Economic Risk and Decision Analysis for Oil and Gas Industry
CE81.9008

School of Engineering and Technology
Asian Institute of Technology

January Semester

Presented by
Dr. Thitisak Boonpramote

Department of Mining and Petroleum Engineering, Chulalongkorn University

Simulation

Deterministic vs. Stochastic

Some phenomena have a **pre-determined path** (more or less):

- buying a bond: the returns are fixed.
- a falling stone: the rate of acceleration is fixed.

Most will follow an **uncertain path**:

- buying a stock: the returns are speculative
- tomorrow's weather: there are theoretical limits of predictability
- rolling dice etc.

Deterministic events have **fixed outcomes**.

Stochastic events have **uncertain outcomes**

- the outcomes are governed by **random processes**.

Stochastic analysis uses **random variables** as inputs, parameters, constraints.

Random Phenomenon & Random Variable

A ***random phenomenon*** is a **repeatable process** with **individual outcomes** that **cannot be predicted**.

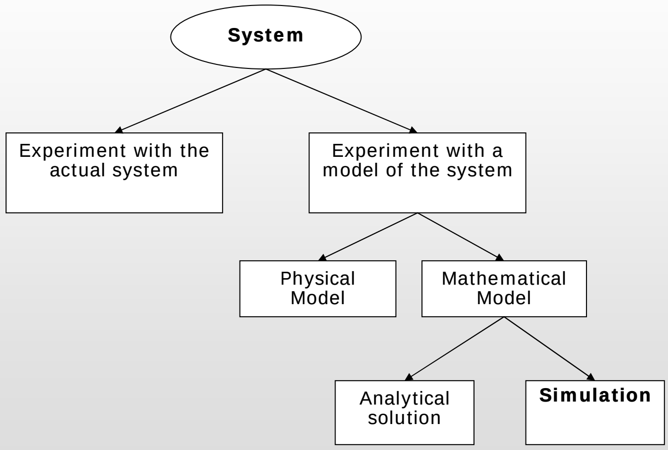
- Although individual outcomes are impossible to predict, sometimes a **clear distribution** emerges after **a large number of repetitions**.
- *Example:*

  The flip of a coin is a random phenomenon with a **known distribution**: a large number coin flips result in half heads and half tails.

A ***random variable*** is **assigned values** as a result of a random phenomenon.

- The result of a coin flip is a **random variable**. It will either take on the value of a **head** or a **tail**.
- Other examples of random variables?
  - precipitation (and other meteorological variables)
  - the size of the pieces of a rock when crushed

What are Simulations?



What are simulations?

An attempt to **model the real world** using a representational framework

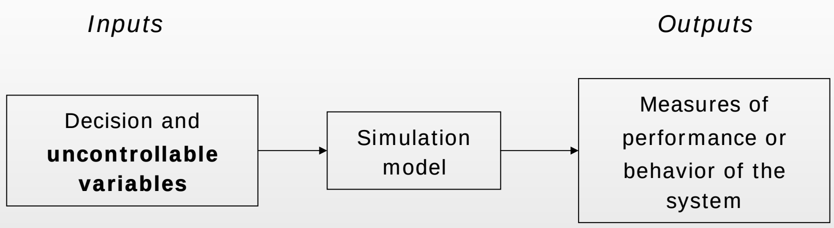
- Models are typically **scaled down**
- Attempt to **capture essential aspects** of the modeled phenomena

Simulations are a **subset of statistics**

- Those that encompass a certain degree of **stochastic modeling**
- Those where certain **parameters** are **unknown** but **can be predicted** based on existing data or **pre-specified assumptions**
- Are relatively dependent on **computing power** – hence have become widely available only over the last decade

- Simulations are especially useful in examining **variation/uncertainty**
  - **Varying the assumptions** of the model provides a natural way to express uncertainty
  - Also allow close examination of **factors that are most important** in causing uncertainty

Simulation Model



Analyzing Stochastic Systems

How does this relate to Decision Trees?

- decision trees work well for decision problems with **clearly defined sequences** of **non-overlapping outcomes**, to which fixed probabilities can be assigned.
- Some problems would be very hard to create a decision tree.

- In some cases, an **analytical solution** is possible, using **mathematics of random variables**.
- In other cases, a **numerical approach** is needed: the numerical approach is called **Monte Carlo Simulation**.

Monte Carlo Simulation Concepts

Monte Carlo Simulation

- For a long time, Monte Carlo, the famous **casino town** in the principality of **Monaco,** was one of the best known venues for roulette and because a fair **roulette wheel** is one of the earliest **random number generators**
- The method was named after that casino town because it based on **random numbers**.
- Birth date of the Monte Carlo method is **1949**, when an article entitled "The Monte Carlo Method" (by N. Metropolis and S. Ulam) appeared.
- The American mathematicians J. Neuman and S. Ulam are considered its originators.

What is Monte Carlo Simulation?

- Monte Carlo simulations provide statistical answers to problems by performing **many calculations** with **randomized variables**, and analyzing the trends in the output data.
- **Mathematical model** of **multiple factors** interacting **simultaneously**
- A **numerical method** for solving mathematical problems using **stochastic sampling**.
- Runs **hundreds** or **thousands of analyses** to find ‘most probable’ answer
- Produces a **simulated probability distribution** that gives a reasonable idea of likely occurrences

What is Monte Carlo Simulation? (cont.)

- It performs simulation of any process whose development is influenced by **random factors**, but also if the given problem involves no chance, the method enables artificial construction of a **probabilistic model**.
- Similarly, Monte Carlo methods **randomly select values** to create **scenarios** of a problem. These values are taken from within a fixed range and selected to fit a **probability distribution** [e.g. bell curve, linear distribution, etc.].
- This is like rolling a dice. The outcome is always within the range of 1 to 6 and it follows a linear distribution - there is an equal opportunity for any number to be the outcome.

Monte Carlo Simulation features

- Use of ***random numbers***
  - ***Sampling***
- City of Monte Carlo ⇒ Roulette
  - **Roulette** is a static game
  - The probability of a certain value is **independent** from the previous value
  - The probability of a certain value is **known a priori** and **equal for each number**

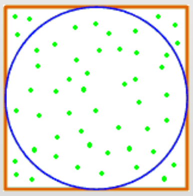
Why use Monte Carlo Simulation?

- It's very difficult *if not impossible* to work out **probability information for outcomes** that depend on **many random variables**
- Simple calculations of statistics are not accurate…
  - Mean × mean = mean
  - 95 % -ile × 95 %-ile ≠ 95 %-ile!
  - Gets "worse" with 3 or more distributions
- Monte Carlo simulation is needed because **closed-form solutions** to many real-world problems are **not possible**. Many previously intractable thermodynamic and quantum mechanics problems have been solved using Monte Carlo techniques
- **Modern computers** make Monte Carlo an approach of choice in this situation.
- **Random numbers** generated by the computer are used to **simulate** naturally random processes

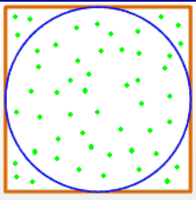
Monte Carlo Simulation Concepts

Example on the **estimation of the area of a circle** even there is a well known formula which is quick and easy to use :

- **Step 1** Draw a square on a piece of paper the length of whose sides are the same as the diameter of the circle.
- **Step 2** Draw a circle in the square such that the centre of the circle and the square are the same.
- **Step 3** Randomly cover the surface of the square with dots, so it looks like this:



Monte Carlo Simulation Concepts

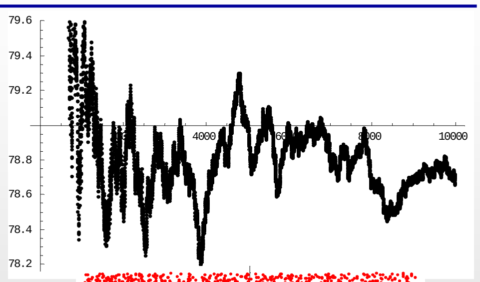


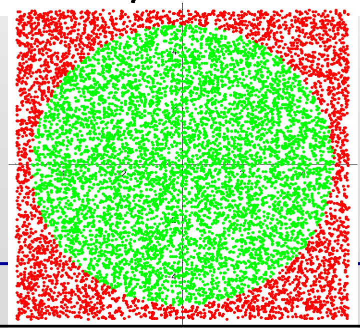
- **Step 4** Count all the dots, then count the ones which fall inside the circle, the area of the circle is estimated thus:

$$Area\,of\,Circle\,(est) = Area\,of\,Square \times \frac{dots\ inside\ circle}{all\ dots}$$

- The larger the number of dots, the greater the accuracy of the estimate.
- By **increasing the number of simulations**, we can **increase the accuracy** and also the time taken to complete the process.

Example: simulation the area of a circle





- Sample points randomly from square surrounding **circle of radius 5**
- **10,000 sample points**
- Acceptance criterion: inside circle
- Actual area: 78.54
- Calculated area: 78.66

Components of Monte Carlo simulation

- ***Probability distribution functions (pdf's)*** - the physical (or mathematical) system must be described by a set of pdf's.
- ***Random number generator*** - a source of random numbers uniformly distributed on the unit interval must be available.
- ***Sampling rule*** - a prescription for sampling from the specified pdf's, assuming the availability of random numbers on the unit interval, must be given.
- ***Variance reduction techniques*** - methods for reducing the variance in the estimated solution to reduce the computational time for Monte Carlo simulation
- ***Parallelization and vectorization*** - algorithms to allow Monte Carlo methods to be implemented efficiently on advanced computer architectures.

Monte Carlo Simulation Method

Monte Carlo Simulation: General Method

• The Monte Carlo simulation is a **converging process**. However, there is no hard and fast rule for selecting **the number of iterations required** for a certain simulation model. The required number of iterations will vary from model to model and depend on the complexity of the model.

• The following steps are recommended to determine the adequate number of iterations for any simulation model:

- Run the simulation start with 200 iterations, generate the CDF for the output and save it.
- Run the simulation again, using say 400 iterations. Compare the CDF for this output to the CDF generated in previous number of trials.
- This process is repeated until a satisfactory match between the current and the previous CDF is obtained, i.e. **further increases in iterations would not yield significant changes in the CDF**.

Sampling Rule

1. Random Sampling
2. Monte Carlo Sampling
3. Latin Hypercube Sampling

Random Sampling Methods

- To simulate process or system with random components, we must generate **set of random numbers**
- After generating **random number**, we then generate **random observations from probability distributions** assigned to each input variable in model
- Computers transform **random numbers** into **random variates** using complex algorithms or from other distributions
- **Random numbers** and **probability distributions** form building blocks of simulation

Random Numbers

Two types of random number:

- **Pseudorandom** numbers are numbers that appear random, but are obtained in a **deterministic, repeatable, and predictable manner**.
- **True random numbers** are generated in **non-deterministic** ways. They are not predictable. They are not repeatable.

• It is **not viable** to generate a **true random number using computers** since they are **deterministic**. However, we can generate a good enough random numbers that have properties close to true random numbers.

Generating Random Numbers

- Spinning disks, electronic randomizers, computers used to generate random numbers
- Tables generated this way
  - Most books on statistics contain table of random numbers
  - Example is Table 6-1 in Mian, vol II, p. 323
- There are sophisticated mathematicians who have worked hard at refining **random number generators**.
- Random number generators most often are designed to return a value between 0 and 1 such as **RAND( )** in Excel.
- A **seed value** is used to start the sequence generation. Some generators allow you to provide the seed, which allows you to regenerate the same sequence, if necessary.

Example random number table

0.03134	0.02472	0.09009	0.33071	0.21537	0.92353
0.39800	0.76542	0.30939	0.92171	0.53607	0.05013
0.40564	0.53449	0.67990	0.12977	0.10273	0.93055
0.55144	0.95666	0.06117	0.66010	0.97143	0.30229
0.86184	0.84805	0.35013	0.59619	0.89659	0.18518
0.12585	0.05650	0.03471	0.02157	0.73789	0.24870
0.70737	0.67381	0.16286	0.51379	0.30397	0.96631
0.58097	0.27832	0.67560	0.88957	0.00526	0.13265
0.12432	0.87536	0.48047	0.24868	0.84931	0.61568
0.07570	0.35596	0.60270	0.91441	0.27316	0.59190
0.87368	0.86953	0.76924	0.09078	0.90977	0.39769
0.11773	0.81101	0.93245	0.78502	0.04384	0.43446
0.13443	0.67273	0.65334	0.37068	0.01132	0.05395
0.19681	0.13410	0.09143	0.70480	0.74590	0.89538
0.23976	0.57713	0.78631	0.45439	0.31639	0.90827
0.60011	0.23851	0.03595	0.00446	0.39770	0.43562
0.37911	0.76155	0.12245	0.40015	0.91815	0.53332
0.03720	0.90763	0.27469	0.00143	0.48567	0.02090
0.65572	0.69865	0.33260	0.95684	0.91302	0.48251
0.01683	0.03898	0.07615	0.33142	0.41718	0.07435
0.95050	0.00944	0.79012	0.60203	0.05900	0.34995
0.18716	0.82710	0.72260	0.20421	0.16923	0.86915
0.97003	0.35187	0.34112	0.18044	0.06201	0.23153
0.49629	0.25173	0.67509	0.05758	0.44831	0.91205
0.03103	0.39139	0.27117	0.82049	0.92390	0.02563
0.75133	0.02358	0.57222	0.20378	0.38917	0.36118
0.06152	0.14712	0.85887	0.61731	0.60548	0.75241
0.98794	0.49891	0.62064	0.98406	0.97554	0.19873
0.18735	0.05743	0.79512	0.15900	0.71377	0.84752
0.06747	0.68036	0.27059	0.09083	0.69116	0.51680
0.37942	0.12261	0.07646	0.74563	0.12631	0.92943
0.80804	0.22580	0.56848	0.13434	0.51776	0.91918

Characteristics of Random Numbers

- Each successive random number in sequence must have **equal probability** of taking on any one of the possible values
- Each number must be **statistically independent** of other numbers in sequence
- Numbers need to be random observations from *uniform distribution* between 0 and 1, referred to as **U(0,1) random numbers**

Random Numbers Between Fixed Limits

Uniformly distributed random numbers between fixed limits A and B can be generated using formula

$$A + x(B - A)$$

where x is uniformly distributed between 0 and 1

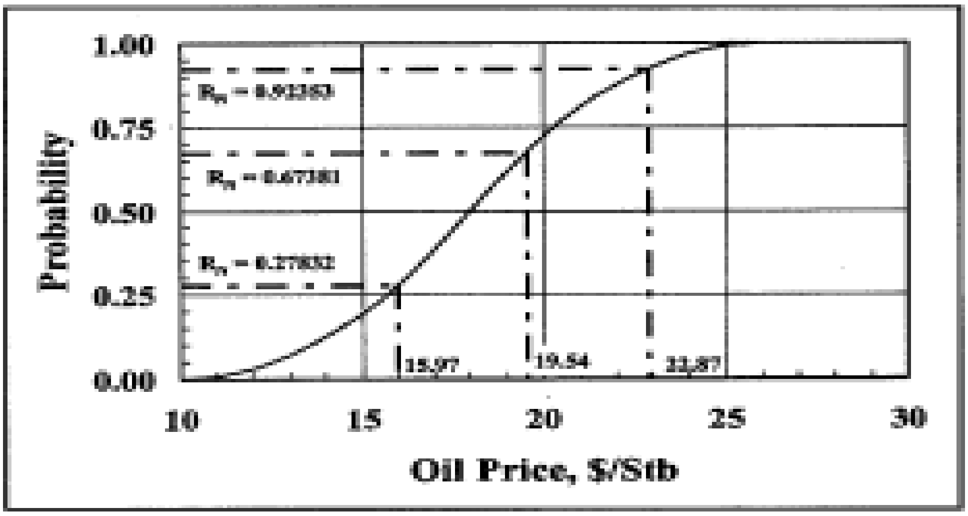
Monte Carlo Sampling

- **Monte Carlo (MC)** random sampling is referred to as **full distribution sampling** of input pdf.
- Each random variable **remains as an element of the distribution**, thus leaving the entire statistical range available for sampling in subsequent iterations.
- In most cases, this result in **clustering of sampling** in some parts of the distribution while other parts are not sampled at all.
- The **transformation** can be done manually by either **graphical approach** or **equation approach** on the **cumulative distribution function (CDF)**.

Manual Monte Carlo Sampling – Graphical

- We need **cumulative distribution function (CDF)** of distribution and **sampling procedure** for uniform distribution on (0,1)
- Following figure illustrates transformation of **random numbers** 0.92353, 0.67381, and 0.27832 into **random variates** (oil price in this case)

Manual Monte Carlo Sampling – Graphical



Manual Monte Carlo Sampling – Analytical

- Graphical method cumbersome for large number of iterations
- Alternative: **Analytical method** requires us to find equation for CDF in form

$$R_N = f(X)$$

and solving for X as function of R_N (i.e., finding inverse function of CDF)

- Method involves **integrating PDF** to **obtain CDF** and then finding inverse of CDF
- Approach practical for some simple distributions (e.g., uniform, triangular, exponential) but not for complex distributions (e.g., normal, lognormal)
  - Complex functions have no simple expression for CDF

Manual Monte Carlo Sampling – Analytical

Sampling from triangular distributions, $T(X_L, X_M, X_H)$

If $R_N \le (X_M - X_L)/(X_H - X_L)$

$$X = X_L + \sqrt{(X_M - X_L)(X_H - X_L)R_N} \qquad (6.1)$$

If $R_N \ge (X_M - X_L)/(X_H - X_L)$

$$X = X_H - \sqrt{(X_H - X_M)(X_H - X_L)(1 - R_N)} \qquad (6.2)$$

Manual Monte Carlo Sampling – Analytical

- Sampling from uniform distribution $U(x_{min}, x_{max})$

$$X = R_N (x_{\max} - x_{\min}) + x_{\min}$$

Manual Monte Carlo Sampling – Analytical

- Sampling from exponential distribution $E(x,\lambda)$
- PDF and CDF given by

$$f(x) = \lambda e^{-\lambda x} \qquad P(X \le x) = 1 - e^{-\lambda x}$$

where

$1/\lambda$ = mean of distribution

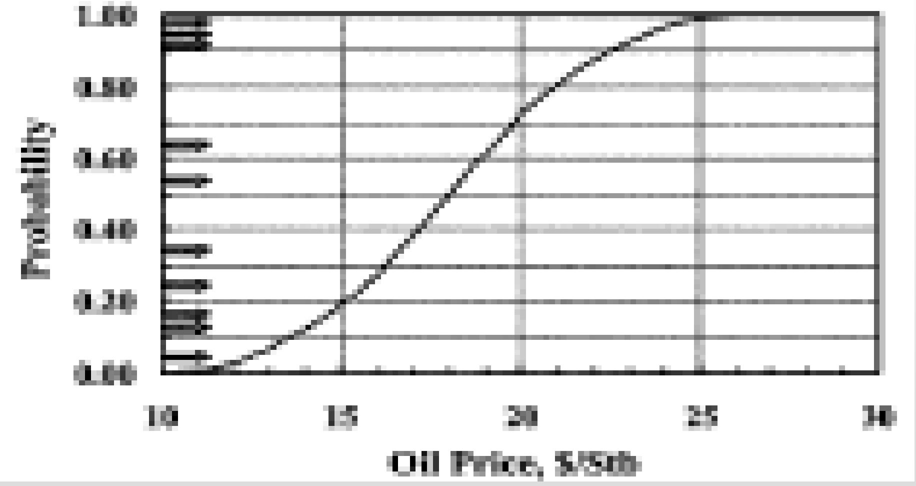
- Inverse function of exponential CDF

$$X = \frac{\ln(1 - R_N)}{-\lambda}$$

Latin Hypercube Sampling

- In **Monte Carlo sampling** (also known as **full distribution sampling**), each random variable remains element of distribution leaving entire statistical range available for sampling in subsequent iterations
  - Often results in clustering of sampling in some parts of distribution, leaving other parts unsampled
- Figure illustrates problem

Latin Hypercube Sampling – Need



Latin Hypercube Sampling – Methodology

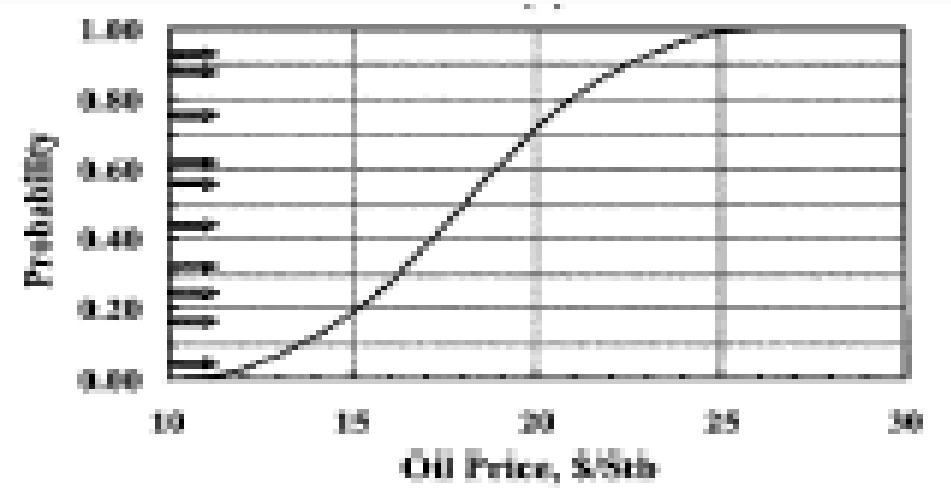
Latin Hypercube Sampling (LHS) is a form of **stratified sampling** that can be applied to multiple variables.

- In **LHS,** cumulative distribution function **partitioned into non-overlapping intervals** of equal probability, in line with number of required iterations
  - Example: if we choose 10 iterations, distribution might be divided into 10 parts
- Random samples then picked from each interval
- The **sampling algorithm** ensures that the distribution function is **sampled evenly**, but still with the same probability trend.
- A probability is randomly picked within **each segment** using a **uniform distribution**, and then mapped to the correct representative value in of the variable's actual distribution.

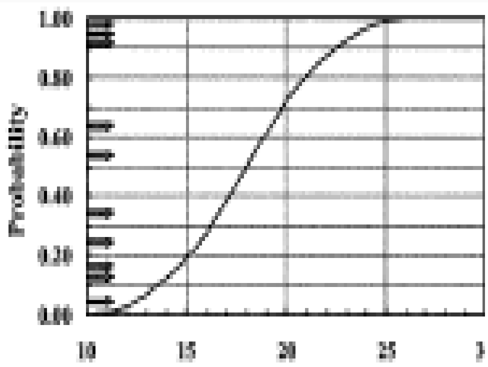
Latin Hypercube Sampling (LHS)

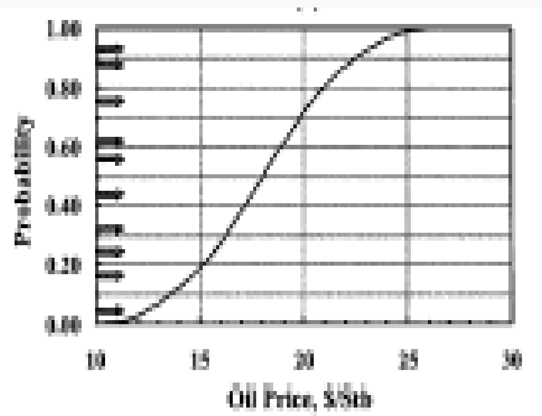
- A simulation with 500 iterations would split the probability into 500 segments, each representing 0.2% of the total distribution.
- For the first segment, a number would be chosen between 0.0% and 0.2%. **Independent uniform selection** is done on each of the variable's generated values.
- For the second segment, a number would be chosen between 0.2% and 0.4%.
- This number would be used to calculate the actual variable value based upon its distribution. **Each value must only be used once**.

Latin Hypercube Sampling – Methodology



MCS, LHS Compared





Latin Hypercube Sampling (LHS)

- The method commonly used to **reduce the number or runs** necessary for a Monte Carlo simulation to achieve a reasonably accurate random distribution. Some simulations may take up to 30% fewer calculations to create a smooth distribution of outputs.
- LHS can be incorporated into an existing Monte Carlo model fairly easily, and work with variables following any analytical probability distribution.

Latin Hypercube Sampling (LHS)

- LHS requires additional overhead to **keep track of the partition sampling without replacement** which may increase **per trial runtime**.
- **CAUTION**: Although LHS has the potential to greatly reduce the number of trials required to achieve the same degree of convergence, it can sometimes provide ***incorrect*** results if used in an improper context.
- Consider modeling a 40-well drilling program with each well having a 0.1 chance of success. If the well result was a unique sampling stream and the number of layer was 10, 20 or 40, then *every trial would have exactly 4 successful wells*. This is obviously unrealistic.

Steps in Monte Carlo Simulation

How do we perform the Monte Carlo Simulation?

Procedure in brief

- Determine **relation** between set of **independent random variables** and one or more **dependent random variables**
- Sample randomly **pdf's of independent variables**
- Calculate repetitively to generate **pdf's of dependent variables**

How do we perform the Monte Carlo Simulation?

- Identify **'uncertain' variables**
- Distinguish **'dependent'** from **'independent' variables**
- Choose **'representative' probability curve** for each 'uncertain' variable
- Probability distributions must **describe range of likely values** for each parameter of interest
- Distributions may be **standard forms** (e.g., normal or lognormal) or **intuitive** (e.g., triangular, rectangular)

Steps in Simulation Modeling – Evaluate Probability Distributions

Methods to determine probability distribution

- Fit **theoretical distribution** to **historical data** (for example: from reservoir or analogous reservoir)
- From **experience of analyst**
- **Heuristic approach** – use "rule of thumb" believed to be appropriate

Steps in Simulation Modeling – Evaluate Probability Distributions

- **Parameters of distribution** may be chosen on basis of studies of geological properties in area or analogous area
- In absence of data, **normal** or **lognormal distributions** likely appropriate for distributions of geological properties
- Some properties must be bounded (to avoid porosities and saturations outside range 0 to 100%, for example)

Example : Manual

Example Simulation – Manual

- Even though we use computers to simulate, manual calculation illustrates process
- Simulate simple case with three variables
  - Oil reserves, MSTB
  - Oil price, $/STB (net of OPEX)
  - Tax rate of 40%

Example Simulation – Manual

- Oil reserves represented by uniform distribution of parameters U(100,250)
  - Minimum 100 MSTB
  - Maximum 250 MSTB
- Tax rate, 40%, constant
- Oil price represented by triangular distribution with parameters T(10,18,26)
  - Minimum $10/STB
  - Most likely $18/STB
  - Maximum $26/STB

Example Simulation – Manual

- Expected value of NCF is product of expected values of each parameter from its distribution
- Expected value of uniform distribution

$$E(X_U) = \frac{(x_{\min} + x_{\max})}{2} = \frac{100 + 250}{2} = 175$$

- Expected value of triangular distribution

$$E(X_T) = \frac{(X_L + X_M + X_H)}{3} = \frac{10 + 18 + 26}{3} = 18$$

- Expected value of net cash flow

$$E(NCF) = (175 x 18)(1 - 0.4) = 1{,}890\, M\,\$$$

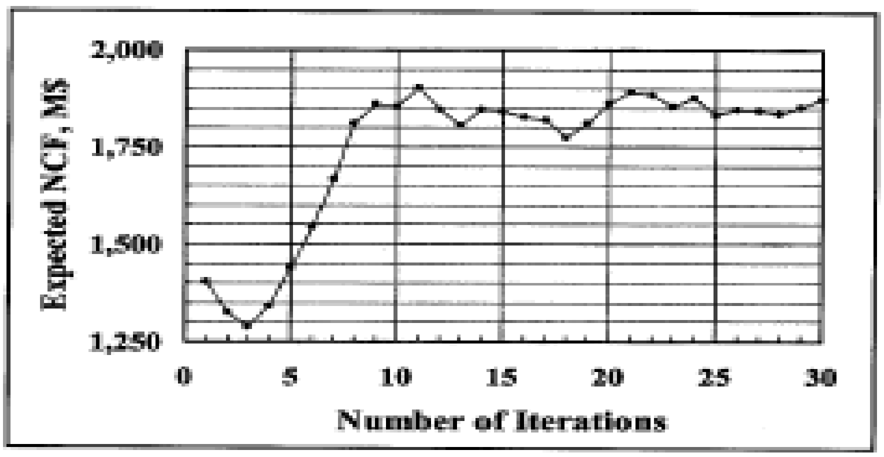
Example Simulation – MC Sampling

		Distribution	Parameters: Low	Parameters: High	Parameters: Most Likely
Reserves	M Stb	Uniform	100	250	
Oil Price	$/Stb	Triangular	10	26	18
Tax Rate	40.0%				
Iteration	**Random Number**	**Reserves (M Stb)**	**Random Number**	**Oil Price ($/Stb)**	**Net Cash Flow (M$)**
1	0.03134	104.701	0.92353	22.871	1,436.79
2	0.39800	159.700	0.05013	12.533	1,200.92
3	0.40564	160.846	0.93055	23.018	2,221.46
4	0.55144	182.716	0.30229	16.220	1,778.23
5	0.86184	229.276	0.18518	14.869	2,045.40
6	0.12585	118.878	0.24870	15.642	1,115.70
7	0.70737	206.106	0.96631	23.923	2,958.45
8	0.58097	187.146	0.13265	14.121	1,585.56
9	0.12432	118.648	0.61568	18.986	1,351.61
10	0.07570	111.355	0.59190	18.772	1,254.25
				E[NCF] =	1,694.84

Example Simulation – LHC Sampling

			Parameters			
		Distribution	Low	High	Most Likely	
Reserves	M Stb	Uniform	100	250		
Oil Price	$/Stb	Triangular	10	26	18	
Tax Rate	40.0%					
Iteration	Random Number	Reserves (M Stb)	Random Number	Oil Price ($/Stb)	Net Cash Flow (M$)	NCF Average (M$)
1	0.03134	104.701	0.89538	22.341	1,403.45	1,403.45
2	0.12585	118.878	0.43446	17.457	1,245.17	1,324.31
3	0.23976	135.964	0.18518	14.869	1,212.95	1,287.19
4	0.39800	159.700	0.24870	15.642	1,498.83	1,340.10
5	0.40564	160.846	0.61568	18.986	1,832.32	1,438.54
6	0.55144	182.716	0.59190	18.772	2,058.02	1,541.79
7	0.65572	198.358	0.75241	20.370	2,424.39	1,667.87
8	0.70737	206.106	0.92353	22.871	2,828.35	1,812.93
9	0.86184	229.276	0.30229	16.220	2,231.37	1,859.43
10	0.95050	242.575	0.05013	12.533	1,824.13	1,855.90

Example Simulation – LHS Results



Example: Reserve

Monte Carlo Simulation: Application

Procedure applied to “School Prospect”

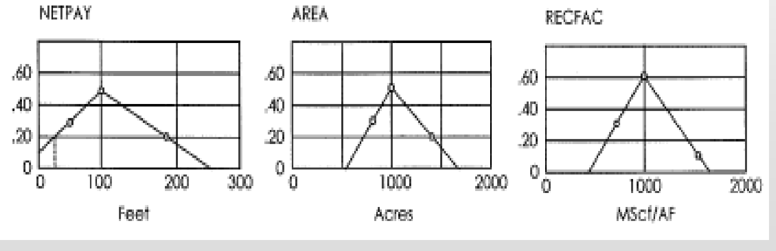
- Establish **algorithm** to **calculate dependent variable**
  - **ULTGAS = NETPAY x AREA x RECFAC**
- Establish **correlations** between **independent variables**
  - Assume no correlations in this example

What will simulator do next?

- Generate **probability distribution** curves for each variable
- Identify **correlations** between variables and take correlations into account
- Choose **random variables** from each curve
- Generate **CDF** based on **PDFs for each variable**

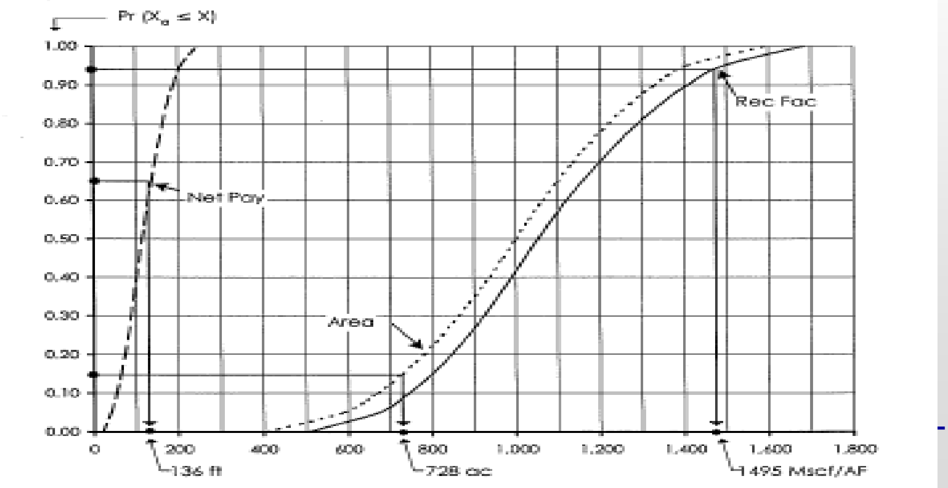
Monte Carlo Simulation: Application

- Establish **pdf's for each independent variable**
  - Assume triangular distributions, using minimum, most likely, maximum values
  - Net pay distribution truncated at 20 ft



Monte Carlo Simulation: Application

- Calculate cdf's for each independent variable



Monte Carlo Simulation: Application

- Generate random number between 0.0 and 1.0 and determine **value for NETPAY** that corresponds to this cumulative probability
  - If random number were 0.65, NETPAY would be 136 ft (see previous figure)

Monte Carlo Simulation: Application

- Generate new random number between 0.0 and 1.0 and determine **value for AREA** that corresponds to this cumulative probability
  - If random number were 0.15, AREA would be 728 acres (see previous figure)

Monte Carlo Simulation: Application

- Generate new random number between 0.0 and 1.0 and determine **value for RECFAC** that corresponds to this cumulative probability
  - If random number were 0.95, RECFAC would be 1495 Mscf/AF (see previous figure)

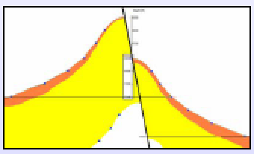
Monte Carlo Simulation: Application

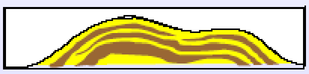
- **Compute value for ULTGAS**
  - ULTGAS = NETPAY x AREA x RECFAC
    = (136)x(728)x(1495)
    = 148 Bscf = ULTGAS(1)
- Repeat and compute ULTGAS(2), ULTGAS(3), ULTGAS(4), … , ULTGAS(n) until enough values are available to **define pdf of ULTGAS** (hundreds or thousands of iterations)
- Latin hypercube sampling more efficient than Monte Carlo sampling, when available

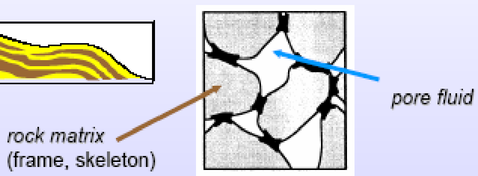
Monte Carlo Simulation: Volumetric Equation

STOIIP = GRV * Net/Gross Ratio * Porosity * oil saturation * (1/Bo)

GIIP = GRV * Net/Gross Ratio * Porosity * gas saturation * (1/Bg)





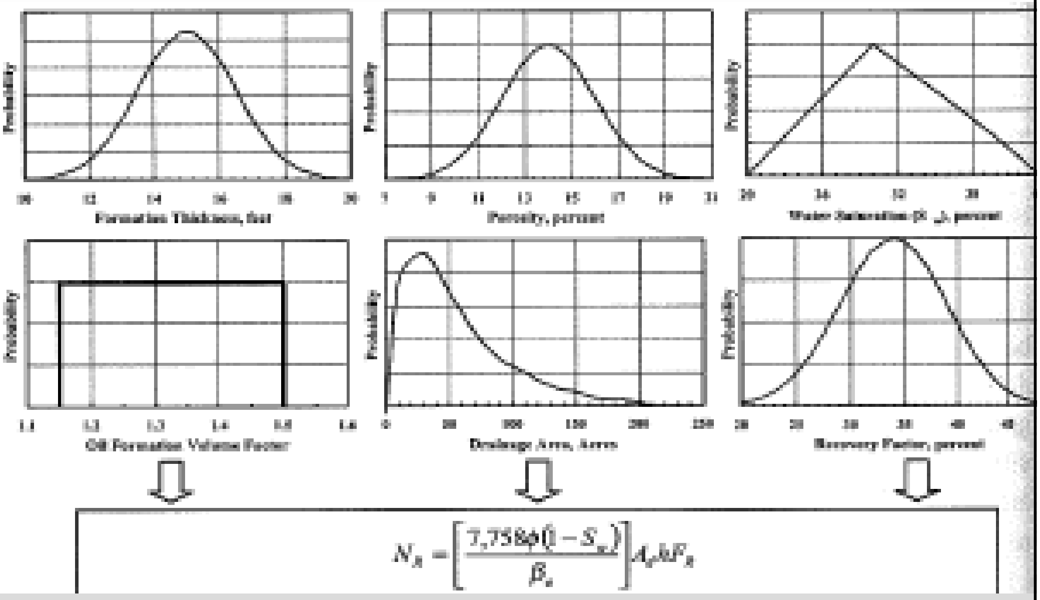


Recoverable oil= STOIIP * Recovery Factor for oil

Recoverable gas= GIIP * Recovery Factor for gas

Where GRV = Gross Rock Volume

Steps in Simulation Modeling – Reserves



How can we do all that?

- Identify variables

Net pay	Area	Porosity
Formation volume factor	Water saturation	Recovery factor

How can we do all that?

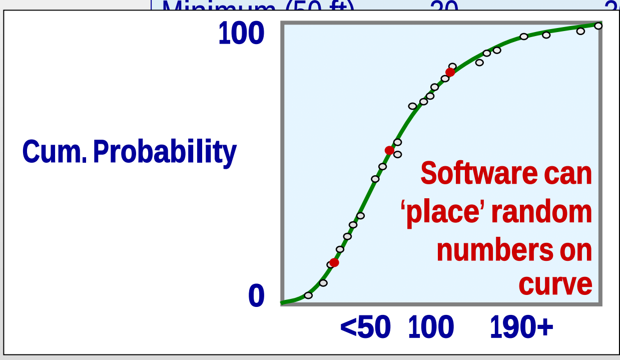
- Identify variables
- Distinguish ‘dependent’ variables

Net pay ↔	Area	Porosity
Formation volume factor	Water saturation ↔	Recovery factor

How can we do all that?

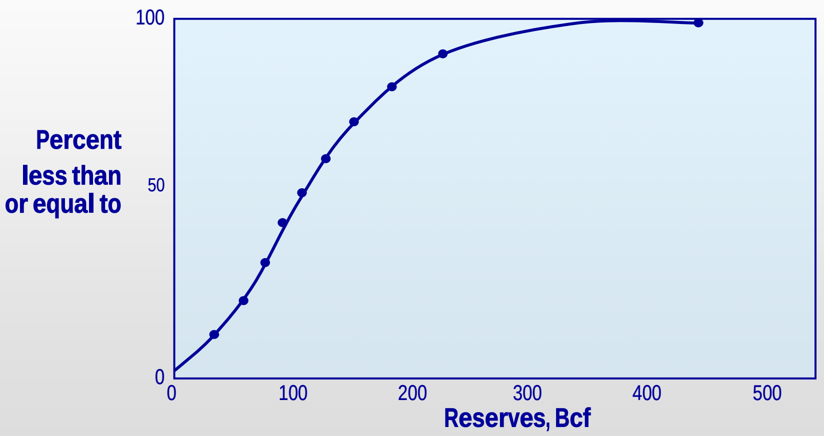
- Choose probability curve for each variable

Net Pay Occurrence	Probability, %	Cum. Prob.	Random # Ranges
Minimum (50 ft)	30	30	00-30
[illegible]	[illegible]	0	31-80
[illegible]	[illegible]	00	81-99



Choose random numbers from continuous functions

What will the result be?



Dependencies in Simulation

- Broad types of dependencies
  - **Correlations between input variables** used to calculate pdf's of desired results (e.g., OIP or reserves for specific accumulation)
  - **Correlations between calculations of desired results** (e.g., OIP or reserves for accumulations that are aggregated)

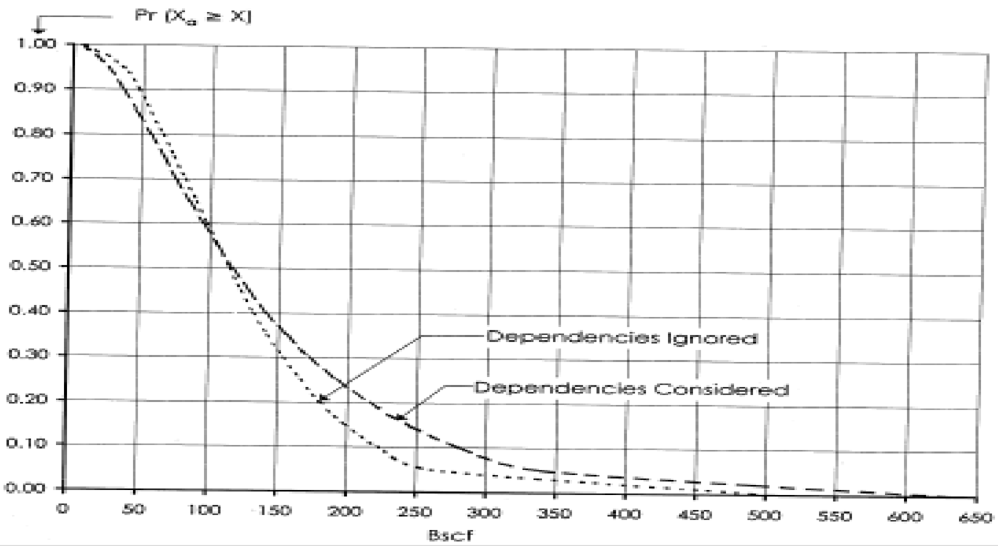
Spearman Rank Correlations

- We can use **Spearman rank correlations (SRC)** to detect dependencies between pairs of input parameters
- **SRC tests** for correlation between data sets based on relative *rankings* of elements in data sets rather than on *values* of elements
- SRC ranges from -1.0 (perfect negative correlation) to +1.0 (perfect positive correlation)

Correlations: Example Calculation

- If we examined “School Prospect” using MCS and assuming SRC of 0.5 between AREA and NETPAY and between RECFAC and NETPAY
- Software automatically adjusted input parameters so that output parameters were sampled at rank comparable to that at which input parameters were sampled

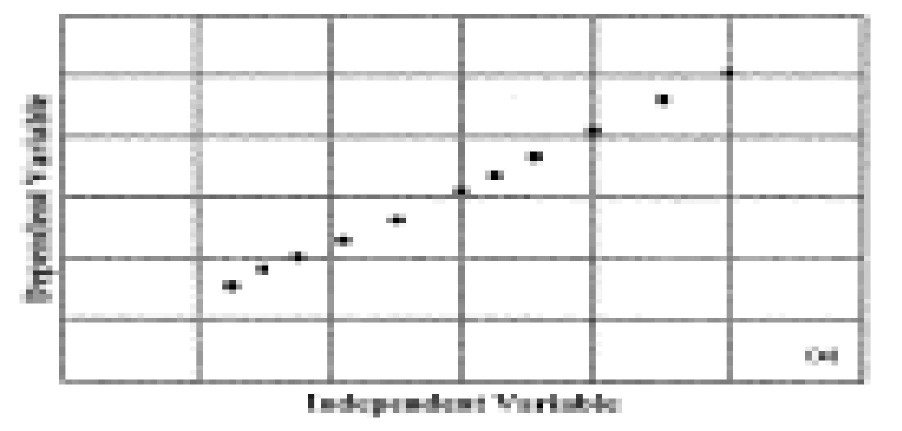
Correlations: Example Calculation



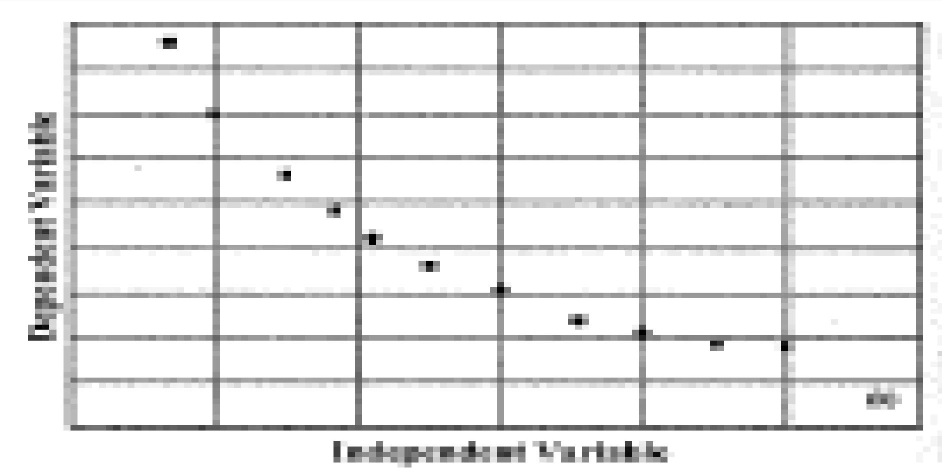
Identifying Correlations Between Variables

- When input variables are correlated and we assume them to be independent, our simulation results will be incorrect
- We must identify **correlations** and take them into account
- **Scatter plots** help us identify correlations

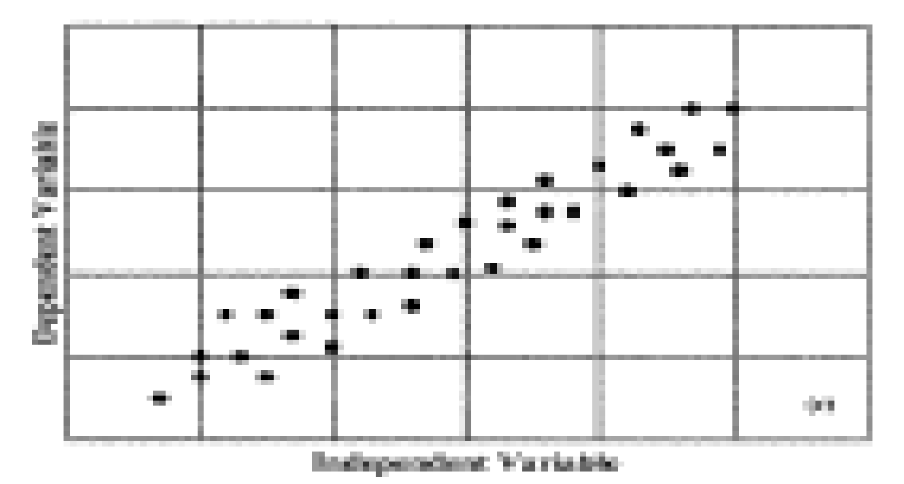
Total Linear Positive Dependence



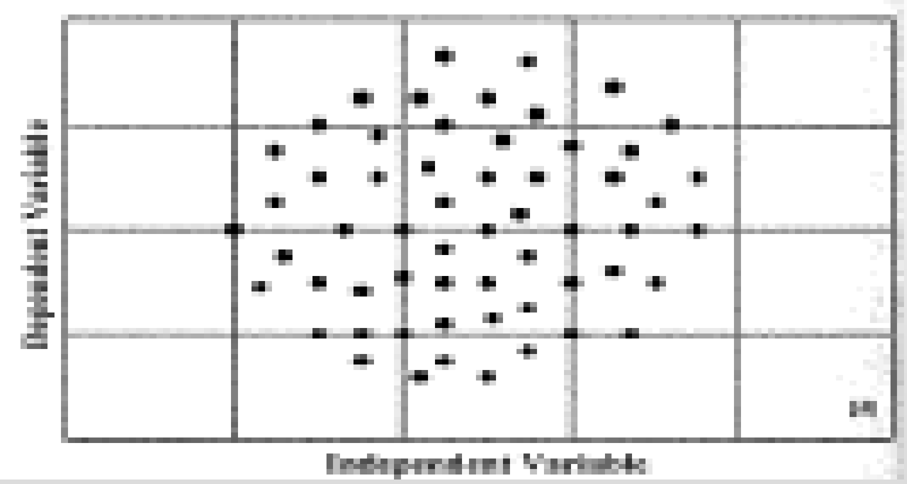
Total Nonlinear Negative Dependence



Diffuse Positive Dependence



Uncorrelated Variables



Quantifying Dependence: Correlation Coefficient

- Correlation coefficient, ***r***, reveals degree of correlation between variables

$$r_{XY} = \frac{\dfrac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{(n-1)}}{s_X s_Y}$$

where

s_X = standard deviation of *X*'s

s_Y = standard deviation of *Y*'s

$\overline{X}$ = average (mean) of X's

$\overline{Y}$ = average (mean) of Y's

Interpretation of Correlation Coefficients

Values of *r* range between -1 and +1

- Value near +1 means strong positive correlation
- Value near -1 means strong negative correlation
- Value near 0 means little or no correlation

Linear regression analysis quantifies relationship between dependent variable *Y* and independent variable *X*, *Y* = a +b*X*

Using Excel to Determine Correlation Coefficient

- Use CORREL function

    =CORREL(X-range,Y-range)

OR

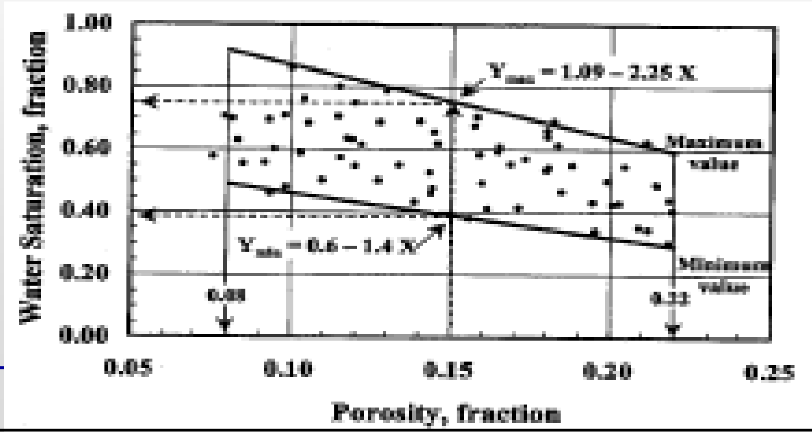
- Use **Tools**⇨ **Data Analysis**⇨**Regression**
    - Provides table of correlations and other parameters

Simulating Total Dependence

- When **correlation coefficient (r)** near 1, we assign same random number for *X* and *Y* in simulation
- When correlation coefficient near -1, use original random number, ***x***, to generate value for independent variable and **1.0** – ***x*** to generate value for dependent variable
    - Sample independent variable, *X*, first
    - Value of *X* influences value of dependent variable *Y* by restricting its range
- Alternative: Use relationship $Y = a + bX$

Simulating Diffuse Dependence

- Data in example figure shows downward sloping trend with *r* = -0.5344
- Similar situations arise frequently in practice



Simulating Diffuse Dependence: Methodology

- Prepare cross plot of random variables X and Y
- Draw box around data points so majority is bounded, maximum and minimum limits defined
- Identify type of variation of Y within box as function of X – random, clustered midway, clustered at upper or lower boundary

Simulating Diffuse Dependence: Methodology

- Generate normalized *Y* distribution
  - For each *X*, unique distribution of *Y* between Y_{min} and Y_{max} exists, and is conveniently represented in terms of normalized Y variable

$$Y_{NORM} = \frac{Y - Y_{\min}}{Y_{\max} - Y_{\min}}$$

- For given iteration, value of *X* selected randomly, and value of *Y* from normalized distribution corresponding to value of *X* selected

Simulating Diffuse Dependence: Methodology

- Develop cumulative probability distribution for independent variable *X* and normalized *Y* from previous step
- Generate random numbers and sample distributions
  - Generate two random numbers, R_{N1} and R_{N2}
  - Use R_{N1} to sample *X* distribution
  - Use R_{N2} to sample Y_{NORM} distribution
  - Values of X_1 and Y_{NORM1} result

Simulating Diffuse Dependence: Methodology

- Obtain Y_{min} and Y_{max} corresponding to X_1 from figure or, better, from fitting equations
- Calculate Y_1 from Y_{NORM1}, Y_{max}, and Y_{min}
- Repeat for each iteration at fixed value of X_1
- Select X_2, … randomly and repeat entire process
- Process illustrated in **Example 6-1**, Mian, pp. 342 - 347

Simulation Using Excel

Generating random numbers

- Use formula **= RAND()** in any cell
- Properties
  - Whenever function is used, numbers between 0 and 1 have same chance of occurring – numbers will be ***uniformly distributed***
  - Numbers ***probabilistically independent*** – when one random number generated, we obtain no information about subsequent random numbers

Simulation Using Excel

Inverse of probability distributions in Excel (built-in functions)

- **=BETAINV()** – returns inverse of cumulative beta function distribution
- **=CHIINV()** – returns inverse of one-tailed probability of chi-squared distribution
- **=FINV()** – returns inverse of F probability distribution
- **=GAMMAINV()** – returns inverse of gamma probability distribution
- **=LOGINV()** – returns inverse of lognormal probability distribution
- **=NORMINV()** – returns inverse of normal cumulative probability distribution
- **=NORMSINV()** – returns inverse of standard normal cumulative probability distribution
- **=TINV()** – returns inverse of student's t-distribution

Simulation Using Excel

- Excel's built-in **inverse probability distribution functions** all have "probability" in argument
  - Example: **=NORMINV(probability,μ,σ)**
- We can replace **probability** with **random number** in simulation
  - Example: **=NORMINV(RAND(),μ,σ)** generates random variate of normal distribution

Simulation Using Excel

Excel has other built-in functions that **generate pdf and cdf but not inverse**

- **BINOMDIST()** – binomial distribution
- **EXPONDIST()** – exponential distribution
- **HYPEGEOMDIST()** – hypergeometric
- **POISSON()** – Poisson distribution
- **WEIBULL()** – Weibull distribution

We can determine inverses of CDF's of Excel's functions with **no built-in inverse** by using =VLOOKUP function

- Table 6-6, page 350 of Mian illustrates use of **=VLOOKUP function**

Using VLOOKUP Function

	A	B	C	D	E	F	G
1							
2	0.0000	0.00	Cell A2: = EXPONDIST(B2,0.85,TRUE)				
3	0.3462	0.50	Copy A2 from A3:A13				
4	0.5726	1.00	Enter random variables in B2:B13				
5	0.7206	1.50					
6	0.8173	2.00	Cell A15: =VLOOKUP(RAND(),A2,B13,2)				
7	0.8806	2.50					
8	0.9219	3.00	Press F9 and see how the value in Cell A15				
9	0.9490	3.50	changes each time F9 is pressed				
10	0.9666	4.00					
11	0.9782	4.50					
12	0.9857	5.00					
13	0.9907	5.50					
14							
15	1.5000						
16							

Example: Reserves Simulation

$$N_R = \frac{7,758\,\varphi(1 - S_w)hA_d F_R}{B_o}$$

- **Porosity:** normal distribution $\varphi(14,2)$
- $\boldsymbol{S_w}$: triangular distribution $S_w(20,30,44)$
- $\boldsymbol{h}$: normal distribution $h(15,1.5)$
- $\boldsymbol{A_d}$: lognormal distribution $A_d(77,63)$
- $\boldsymbol{F_R}$: normal distribution $F_R(34,5)$
- $\boldsymbol{B_o}$: uniform distribution $B_o(1.15,1.5)$

Example Simulation with Excel: Ex. 6-2, Mian

Calculate volumetric oil reserves

- **Porosity:** normally distributed, mean = 0.14, standard deviation = 0.02
- **Water saturation:** triangular, min, most likely, max values 0.2, 0.3, 0.44
- **Formation thickness:** normally distributed, mean = 15 ft, std. dev. = 1.5 ft
- **Drainage area:** normally distributed, mean = 77 acres, std. dev. = 63 acres (careful – can cause ***negative areas*** in simulation)
- **Recovery factor:** normally distributed, mean = 0.34, std. dev. = 0.05
- **Oil FVF:** uniform distribution, parameters 1.15 and 1.5

Example Simulation with Excel: Ex. 6-2, Mian

	A	B	C	D	E	F	G	H	I	J
1										
2										
3				**Distribution**		**Parameters**				
4	Porosity		Fraction	Normal	0.14	0.02				
5	Water Saturation		Fraction	Triangular	0.20	0.30	0.44			
6	Formation Thickness		Feet	Normal	15	1.50				
7	Drainage Area		Acres	Normal	77	63.00				
8	Recovery Factor		Fraction	Normal	0.34	0.05				
9	Oil FVF		RB/STB	Uniform	1.15	1.50				
10	Average Reserves		STB	**211,567**						
11		Porosity	Sw	Thickness	Area	RF	Oil FVF	Reserves		
12	No.	(fraction)	(fraction)	(feet)	(acres)	(fraction)	(RB/STB)	(STB)		
13	1	0.1388	0.2643	15.6572	84.8440	0.3227	1.4710	230,867		0.172190
14	2	0.1615	0.3899	15.1491	148.4853	0.3260	1.2449	450,324		0.925371
15	3	0.1485	0.2992	14.7222	111.8553	0.3829	1.4159	359,589		0.409762
16	4	0.1485	0.3050	13.8724	83.5089	0.3144	1.2608	231,265		0.457940

Steps in Setting Up Spreadsheet for Ex. 6-2

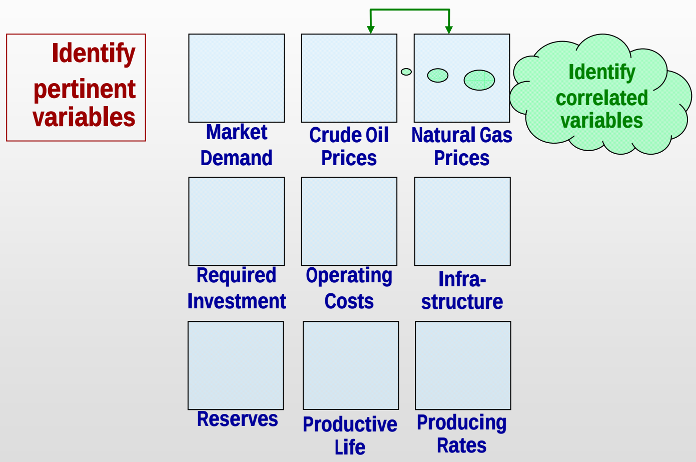
- Enter **inputs for probability distributions** in cells E4 to G9
- Cell B13 =NORMIN(RAND(),E$4,F$4)
- Cell C13 =IF(J13<=((F$5-E$5)/(G$5-E$5)),E$5+SQRT((F$5-E$5)*J13),G$5-SQRT((G$5-F$5)*(G$5-E$5)*(1-J13)))
  Formula refers to Eqs. 6.1,6.2.
- Cell refers to random numbers in cell J13.
- Enter =RAND() in cell J13 and copy down to cell J550.

Steps in Setting Up Spreadsheet for Ex. 6-2

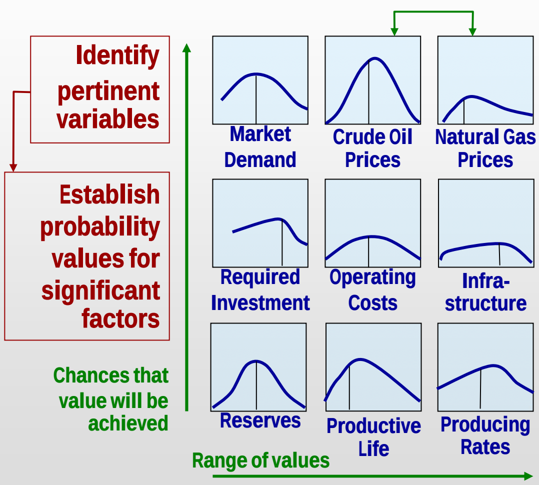
- Cell D13 =NORMINV(RAND(),E$6,F$6)
- Cell E13 =NORMINV(RAND(),E$7,F$7)
- Cell F13 =NORMINV(RAND(),E$8,F$8)
- Cell G13 =RAND()*(F$9-E$9)+E$9 (Eq. 6.3)
- Cell H13 =((7758*B13*(1-C13)*D13*E13)/G13)*F13 … volumetric reserve equation
- Copy cells B13:H13 to cells B550 to H550, providing 538 iterations for simulation
- Cell D10 =AVERAGE(H13:H550) … calculates average reserve for 538 iterations

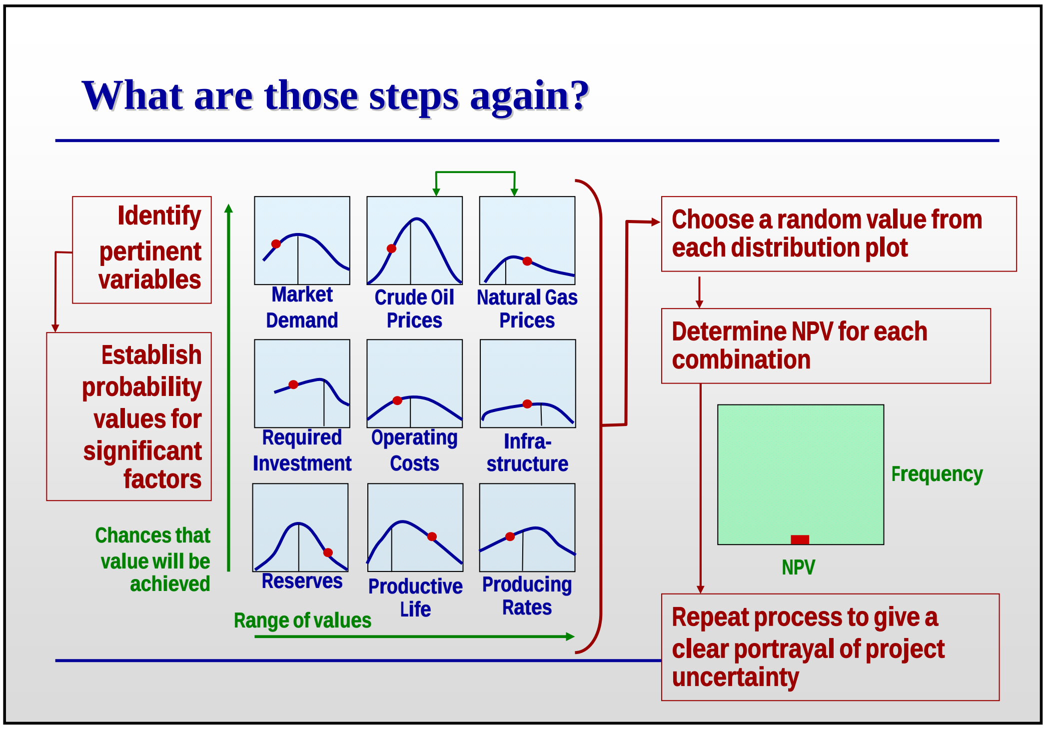
Example: NPV

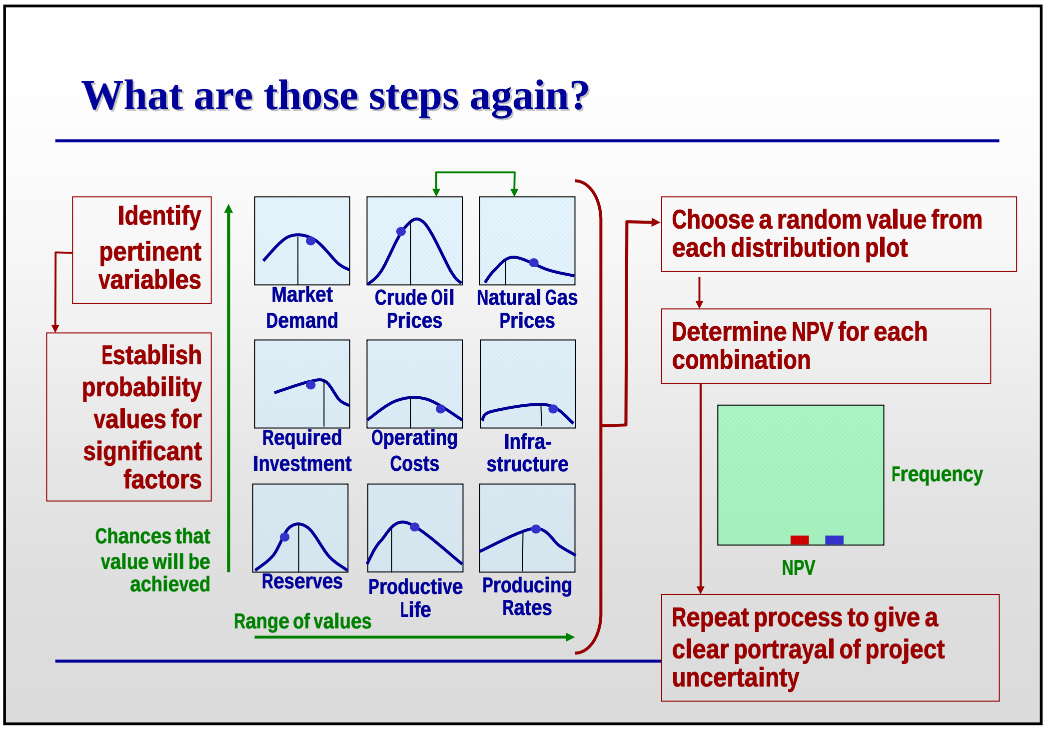
Another Example: NPV Distribution

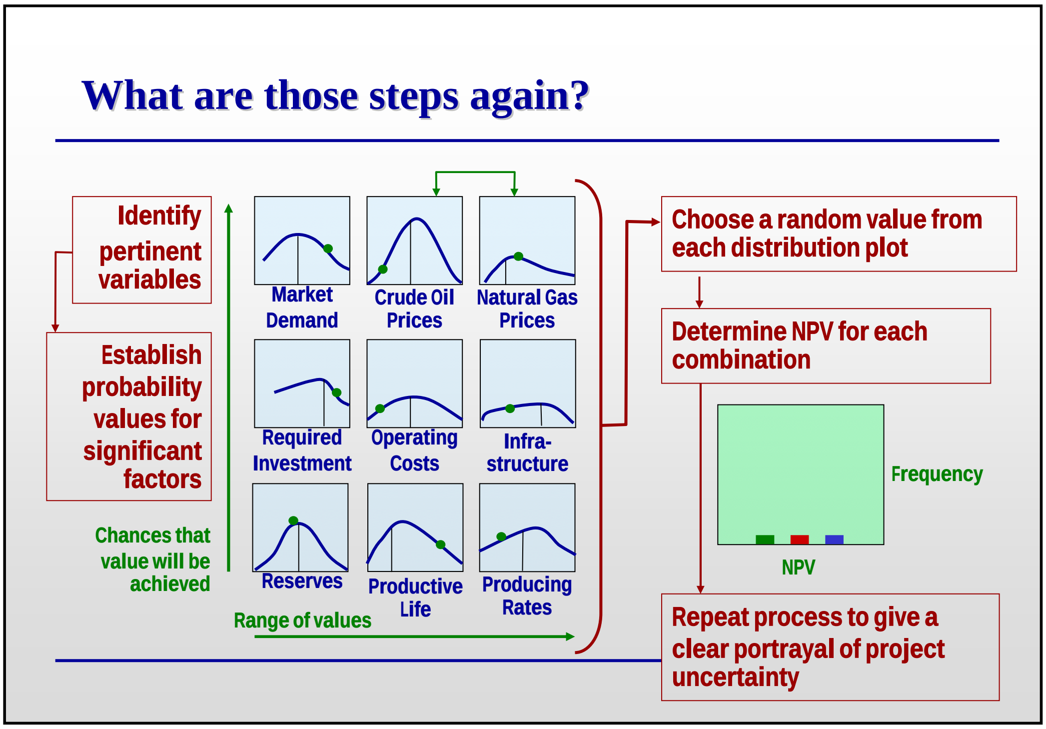


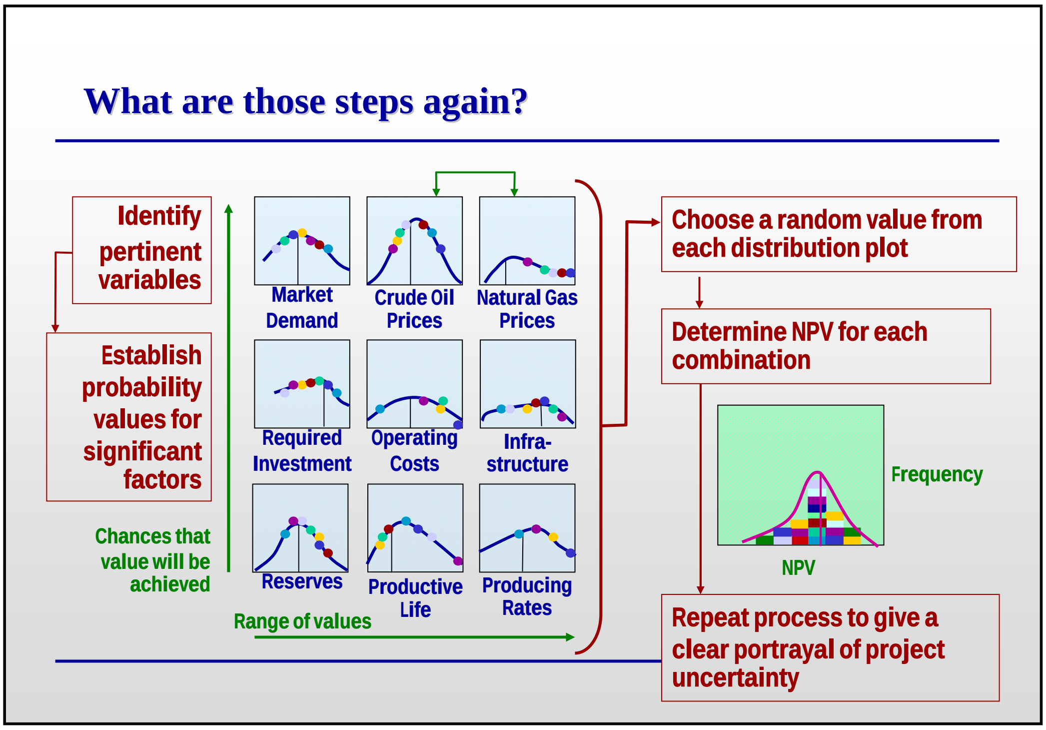
What are those steps again?




What are those steps again?
Identify pertinent variables
Establish probability values for significant factors
Chances that value will be achieved
Market Demand
Crude Oil Prices
Natural Gas Prices
Required Investment
Operating Costs
Infra-structure
Reserves
Productive Life
Producing Rates
Range of values
Choose a random value from each distribution plot
Determine NPV for each combination
Frequency
NPV
Repeat process to give a clear portrayal of project uncertainty


What are those steps again?
Identify pertinent variables
Establish probability values for significant factors
Chances that value will be achieved
Market Demand
Crude Oil Prices
Natural Gas Prices
Required Investment
Operating Costs
Infra-structure
Reserves
Productive Life
Producing Rates
Range of values
Choose a random value from each distribution plot
Determine NPV for each combination
Frequency
NPV
Repeat process to give a clear portrayal of project uncertainty


What are those steps again?
Identify pertinent variables
Establish probability values for significant factors
Chances that value will be achieved
Market Demand
Crude Oil Prices
Natural Gas Prices
Required Investment
Operating Costs
Infra-structure
Reserves
Productive Life
Producing Rates
Range of values
Choose a random value from each distribution plot
Determine NPV for each combination
Frequency
NPV
Repeat process to give a clear portrayal of project uncertainty


What are those steps again?
Identify pertinent variables
Establish probability values for significant factors
Chances that value will be achieved
Market Demand
Crude Oil Prices
Natural Gas Prices
Required Investment
Operating Costs
Infra-structure
Reserves
Productive Life
Producing Rates
Range of values
Choose a random value from each distribution plot
Determine NPV for each combination
Frequency
NPV
Repeat process to give a clear portrayal of project uncertainty

Is Monte Carlo our best option?

- Handles **large numbers of variables**, data points
- Demonstrates **range of possibilities** as opposed to **single ‘answer’**
- Describes **potential**, **risks** better than hand calculations
- Uses **widely available computer software** such as:
  - **@RISK**
  - **Crystal Ball**

How can we manage risk?

- **Avoidance (walk away)**
- **Reduction**
  - Walk away
  - Reduce scope of venture
  - Decide to accept loss
- **Transfer**
  - Add partners
  - Spread over time