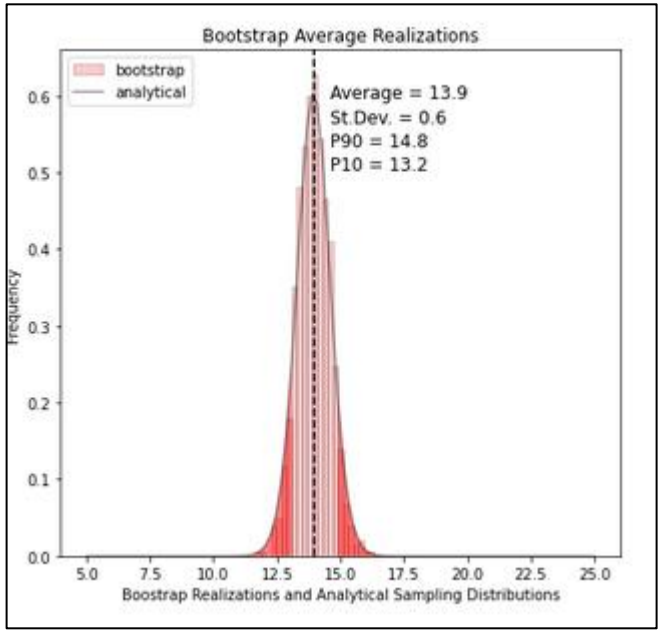
- Statistic: Average, $\bar{x}_{\phi} = \frac{1}{12}\sum_{i=1}^{12} \bar{x}_{\phi_i}$

- Steps:
  - 30 (number of samples in dataset) resamples with replacement
  - calculate average
  - repeat 1,000 times to build a distribution of averages
  - calculate P10 and P90 to report to management
  - fit entire distribution and use it in subsequent Monte Carlo simulation (e.g., OIP for Unit B)

# Bootstrap Demonstration in Python

Bootstrap in Python demonstration

- Including a general Python bootstrap function.
- Applied to a variety of statistics.




**Data Science Basics in Python**

**Bootstrap for Uncertainty Models**

**Michael Pyrcz, Associate Professor, University of Texas at Austin**

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

**Bootstrap in Python**

Here's a simple workflow, demonstration of bootstrap for modeling workflows. This should help you get started with this important data analytics method to evaluate and integrate uncertainty for any statistic or model.

**Bootstrap**

Uncertainty in statistics

- one source of uncertainty is the paucity of data.
- do 200 or even less samples provide a precise (and accurate estimate) of the mean? standard deviation? skew? P13?

Would it be useful to know the uncertainty in these statistics due to limited sampling?

- what is the impact of uncertainty in the mean porosity e.g. 20%+/-2%?
- empirically calculate standard errors, confidence intervals and to conduct hypothesis testing

**Bootstrap** is a method to assess the uncertainty in a statistic by repeated random sampling with replacement to create multiple simulated data sets.

**The Bootstrap Workflow (Efron, 1982)**

Statistical resampling procedure to calculate uncertainty in a calculated statistic from the data itself.

- Does this work? Prove it to yourself, for uncertainty in the mean solution is standard error:

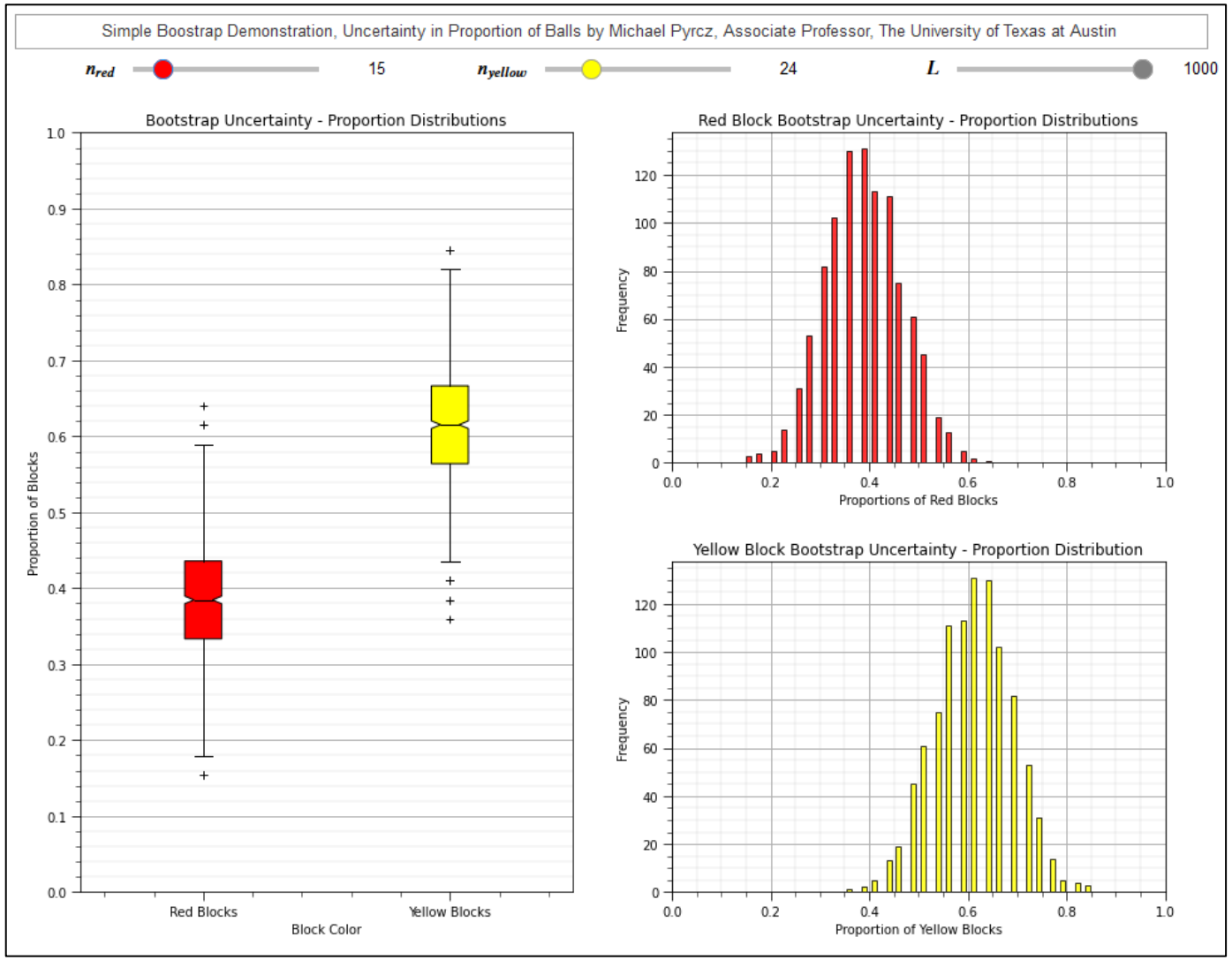
$$\sigma^2_{\bar{x}} = \frac{\sigma^2_s}{n}$$

Bootstrap in Python demo in file: PythonDataBasics_Bootstrap.ipynb

# Interactive Bootstrap Demonstration in Python

Interactive bootstrap in Python demonstration

- The red and yellow blocks with a code-based cowboy hat.
- The code version of the in-class demonstration.




Interactive Bootstrap in Python demo in file: Interactive_Bootstrap_Simple.ipynb

# A phone call this morning…

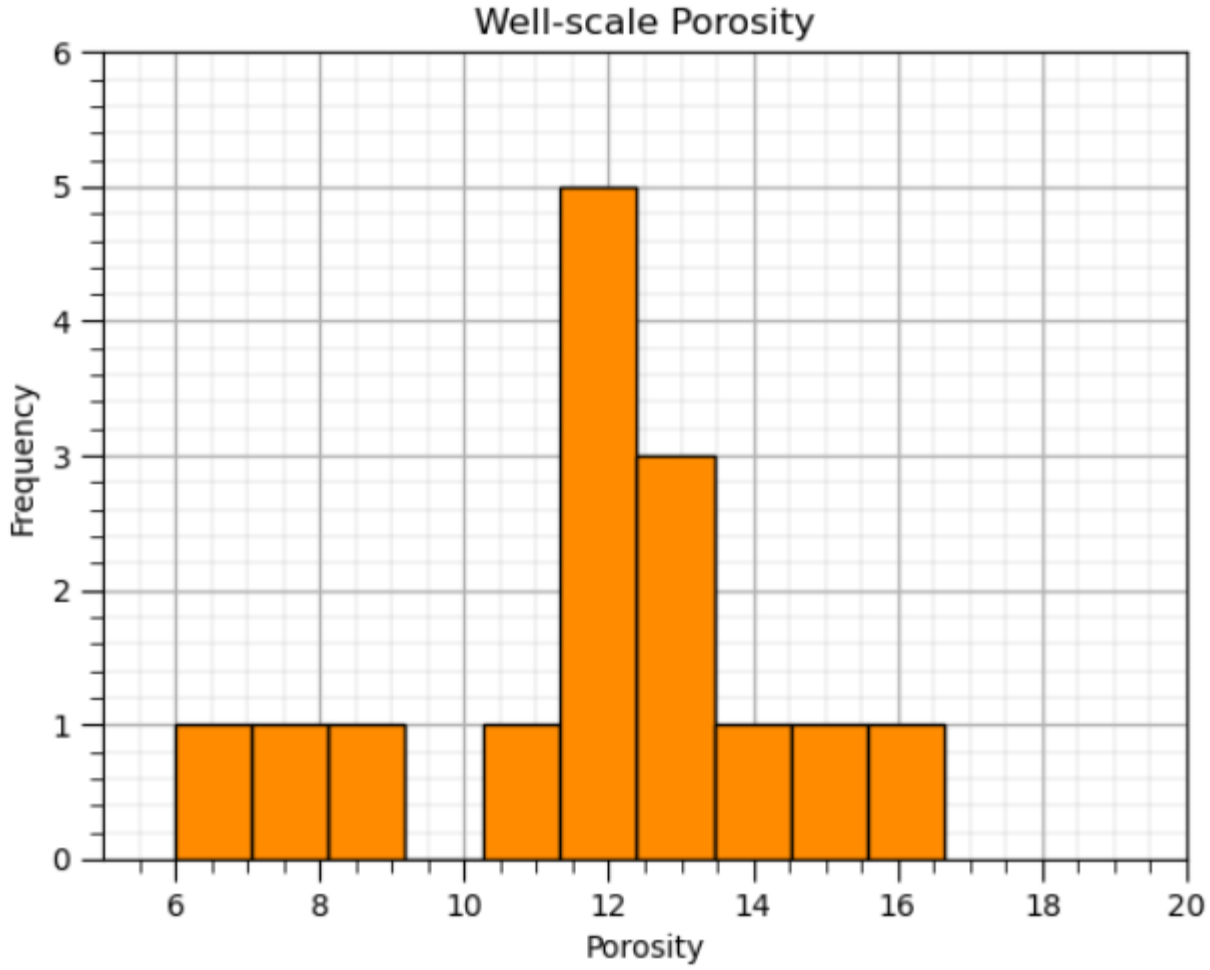
I got a phone call this morning from Tiffany Wilson, exploration manager of Western Canada.

They just got 15 new wells with well scale porosity. They need a fast OIP uncertainty model (Well_Porosity_MooseJaw.csv).

$$OIP = \bar{\phi} \cdot \overline{th} \cdot \overline{s_o} \cdot A$$

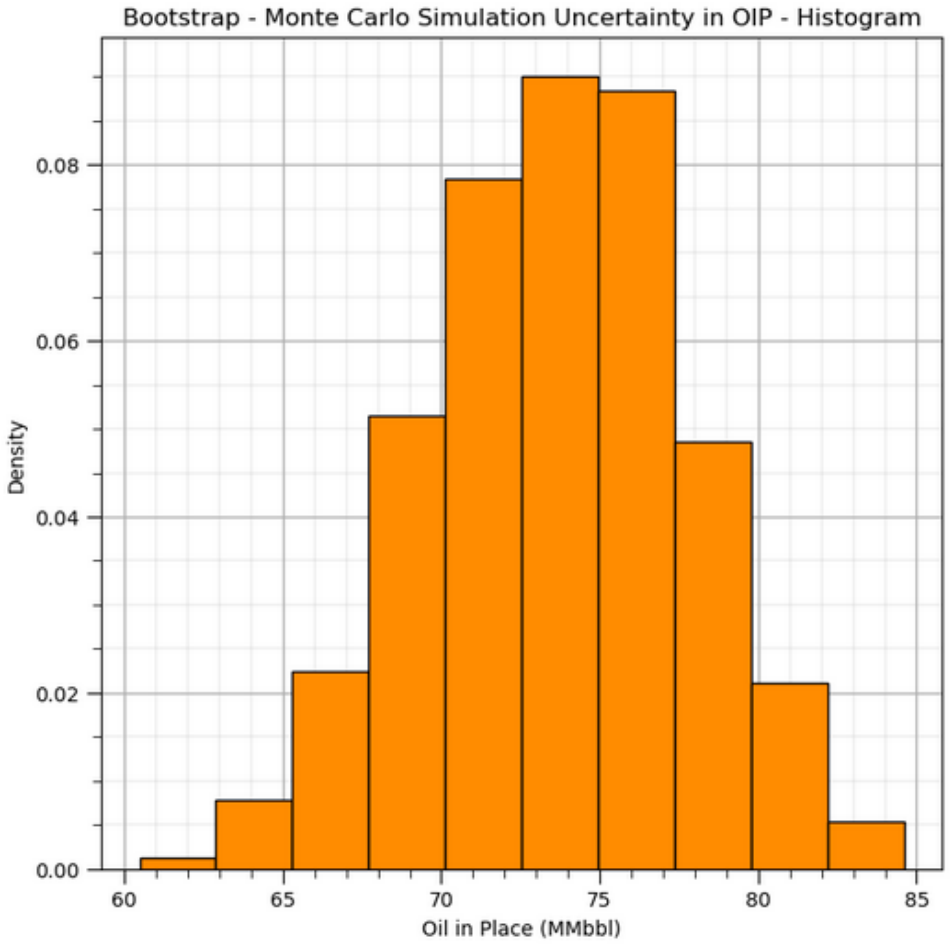
Area = 1,000,000 m^3
Average Thickness = 100 m
$s_o$ = 1.0
bbl/m^3 = 6.28981

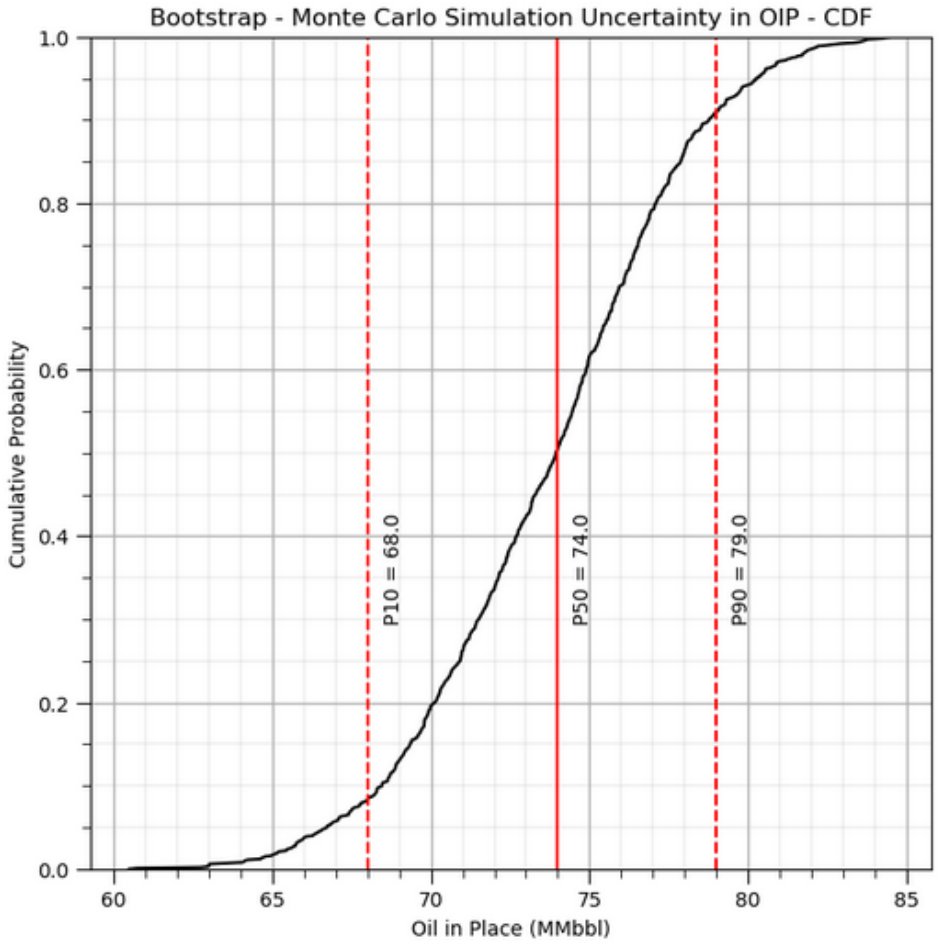
random sample – np.random.choice()
calc average – np.average()
loop – for i in range(0,n):
percentile – np.percentile()




The Data set with 15 new wells: Well_Porosity_MooseJaw.csv

# A phone call this morning…




A OIP Uncertainty Model for Well_Porosity_MooseJaw.csv

# PGE 338 Data Analytics and Geostatistics

## Lecture 5: Univariate Distributions

**Lecture outline . . .**

- **Distribution Transforms**

**Michael Pyrcz, The University of Texas at Austin**


**Introduction**

**General Concepts**

**Univariate**

- **PDF / CDF**
- **Statistics**
- **Distributions**
- **Heterogeneity**
- **Hypothesis**

**Bivariate**

**Time Series Analysis**

**Spatial Analysis**

**Machine Learning**

**Uncertainty Analysis**

# An Opportunity to Excel

VP Gulf of Mexico Exploration in Your Office at 8 am Monday

We have 10 wells in a new reservoir. At each well you have the features, porosity ($\phi$) and reservoir thickness ($th$). Average saturation & area are assumed as constants ($\bar{s_o}$, $A$).

$$OIP = \bar{\phi} \cdot \overline{th} \cdot \bar{s_o} \cdot A$$

Develop a workflow to calculate an uncertainty model for oil in place ($OIP$).

1. Apply bootstrap to calculate the uncertainty distributions for $\bar{\phi}$, $\overline{th}$.

2. Apply Monte Carlos simulation, draw random realizations from the uncertainty distributions of $\bar{\phi}$, $\overline{th}$, apply to the transfer function to calculate $L$ random realizations of $OIP$.

3. Summarize distribution of the realizations of $OIP$, e.g., histogram, CDF, p10, mean, p90.

Note: if all you need is the uncertainty distribution for $OIP$ you could directly apply the bootstrap realizations of $\bar{\phi}$, $\overline{th}$ directly to the transfer function, combining steps 1 and 2.

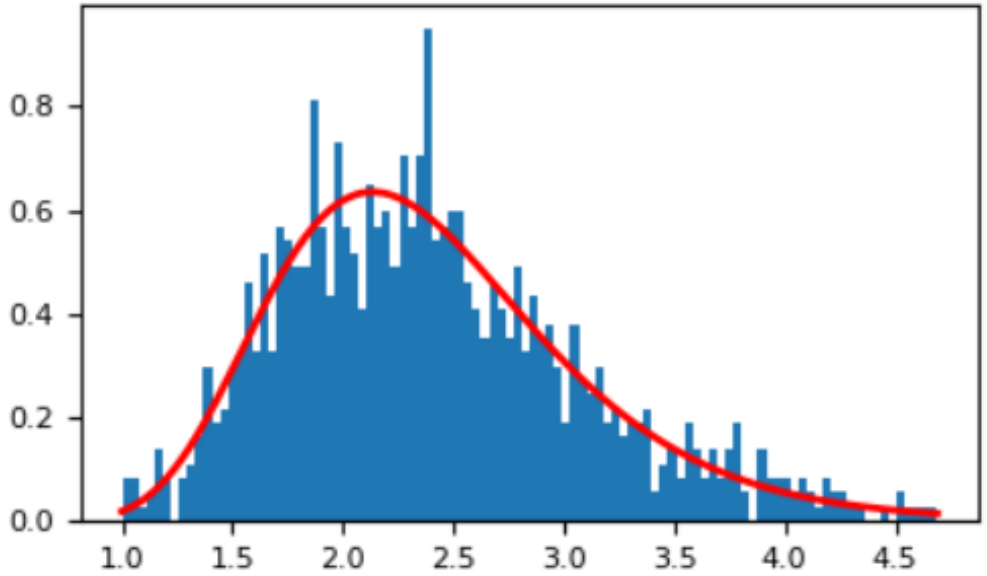
# Distribution Transforms

## Distribution Transforms

Transform the data to have a new distribution, PDF / CDF

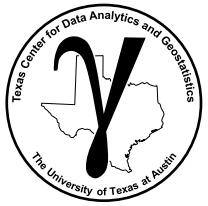
How is distribution transformation used in data analytics, geostatistics and machine learning workflows?

- **1. Inference:** to match a specific distribution that is expected for a feature
- **2. Data Preparation / Cleaning:** correcting for too few data and outliers
- **3. Theory:** to match a specific distribution required for a method in your workflow




Noisy data fit to a lognormal distribution.

Image from https://docs.scipy.org/doc/numpy-1.14.0/reference/generated/numpy.random.lognormal.html




# Distribution Transforms

## Distribution Transforms

Transform the data to have a new distribution, PDF / CDF

- Mapping from one distribution to another through cumulative probabilities / percentiles

$$Y = G_Y^{-1}(F_X\,(X))$$

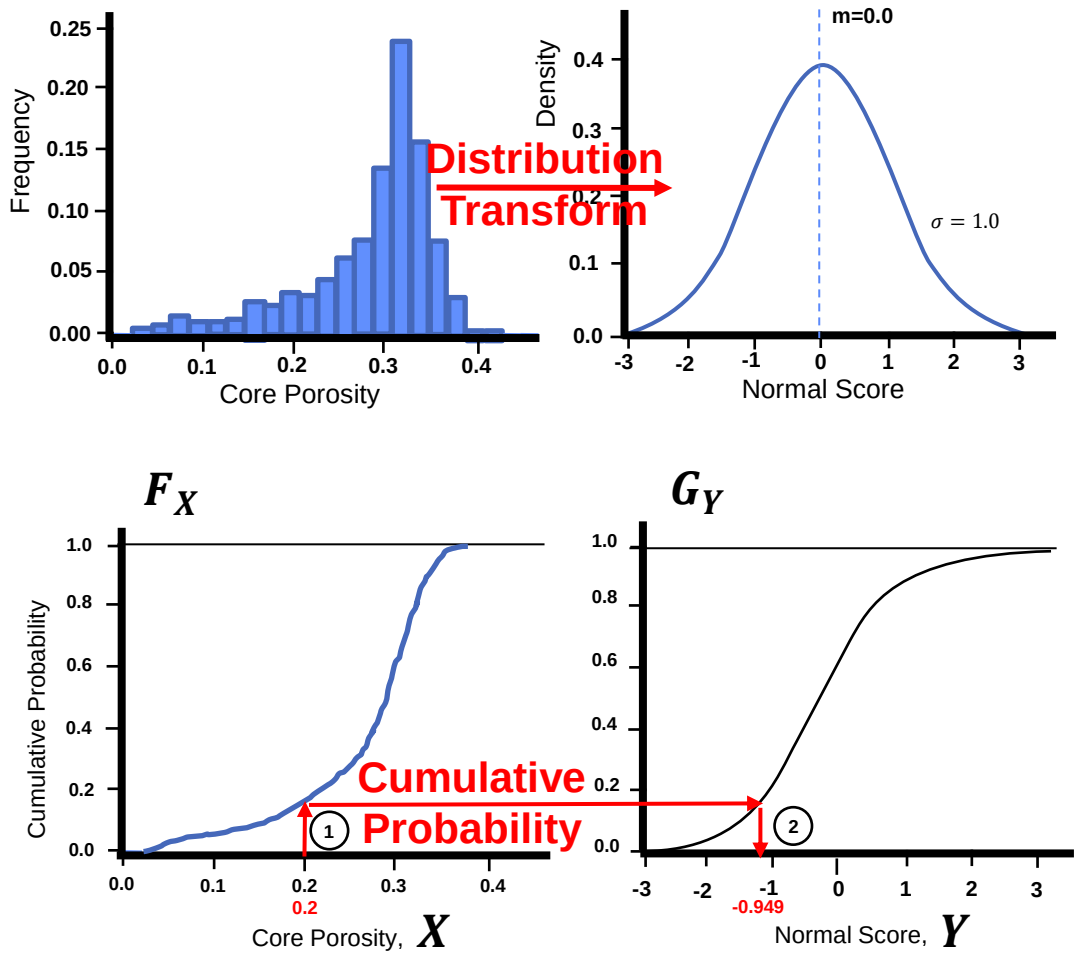
- This may be applied to any parametric or nonparametric distributions

Each original value, $x_1, x_2, \ldots, x_n$ is mapped to new values, $y_1, y_2, \ldots, y_n$

- Rank preserving transform, e.g., P50 of $X$ is still P50 of $Y$ etc.

# Distribution Transforms

## Distribution Transform Graphical Representation of $Y = G_Y^{-1}(F_X(X))$

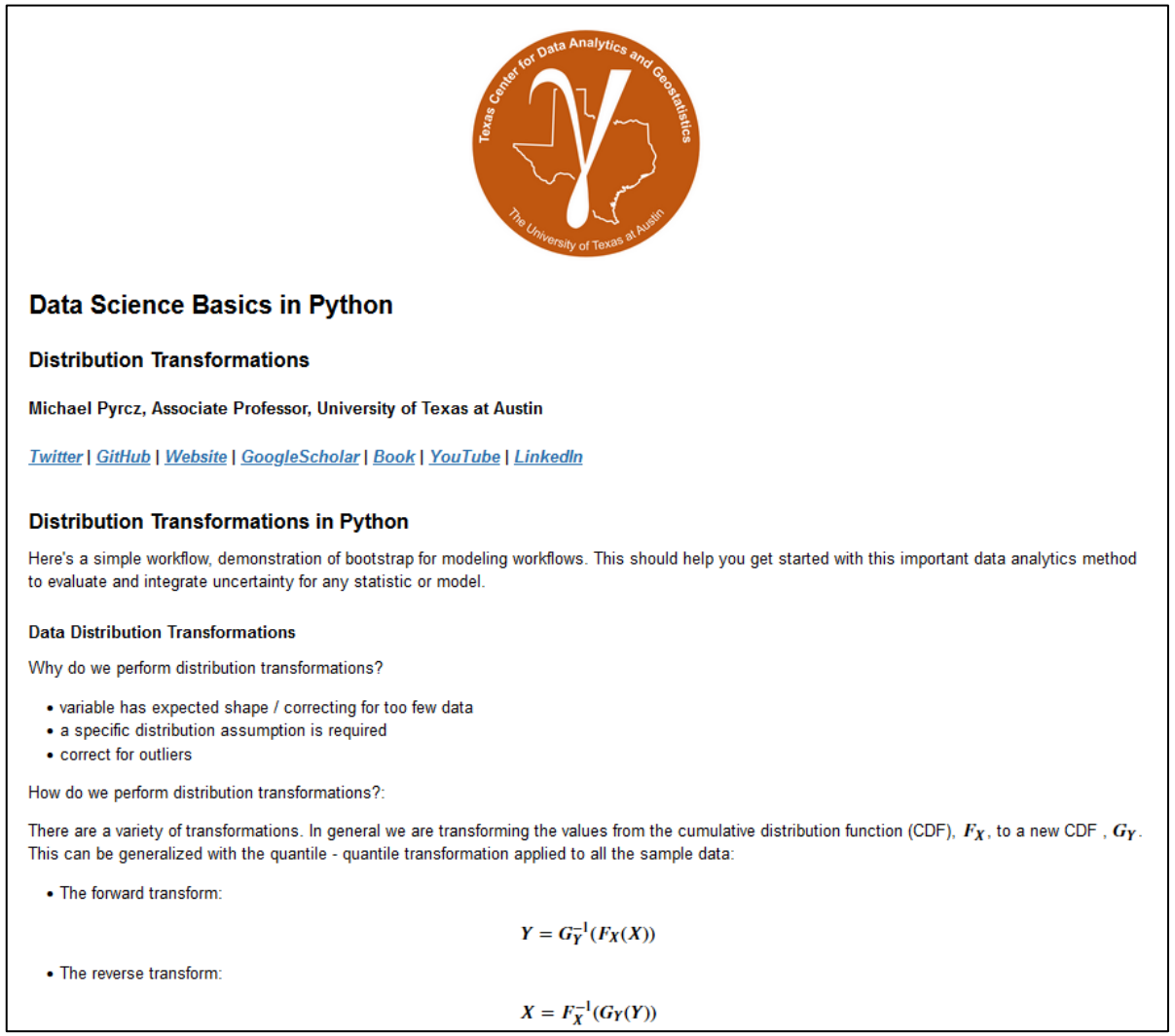



Graphical representation of NSCORE transform, $Y = G_Y^{-1}(F_X(X))$

# Distribution Transforms in Python

**Well-documented workflow with distribution transformations**

- codes for transformations to parametric and nonparametric distributions.



## Data Science Basics in Python

### Distribution Transformations

**Michael Pyrcz, Associate Professor, University of Texas at Austin**

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

### Distribution Transformations in Python

Here's a simple workflow, demonstration of bootstrap for modeling workflows. This should help you get started with this important data analytics method to evaluate and integrate uncertainty for any statistic or model.

#### Data Distribution Transformations

Why do we perform distribution transformations?

- variable has expected shape / correcting for too few data
- a specific distribution assumption is required
- correct for outliers

How do we perform distribution transformations?:

There are a variety of transformations. In general we are transforming the values from the cumulative distribution function (CDF), $F_X$, to a new CDF , $G_Y$. This can be generalized with the quantile - quantile transformation applied to all the sample data:

- The forward transform:

$$Y = G_Y^{-1}(F_X(X))$$

- The reverse transform:

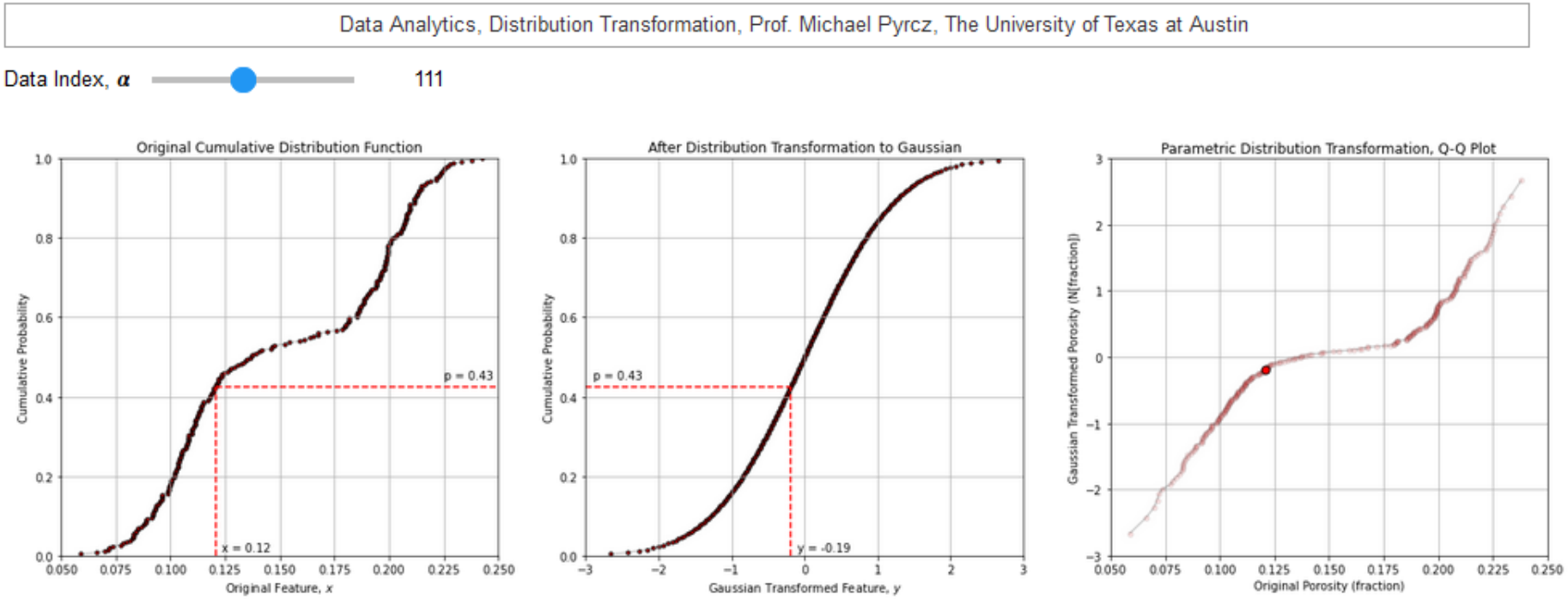
$$X = F_X^{-1}(G_Y(Y))$$

Demonstrations of distribution transformations to parametric and nonparametric distributions PythonDataBasics_Distribution_Transformations.ipynb

# Interactive Distribution Transforms in Python

**Distribution Transform Graphical Representation of $Y = G_Y^{-1}(F_X(X))$**

- Demonstrations of distribution transformations to parametric and nonparametric distributions.




Interactive demonstration of distribution transformation to parametric distributions Interactive_Distributions_Transformations.ipynb.

# Distribution Transforms

## Distribution Transform Examples

- Well Log Porosity ($\boldsymbol{WL\varphi}$) [10, 13, 14, 15, 17]
- Core Porosity ($\boldsymbol{Core\varphi}$) [6, 9, 10, 13, 17]

How would you transform the Log Porosity to the Core Porosity Distribution?

We need to do this. $Y = G_Y^{-1}(F_X\,(X))$ where F is CDF of Well Log Porosity and G is CDF of core

- We need the CDF of both.

- We could further specify this transformation by substituting $Y$ and $X$ with $Core\varphi$ and $WL\varphi$ and indicating data sample index $i$:

$$Core\varphi_i = G_{Core\varphi}^{-1}\left(F_{WL\varphi}\,(WL\varphi_i)\right)$$

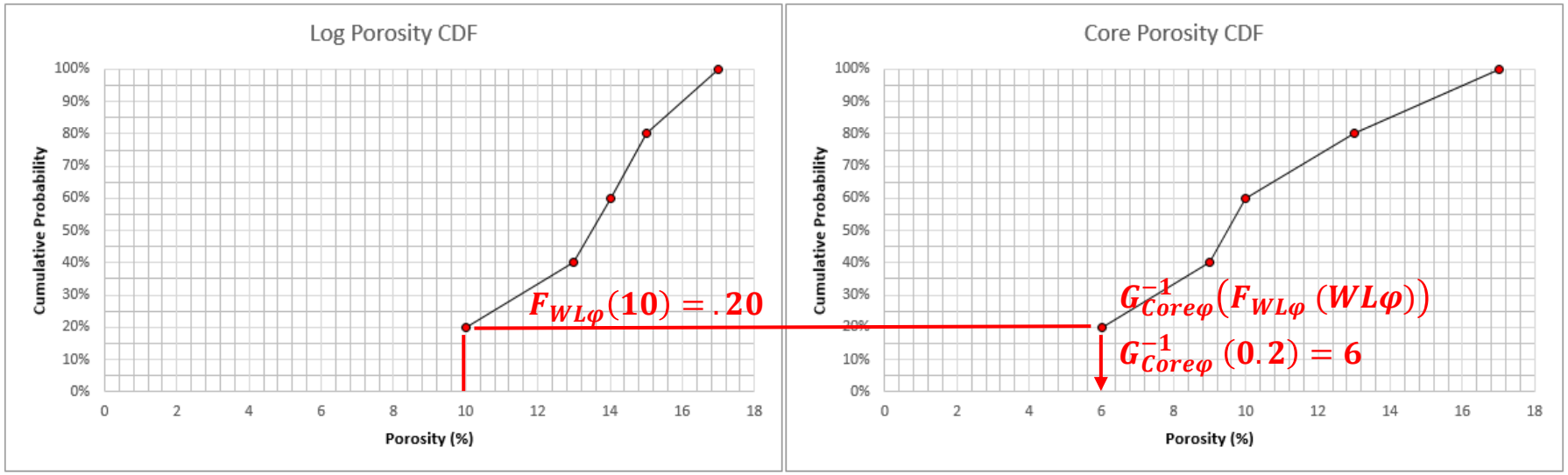
# Distribution Transforms Example

How would you transform the Log Porosity to the Core Porosity Distribution?

| Index | Cum Prob. | Log | Core |
|---|---|---|---|
| 1 | 20% | 10 | 6 |
| 2 | 40% | 13 | 9 |
| 3 | 60% | 14 | 10 |
| 4 | 80% | 15 | 13 |
| 5 | 100% | 17 | 17 |

**Well Log Transformed to Core**

| Transformed |
|---|
| 6 |
| 9 |
| 10 |
| 13 |
| 17 |




Data CDFs and graphical mapping of the distribution transformation for data sample index 1.

If you have the same number of data, sort the data, data have the same cumulative probabilities so you can just substitute the values!

# Distribution Transforms Example

**How would you transform this core porosity to:**

N[0,1] Distribution (standard normal with mean = 0.0, standard deviation = 1.0)?

| Core Porosity |
|---|
| 5 |
| 7 |
| 8 |
| 9 |
| 9 |
| 10 |
| 13 |
| 15 |
| 17 |
| 29 |

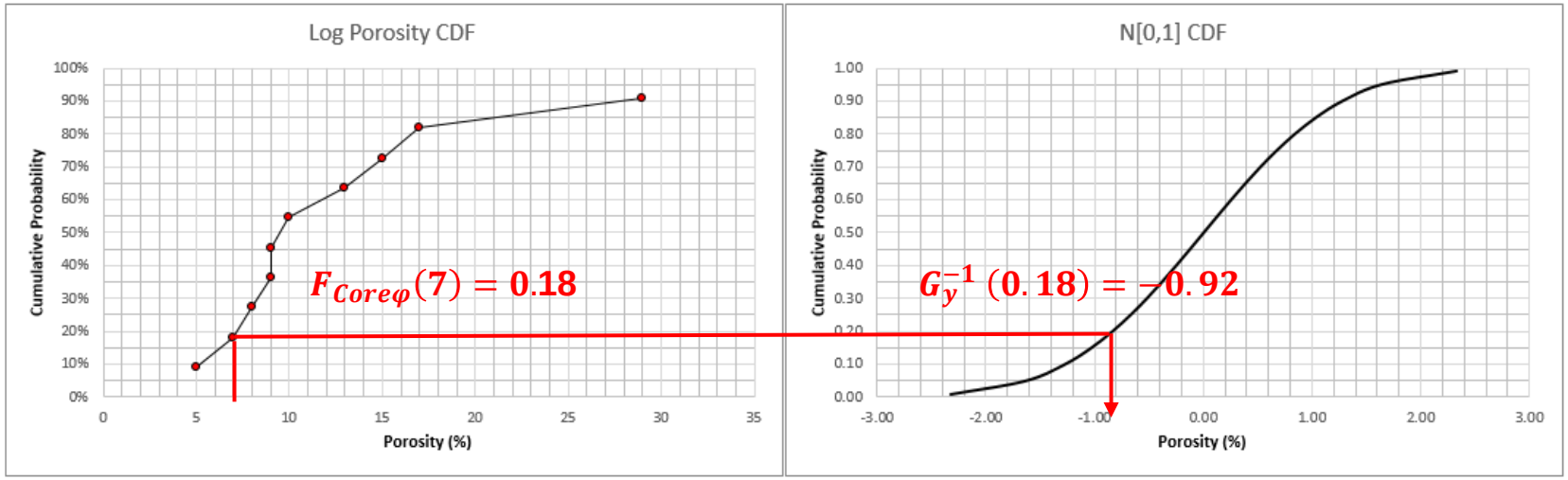
- Use the $i/_{N+1}$ basis so that we assume tails are not known (recall the Gaussian Distribution is unbounded).

What are the steps?

# Distribution Transforms Example

**How would you transform this core porosity to:**

N[0,1] Distribution (standard normal with mean = 0.0, standard deviation = 1.0)?




Data CDF and standard normal CDF and graphical mapping of the distribution transformation for data sample index 1.

- Use the $^{i}/_{N+1}$ basis so that we assume tails are not known (recall the Gaussian Distribution is unbounded).

What are the steps?

1. Calculate the cumulative probability for each sorted data sample, $F\ (Core\varphi_i)\ = {}^{i}/_{N+1}$
2. Take the cumulative probability for each value in your dataset and apply the inverse of the target distribution, N[0,1]. HINT: NORM.INV(<Cumulative Probability>, 0.0,1.0)

# Distribution Transforms Example

**How would you transform this core porosity to:**

N[0,1] Distribution (standard normal with mean = 0.0, standard deviation = 1.0)?

| Sample | Core Porosity | F(Coreϕ) |
|---|---|---|
| 1 | 5 | 9% |
| 2 | 7 | 18% |
| 3 | 8 | 27% |
| 4 | 9 | 36% |
| 5 | 9 | 45% |
| 6 | 10 | 55% |
| 7 | 13 | 64% |
| 8 | 15 | 73% |
| 9 | 17 | 82% |
| 10 | 29 | 91% |

$G_N^{-1}(F_\varphi(\varphi))$

**Norm.Inv**

| N[Core Porosity] |
|---|
| -1.34 |
| -0.91 |
| -0.60 |
| -0.35 |
| -0.11 |
| 0.11 |
| 0.35 |
| 0.60 |
| 0.91 |
| 1.34 |

- Use the $^{i}/_{N+1}$ basis so that we assume tails are not known (recall the Gaussian Distribution is unbounded).

What are the steps?

1. Calculate the cumulative probability for each sorted data sample, $F(Core\varphi_i) = {}^{i}/_{N+1}$
2. Take the cumulative probability for each value in your dataset and apply the inverse of the target distribution, N[0,1]. HINT: NORM.INV(<Cumulative Probability>, 0.0,1.0)

# Distribution Transforms Exercise

**Now you try.  Transform these fraction of shale (Vsh) values to N[10,4].**

3, 5, 8, 9, 11,14, 19% Vsh

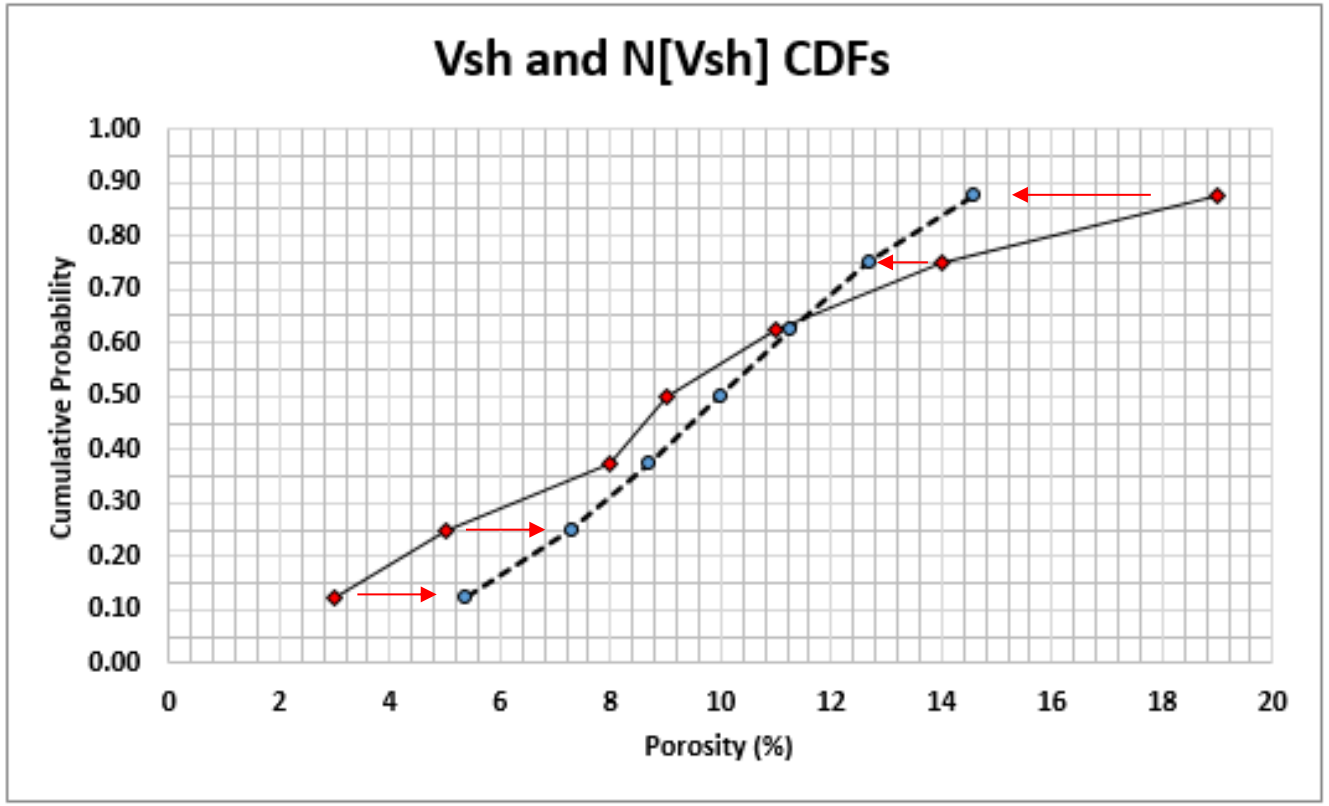
# Distribution Transforms Exercise

**Now you try. Transform these fraction of shale (Vsh) values to N[10,4].**

3, 5, 8, 9, 11,14, 19% Vsh

| Index | Vsh | Cumul. Prob. | N[Vsh] |
|---|---|---|---|
| 1 | 3 | 0.13 | 5.40 |
| 2 | 5 | 0.25 | 7.30 |
| 3 | 8 | 0.38 | 8.73 |
| 4 | 9 | 0.50 | 10.00 |
| 5 | 11 | 0.63 | 11.27 |
| 6 | 14 | 0.75 | 12.70 |
| 7 | 19 | 0.88 | 14.60 |

Original sorted data and transformed to N[10,4].




Data and transformed data CDFs with assume linear interpolation (lines between points).

# PGE 338 Data Analytics and Geostatistics

## Lecture 5: Univariate Distributions

**Lecture outline . . .**

- **Parametric Distributions**
- **Nonparametric Distributions**
- **Monte Carlo Simulation**
- **Bootstrap**
- **Distribution Transforms**

**Michael Pyrcz, The University of Texas at Austin**


- Introduction
- General Concepts
- Univariate
  - PDF / CDF
  - Statistics
  - Distributions
  - Heterogeneity
  - Hypothesis
- Bivariate
- Time Series Analysis
- Spatial Analysis
- Machine Learning
- Uncertainty Analysis