SUNGARD®

DOING MORE WITH LESS: GETTING BETTER VALUE OUT OF YOUR CURRENT DATA

Neil Covington
Director of Solutions Management P&C

Advance.

Contents

SUNGARD®

01

Background

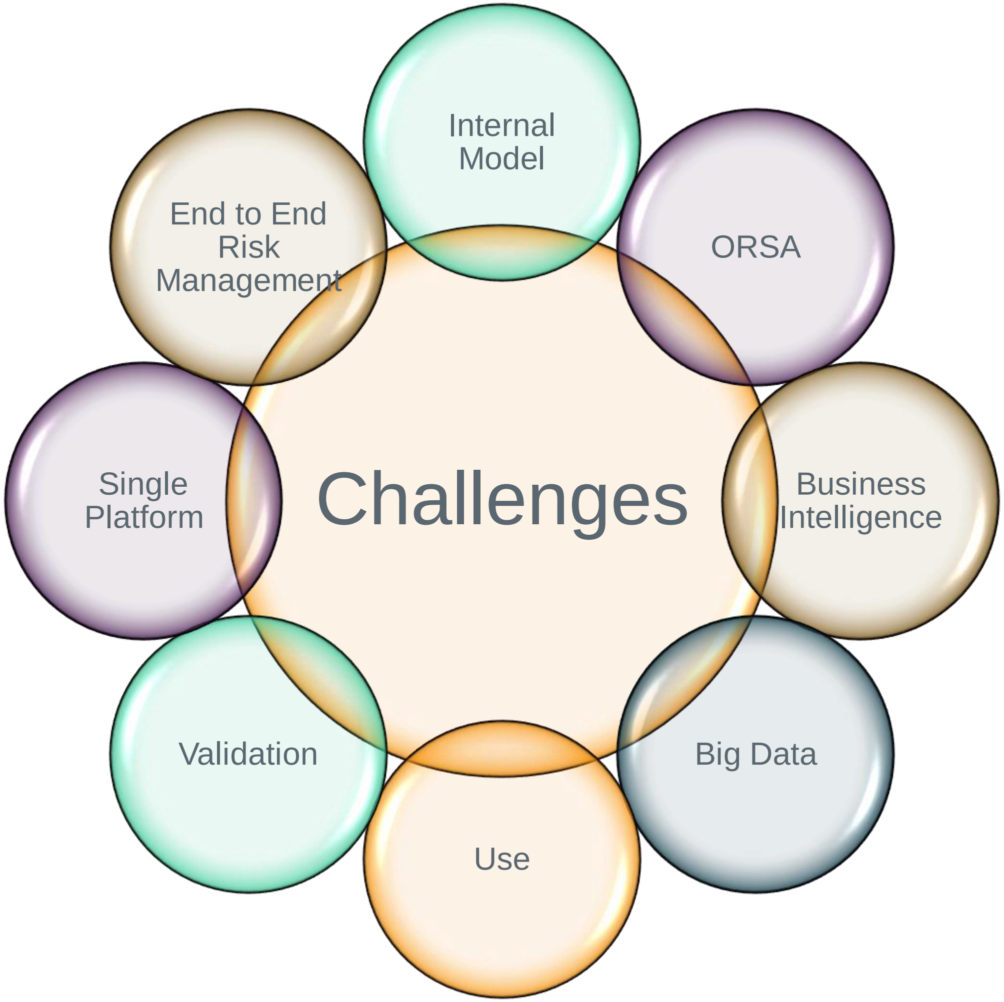
Current Environment

Do more

Do it faster

Do it cheaper

Challenges



SUNGARD®

Possible Solutions

Ideas

High level, not technical

Options

SUNGARD®

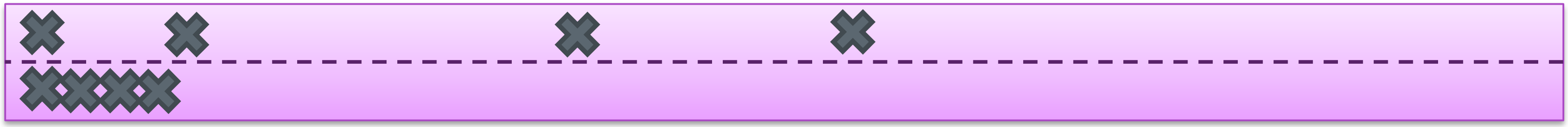
02

Stratified Sampling

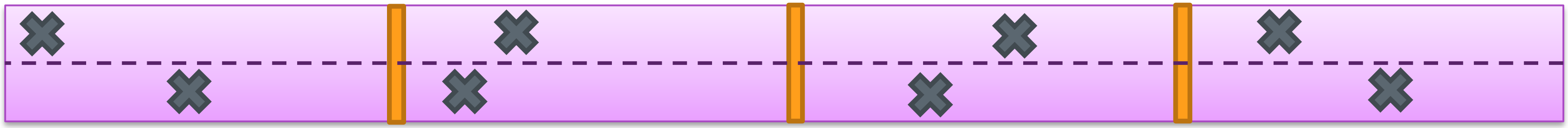
Stratified Sampling

- Monte Carlo sampling is random sampling across the full probability space
- Stratified sampling segments the probability space and provides quicker convergence to the underlying distribution
- Consider a uniform distribution, 4 simulations, 2 runs

Monte Carlo random sampling



Stratified Sampling

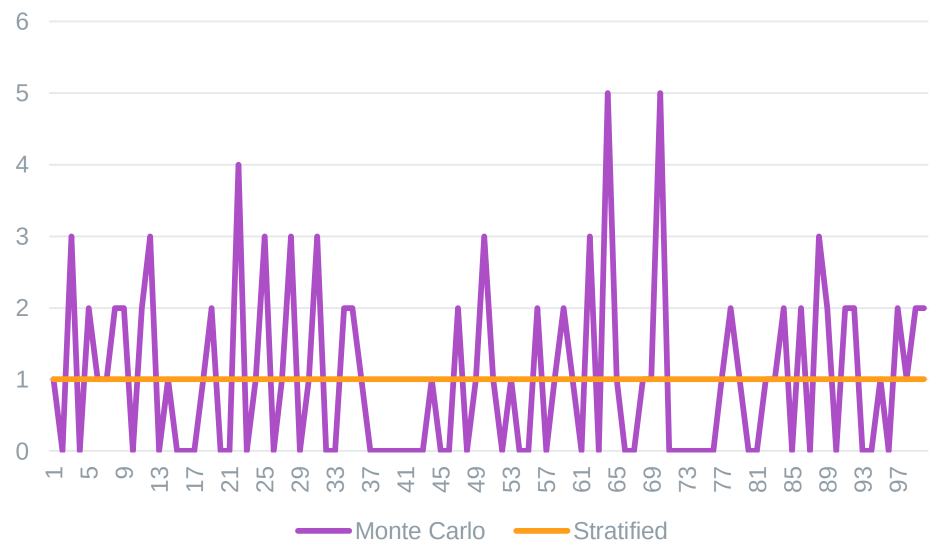


Latin Hypercube is a multi-dimensional extension

Stratified Sampling

Extreme Example

- Consider a uniform discrete distribution on the number 1 to 100 inclusive
- Consider 100 samples from this distribution
- Monte Carlo sampling would randomly pick from these
- Stratified sampling with 100 strata would select each value once and once only



SUNGARD®

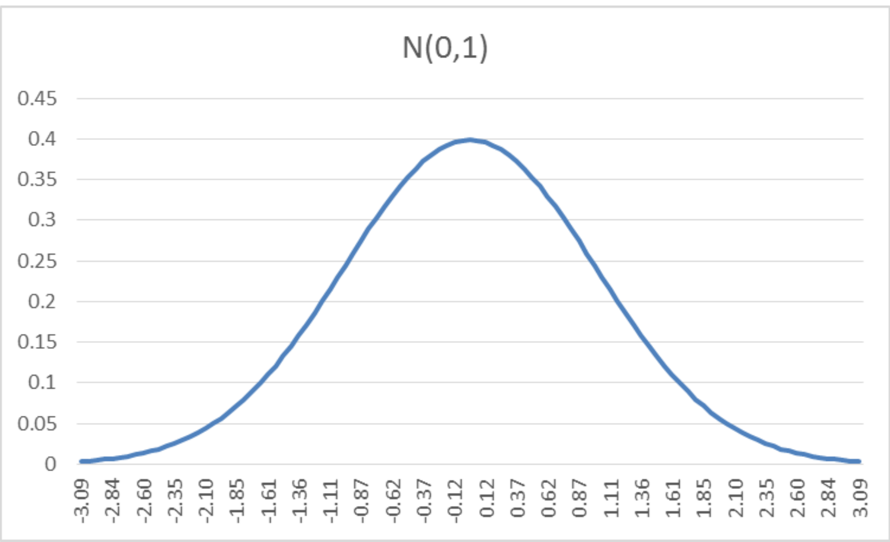
Mean Convergence Examples

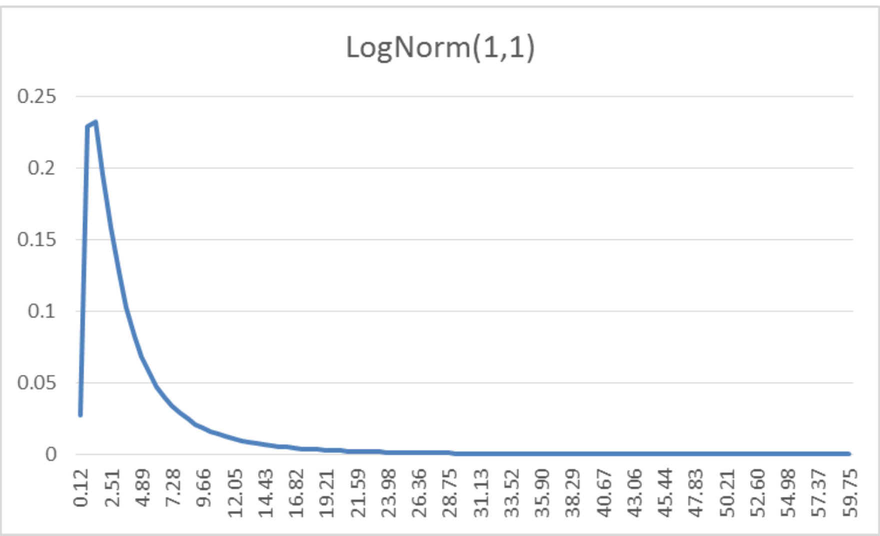
Consider the sample mean

Independent simulations

Independent example samples

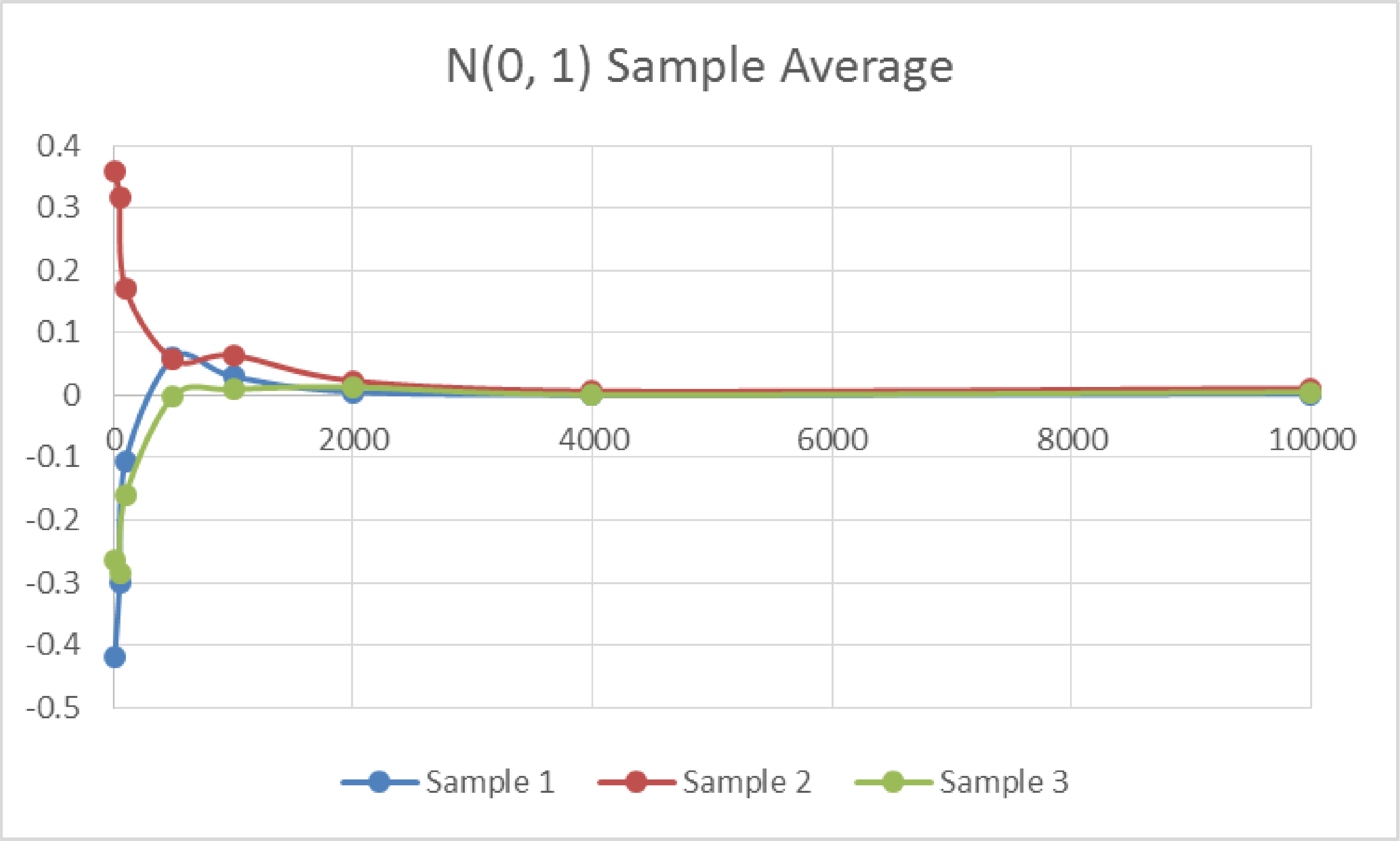
Consider both a standard normal N(0,1) and a Log Normal LogNorm(1,1)





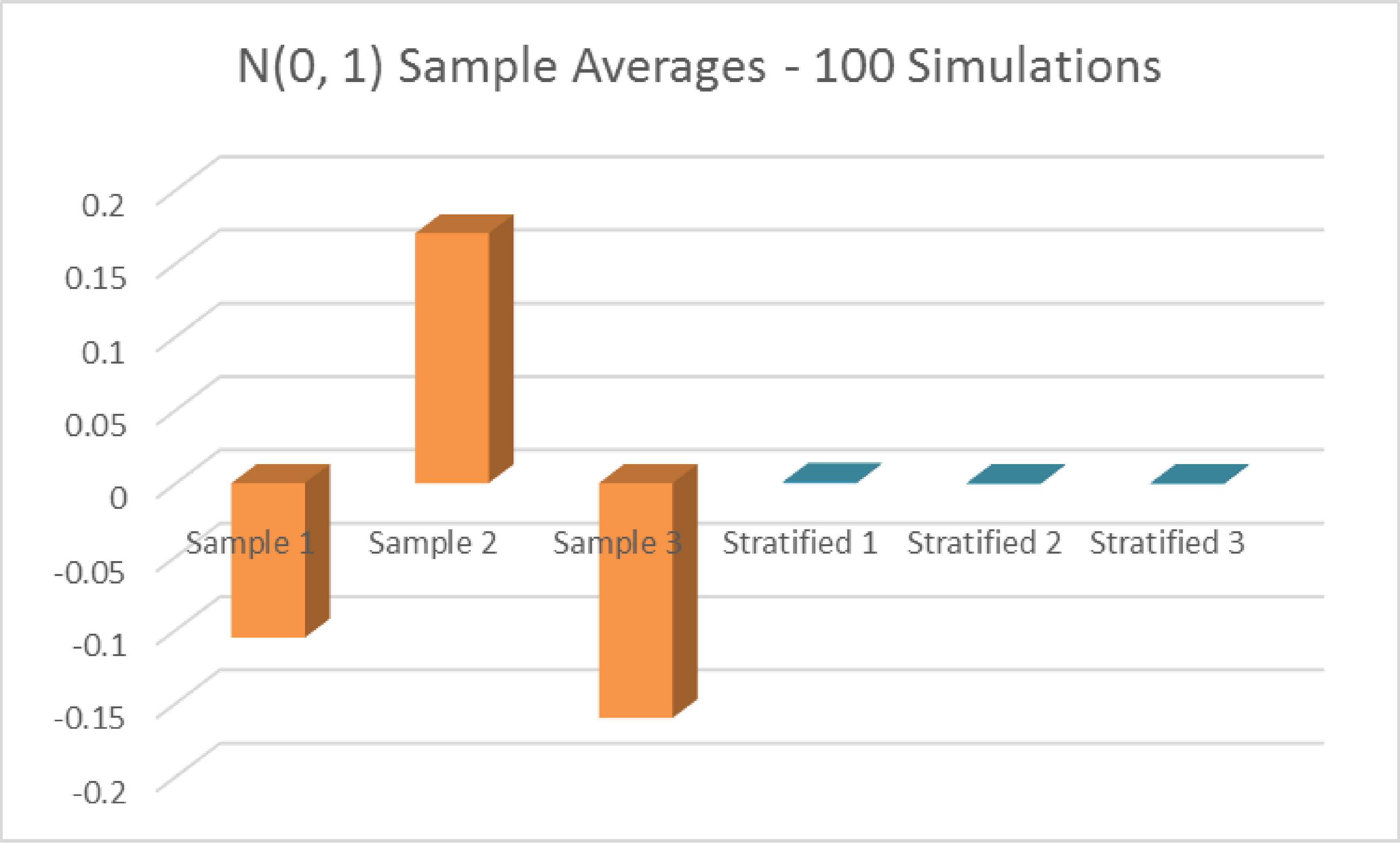
SUNGARD®

Monte Carlo Samples

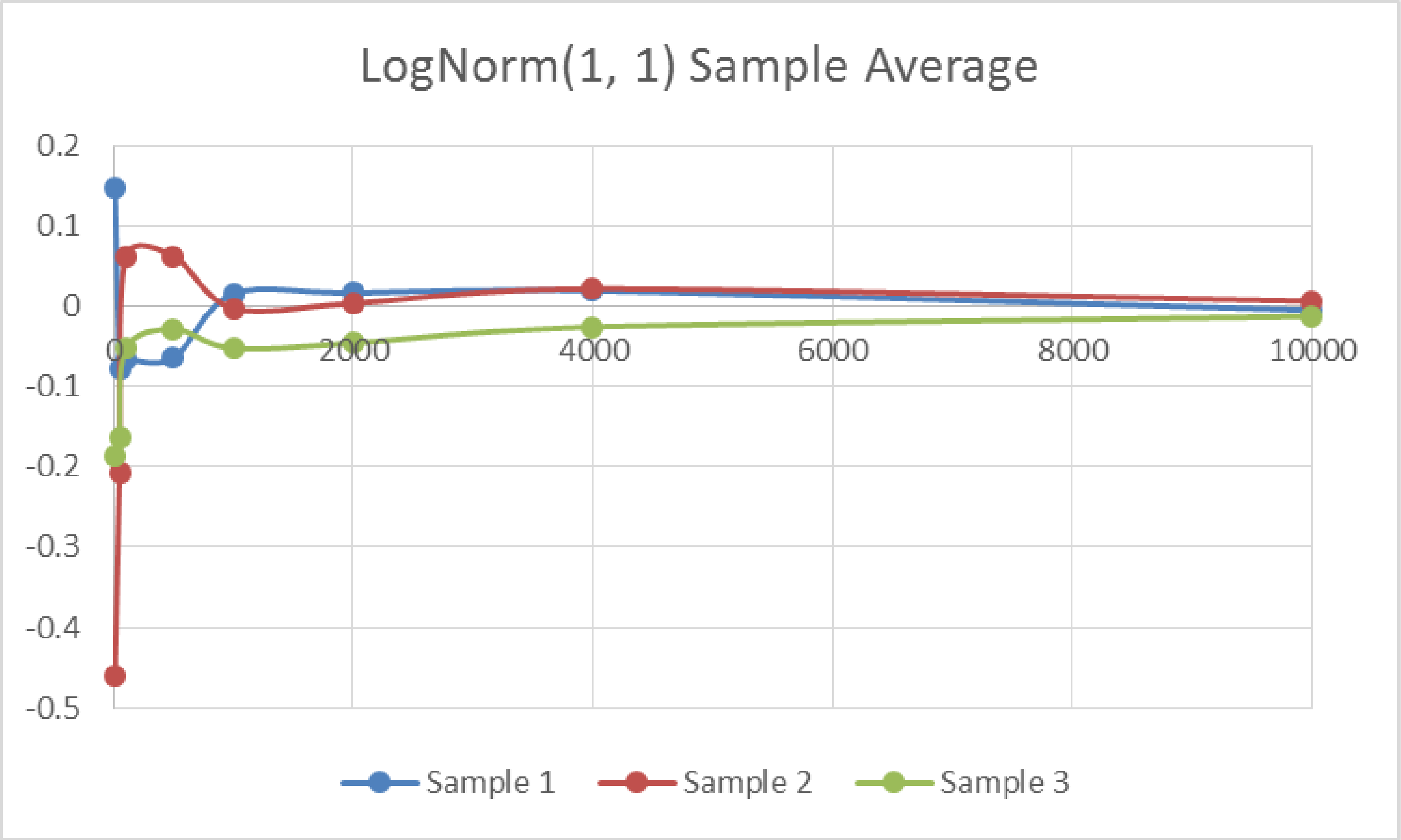


SUNGARD®

Stratified Samples – 100 Simulations

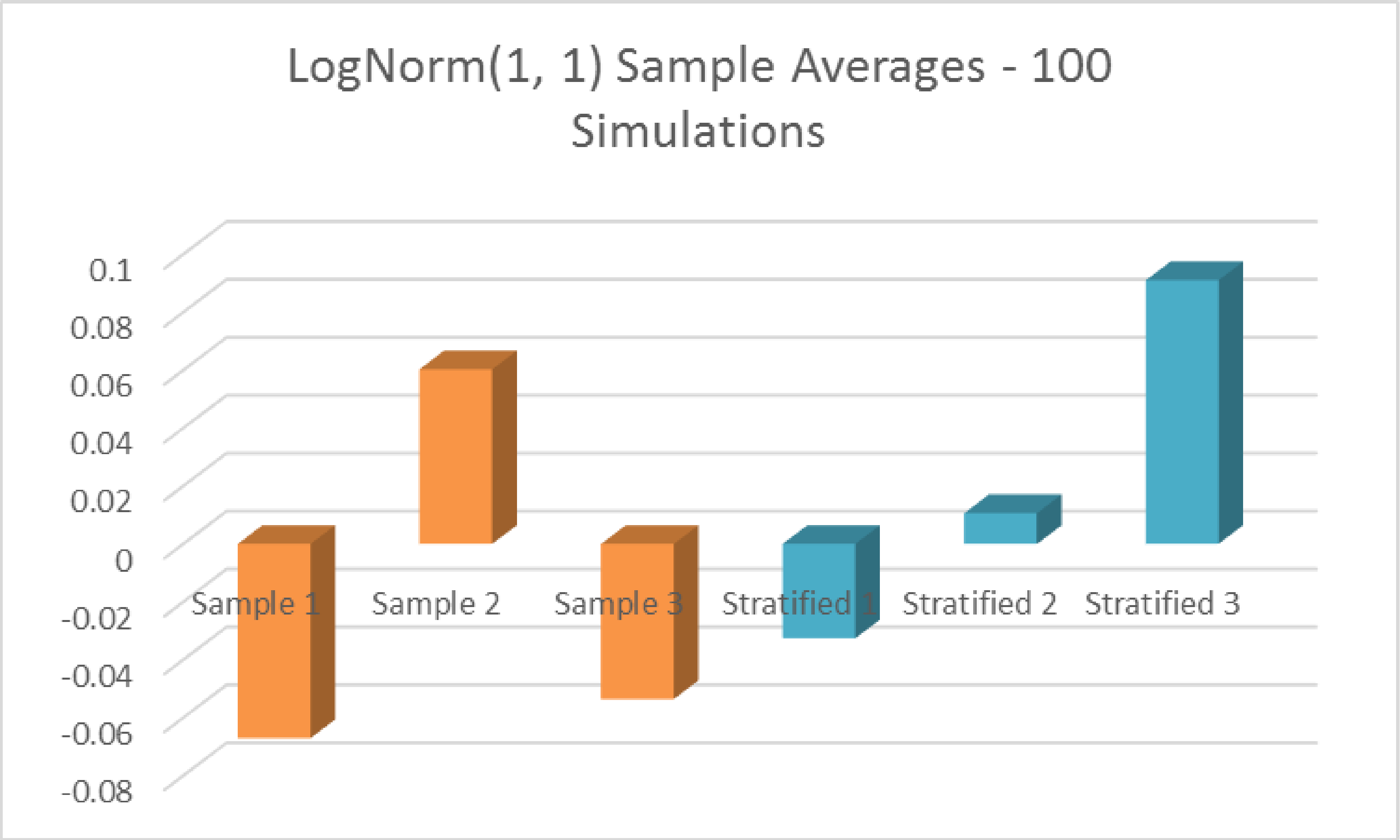


Monte Carlo Samples



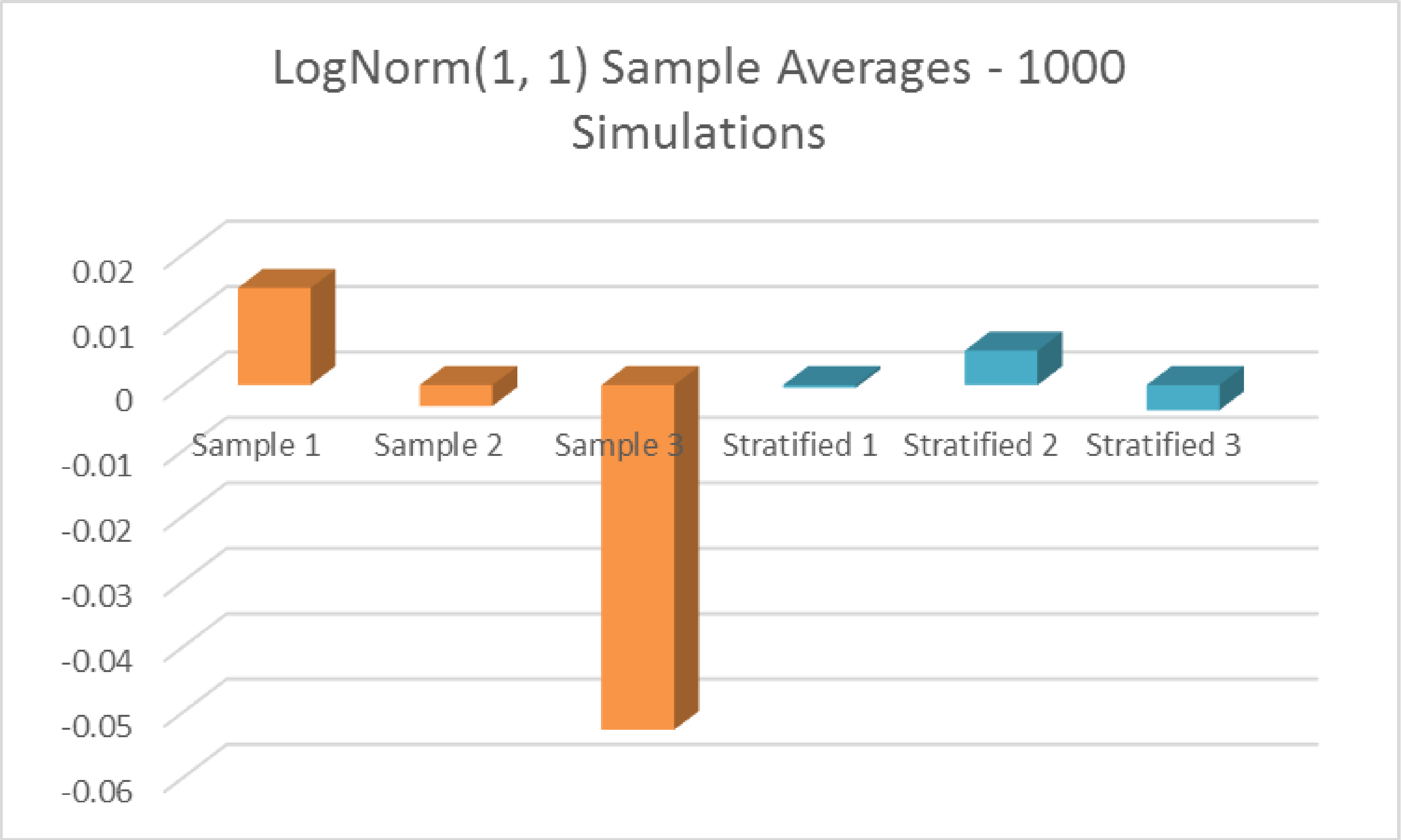
SUNGARD®

Stratified Samples – 100 Simulations

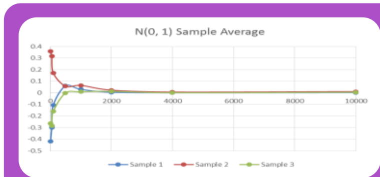


SUNGARD®

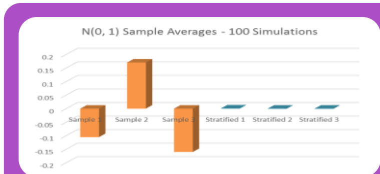
Stratified Samples – 1000 Simulations



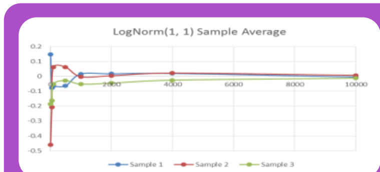
Mean Convergence Examples



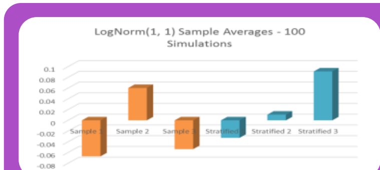
Normal converges reasonably well



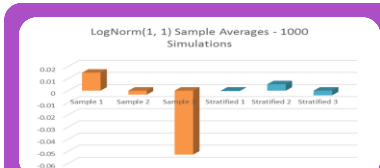
Stratified samples show less volatility and faster convergence for only 100 simulations



Log normal converges slower due to longer tail



Stratified samples show similar volatility and convergence for 100 simulations



Stratified samples show less volatility and faster convergence for 1,000 simulations

Convergence Measures

Convergence is a measure of how well a set of simulations based on sampling potentially represent the true underlying distribution

For additional simulations, will the distribution be significantly different

Mean convergence will tend to be more stable than extreme tail convergence

Statistical measures

- a confidence interval for the mean based on a specified level of confidence, using the t-interval for the mean
- a confidence interval for a specified percentile based on a specified level of confidence, using a binomial approach applied to the sample

Benefits

Fewer simulations needed for convergence

Faster

More stable

Potential Applications

Applications

- Capital modelling
- Stochastic reserving

Examples

- Claim modelling simulations
- Default risk simulations
- Bootstrap simulations

03

Cluster Modelling

What is a Cluster Analysis?

Loose definition would be:
“arranging data into groups whose members are similar in some defined way”

Need a measure of (dis)similarity and an algorithm to arrange the data based on the measure.

Renewed interest due to applications in:

- Segmentation of customer databases for cross-selling
- Clustering of documents for information retrieval
- Data Mining
- Image Analysis & Image Compression
- Insurance Data Compression

Not a New Concept

“arranging data into groups whose members are similar in some defined way"

SUNGARD®

Properties of Clustering Algorithms

Goal of Algorithm

Monothetic

- Groups within the data have a common value for a defined property e.g. all members aged 21

Polythetic

- Members of a cluster are similar, but no one property is exactly the same

Overlap

Hard

- Clusters are not allowed to overlap, so each member of the dataset can belong to only one cluster

Soft

- Clusters may overlap, so each member may be placed in more than one cluster. There will be a measure of association to represent how strongly the datapoint belongs to each group. e.g. Whale Shark

Properties of Clustering Algorithms

Structure

Hierarchical / Connectivity

- Builds a tree structure out of the dataset with clusters forming sub-groupings assuming closer objects are more related than further objects

K-means / Centroid

- Represents clusters using a single mean vector

Distribution

- Modelled using statistical distributions

Density

- Modelled using areas of higher density

Approach to Hierarchy

Dissociative/Divisive

- Top Down Approach
  Start with whole dataset and partition

Agglomerative

- Bottom Up Approach
  Start with elements and aggregate into clusters

Typical Agglomerative Clustering Algorithm

Declare each data point to be its own cluster

#clusters = #datapoints

Calculate the inter-cluster distances according to the measure defined (e.g. Euclidean / ‘straight line’) and the Linkage required between clusters.

Take the closest, most similar, pair of clusters and merge them into a single cluster

#clusters = #clusters – 1

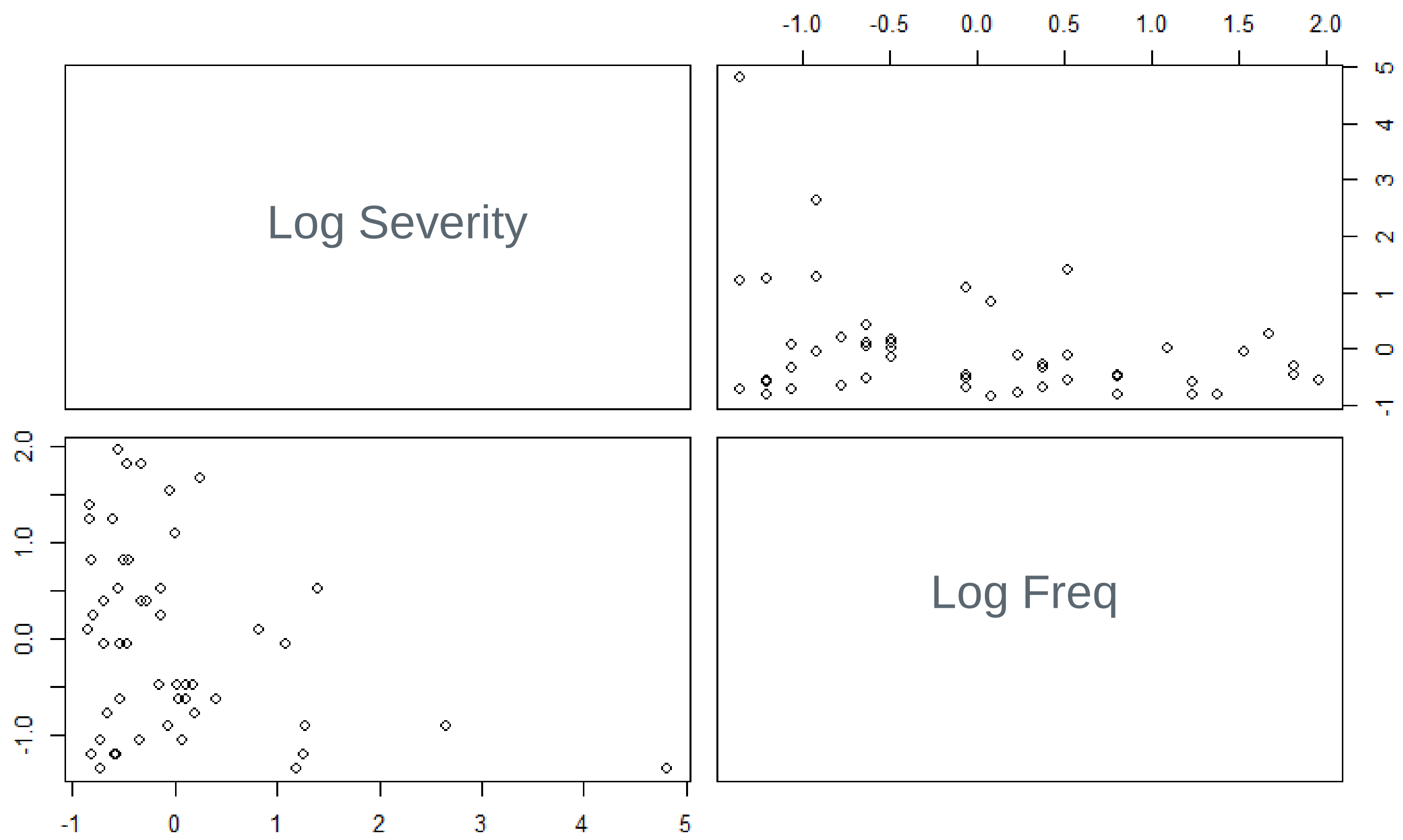
Calculate the inter-cluster distances between the new cluster and the existing clusters

Repeat until all items are in the required level of clustering

SUNGARD®

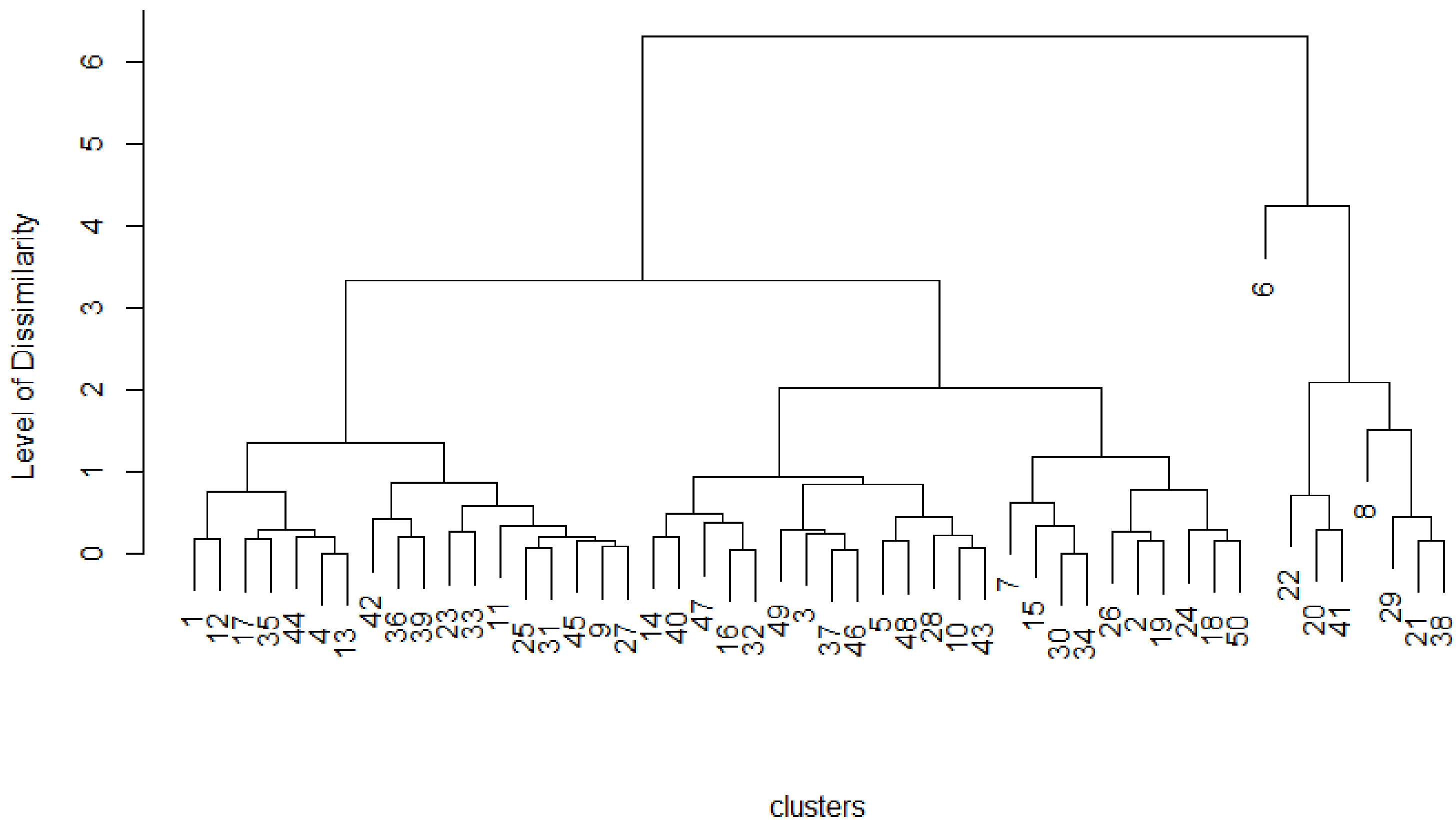
Quick Example Using Two Variables

50 Data points of randomly generated data



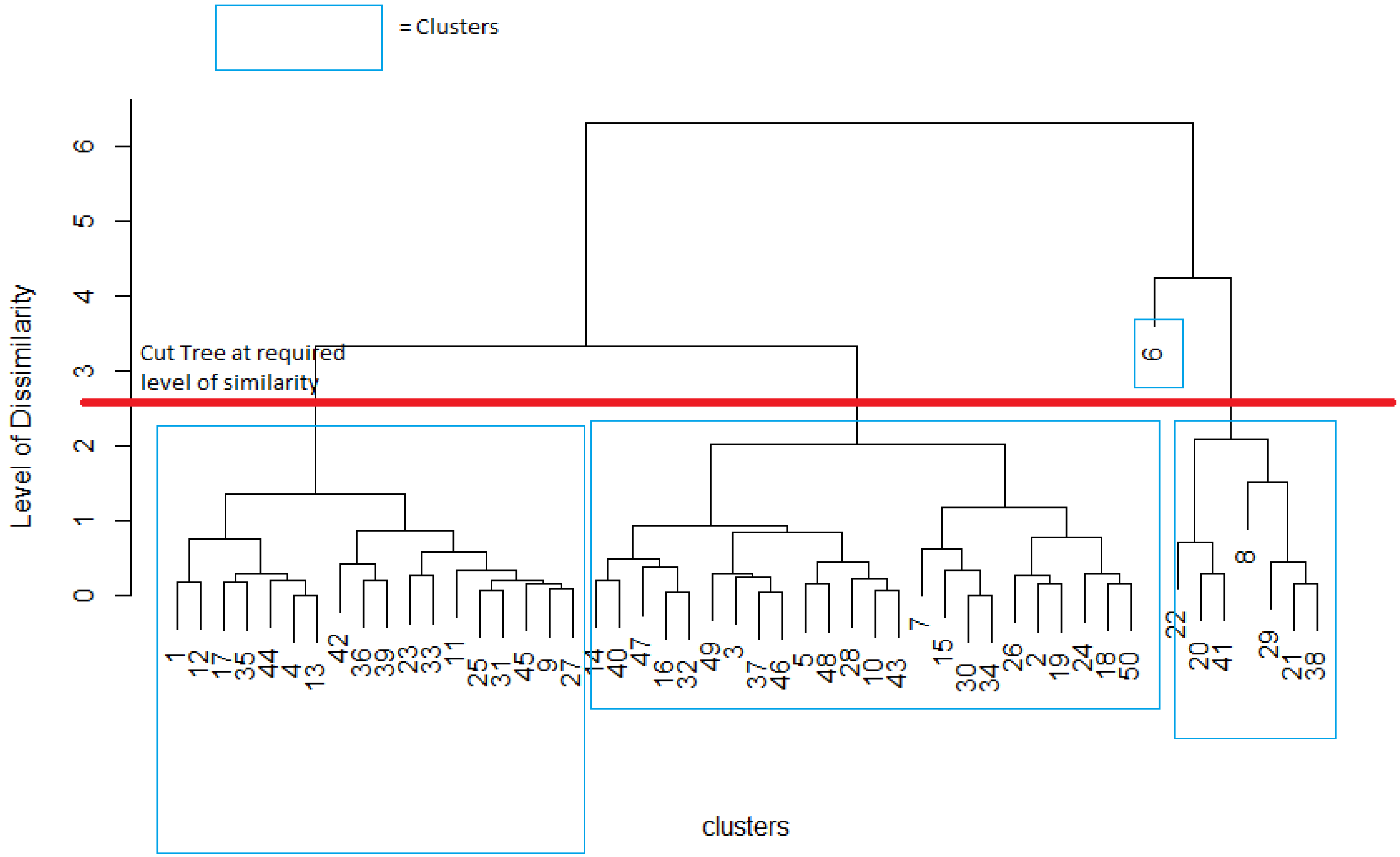
SUNGARD

Outputs from Agglomerative Clustering



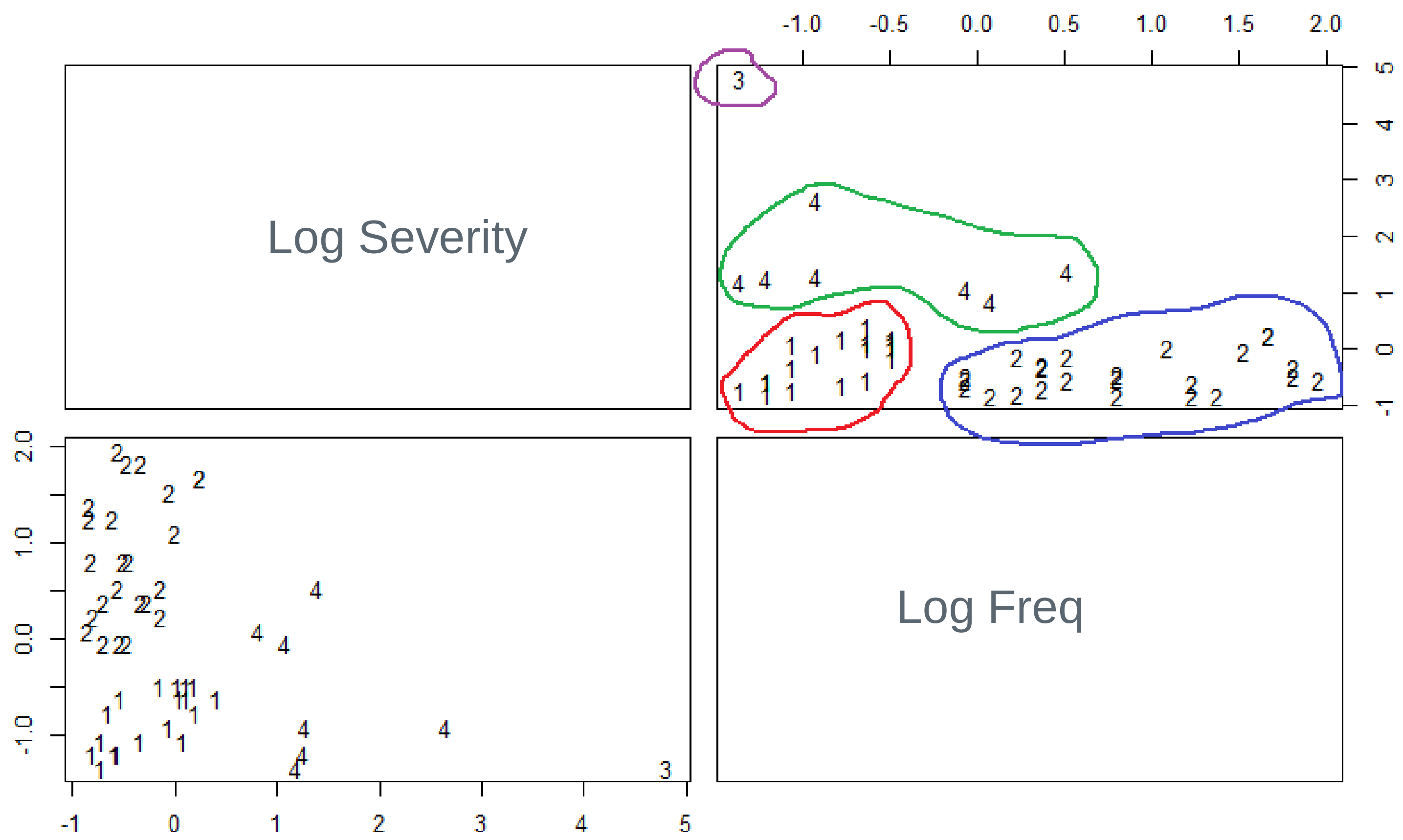
SUNGARD®

Outputs from Agglomerative Clustering



SUNGARD®

Outputs from Agglomerative Clustering

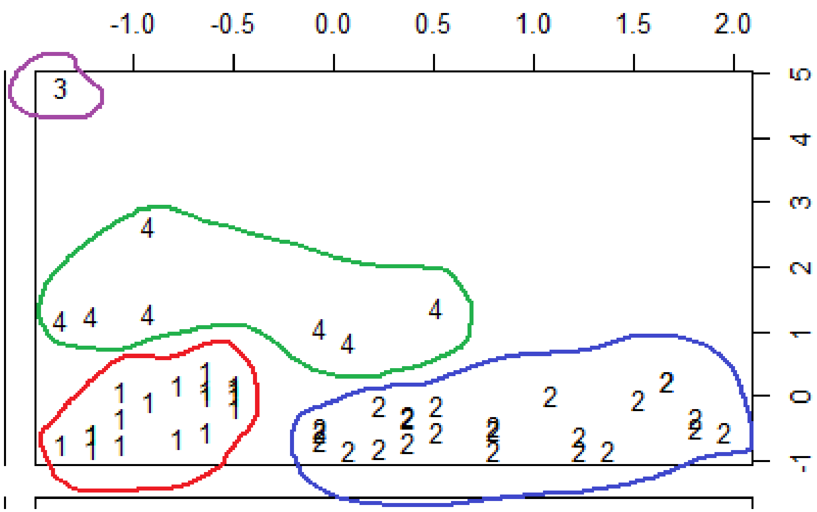


Examples

Assuming data axes are (log) frequency and severity

Simple example of groups could be:

- Vehicle makes and models
- Geographical locations
- Occupations
- Property types
- And so on………



SUNGARD®

Process

Grouping criteria

- Can include results from ungrouped model calculations not just standing data
- Can improve the relevance of the grouping for a particular use

Choices as to how to combine data within each cluster

- Add values
- Weighted average
- Most representative model point

Still requires validation

- Tweaks via:
  - Selection of dimensions
  - Weightings applied to dimensions

Other Clustering Examples

Centroid / k-means

- Clusters are represented by a central vector, which may not necessarily be a member of the data set
- Principal Component Analysis (PCA) groups variables, and can be considered a relaxation of k-means, centroid based, clustering

Distribution

- Clusters are defined as objects belonging most likely to the same distribution
- Common method is a Gaussian mixture model using the Expectation-Maximization (EM) algorithm
- Uses a fixed number of Gaussian distributions

Density

- Clusters are defined as areas of higher density than the remainder of the data
- Objects in sparse areas are required to separate clusters and are usually considered to be noise and border points

Potential Applications

Applications

- Capital modelling
- Pricing analyses
- Predictive analytics
- Reserving

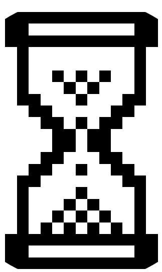
Examples

- Claim burn cost for property damage and liability to create property type risk groups
- Claim burn cost for motor damage and liability to create motor make and model risk groups
- Claim development patterns to create reserve groups

Benefits

Reduces model points, speeds up runs

Make a ‘Heavy’ model ‘Lighter’



Use for quick updates

SUNGARD®

04

Proxy Modelling

What is Proxy Fitting?

Proxy fitting techniques seek to represent one model with another model

Reduces complexity and increases potential understanding

Common techniques include Replicating Portfolio and Risk Factor Polynomial models

Usually fit to liability results from explicit models

Proxy Process

Explicit Model

Accurate but time consuming	Full asset and liability model

↓

Proxy Fitting Tool

Risk Factor Polynomial	Replicating Portfolio

↓

Proxy Definition

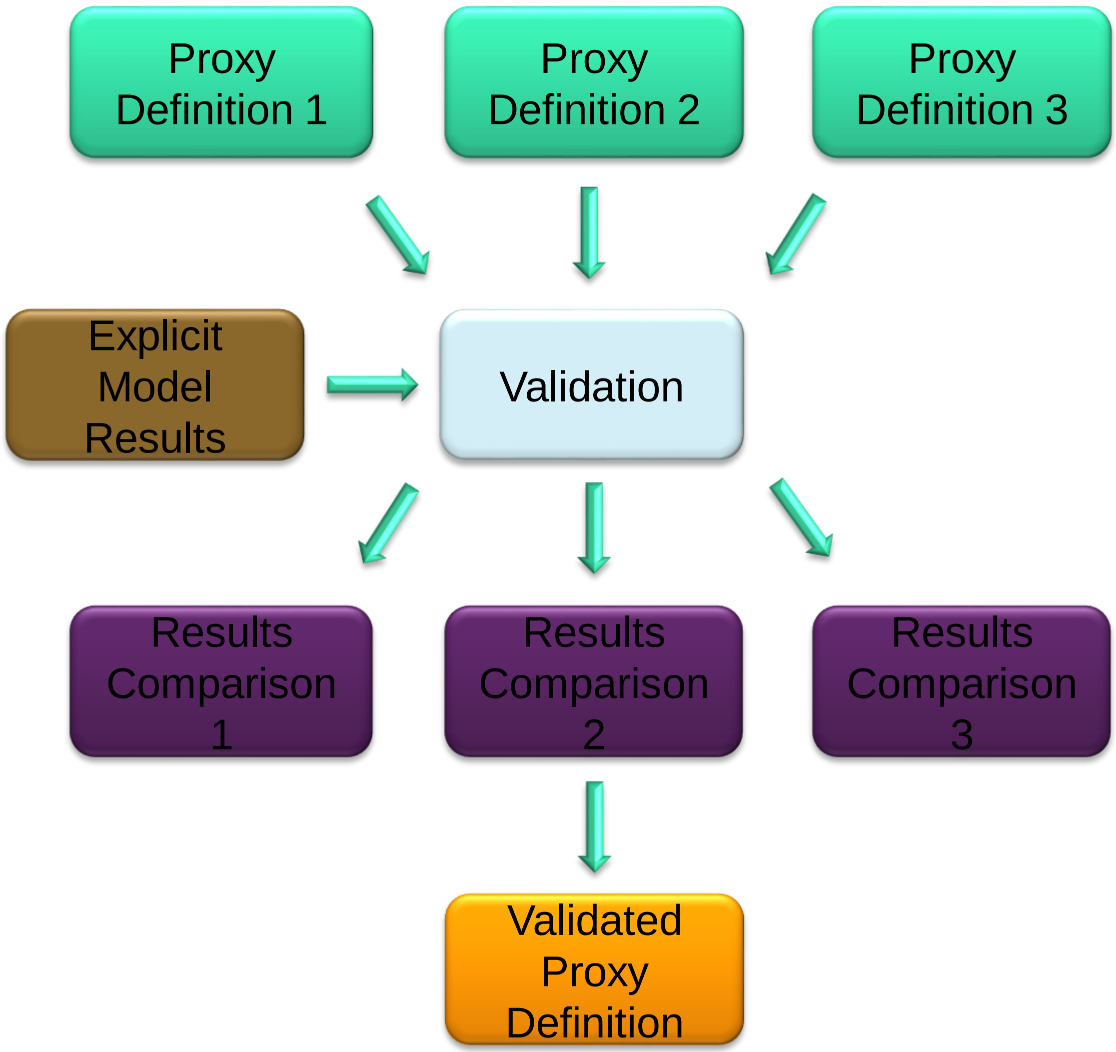
A proxy distribution definition

↓

Proxy Model

Very fast approximation of the explicit model

Proxy Validation



- Use multiple proxy definitions to test against a second set of explicit model results

- Statistical measures might include Chi squared and R^2
- Graphical measures can include residual plots

SUNGARD®

Asset Based Replicating Portfolio

More applicable to investment based risks, e.g. life contracts

Seeking an asset portfolio whose behaviour matches the behaviour of the explicit model

- Model both the assets and the explicit model under different scenarios using a large number of simulations
- Use regression techniques to identify a portfolio of the candidate assets that closely match the explicit model under the different scenarios
- Recalculating results is a matter of revaluing the replicating portfolio assets under different scenarios

Risk Factor Polynomial based Proxy

A polynomial proxy fitting model can represent any type of explicit model

The explicit model must be influenced by the risk factors that are used to form the polynomial proxy fitting model

- A regression algorithm is used to fit a formula whose results closely match the explicit model
- Curve Fitting techniques used including Least Squares Monte Carlo ("LSMC")
- Recalculating results using the proxy simply means revaluing the fitted formula based on changes in the inputs – i.e. the risk factors

What does a Risk Factor Polynomial look like?

Example proxy polynomial:

Explicit model results $\approx 4.2 + 2.3X - 0.9Y^2Z + 0.57Z^2$

- Three risk factors – X, Y & Z
- Example shows four fitted terms – could be different
- Three types of terms
  - Intercept (all risk factors have order 0)
  - Single-factor terms (X and Z^2)
  - Cross-factor terms (Y^2Z)
- Terms may themselves be polynomials – e.g. Legendre, Chebyshev
  - e.g. Legendre order 2 ~ $\frac{1}{2}(3Z^2 - 1)$

SUNGARD®

Curve Fitting

- Can be simple with few fitted points
- Can be more complex with many fitted points

Example shows two dimensions, but n-dimensions in reality

SUNGARD®

Fitting Nodes

Fitting nodes can be many things producing different proxy curves

Fitting Nodes	Proxy Curve
Simulation values by risk inputs	Values by risk input
Mean values by scenarios for differing starting assumptions	Mean values by starting assumption
Percentile values by scenarios (e.g. 1 in 200 year, 99.5th percentile) for differing starting assumptions	Percentile values by starting assumption
Simulation values by percentile	CDF

SUNGARD®

Risk Factor Polynomial Terms

Explicit model results $\approx 4.2 + 2.3X - 0.9Y^2Z + 0.57Z^2$

Prescribe candidate risk factor terms

- Linear programming

Generate terms in a systematic way

- Stepwise regression to select from a possible population

Risk factor terms can be polynomials

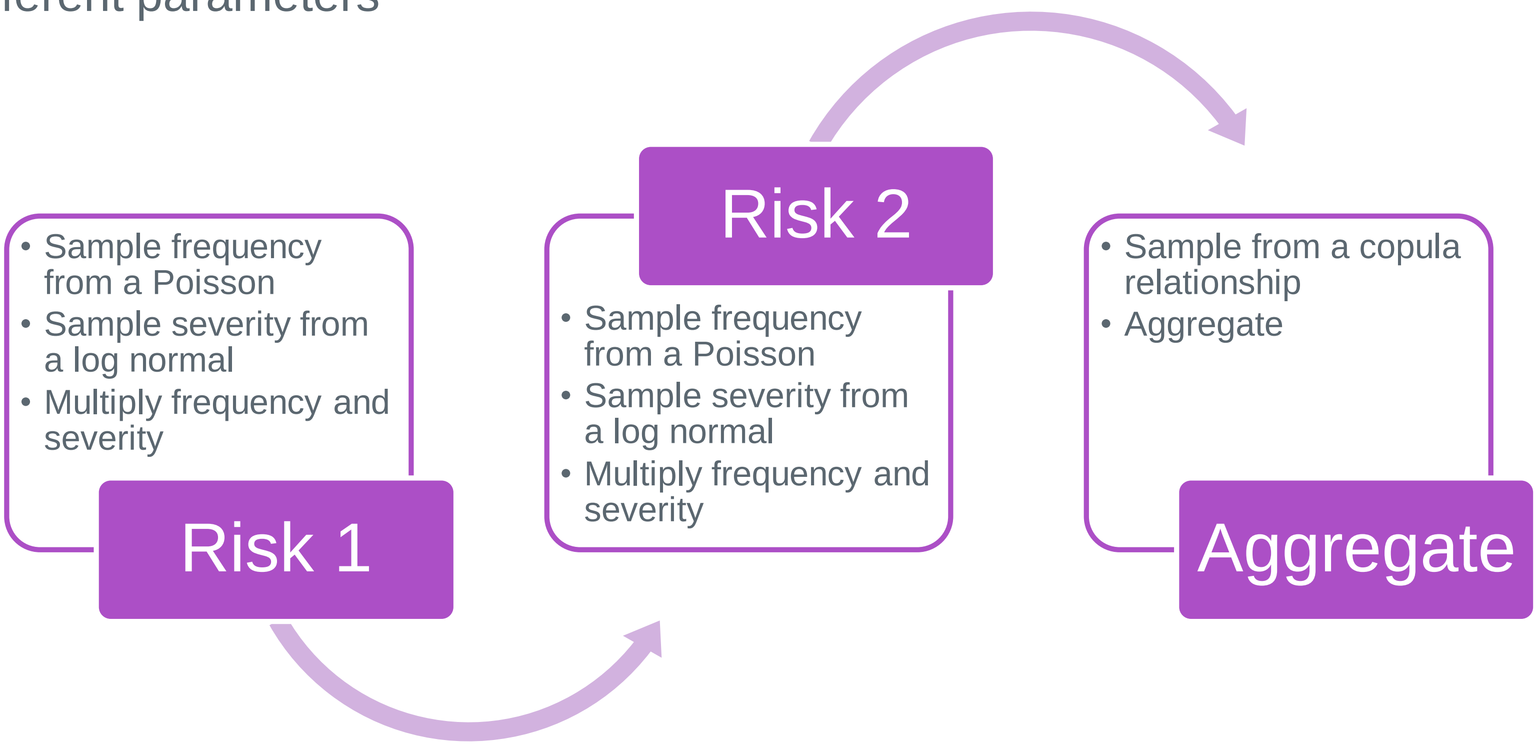
- Simple, e.g. X or XZ^2
- Mathematical, e.g. e^{XZ}
- Legendre & Chebyshev polynomials, e.g. $\frac{1}{2}(3Z^2 - 1)$

SUNGARD

Simple Example

Direct simulation

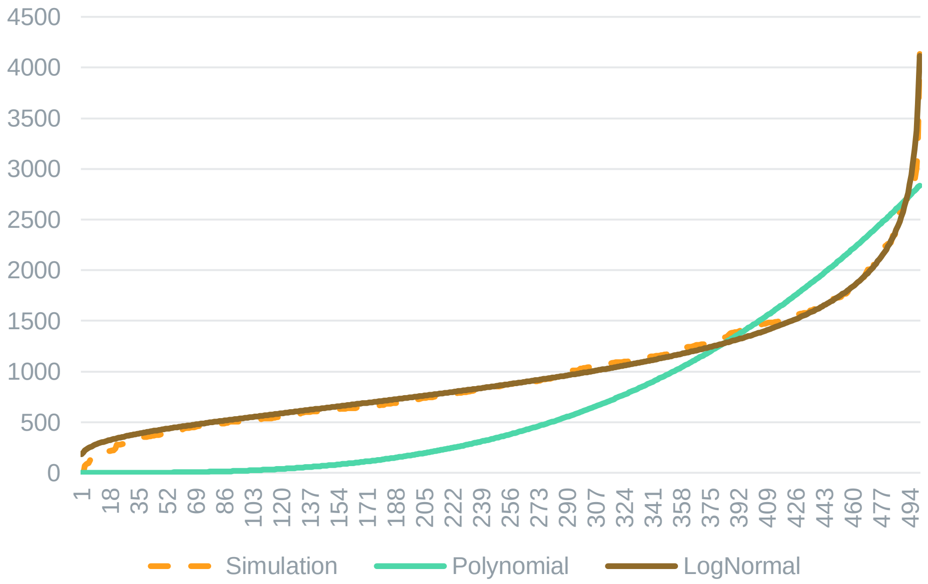
- Two independent classes of risk
- Poisson frequency distribution
- One Log Normal and one Gamma severity distribution
- Different parameters



SUNGARD®

Simple Example

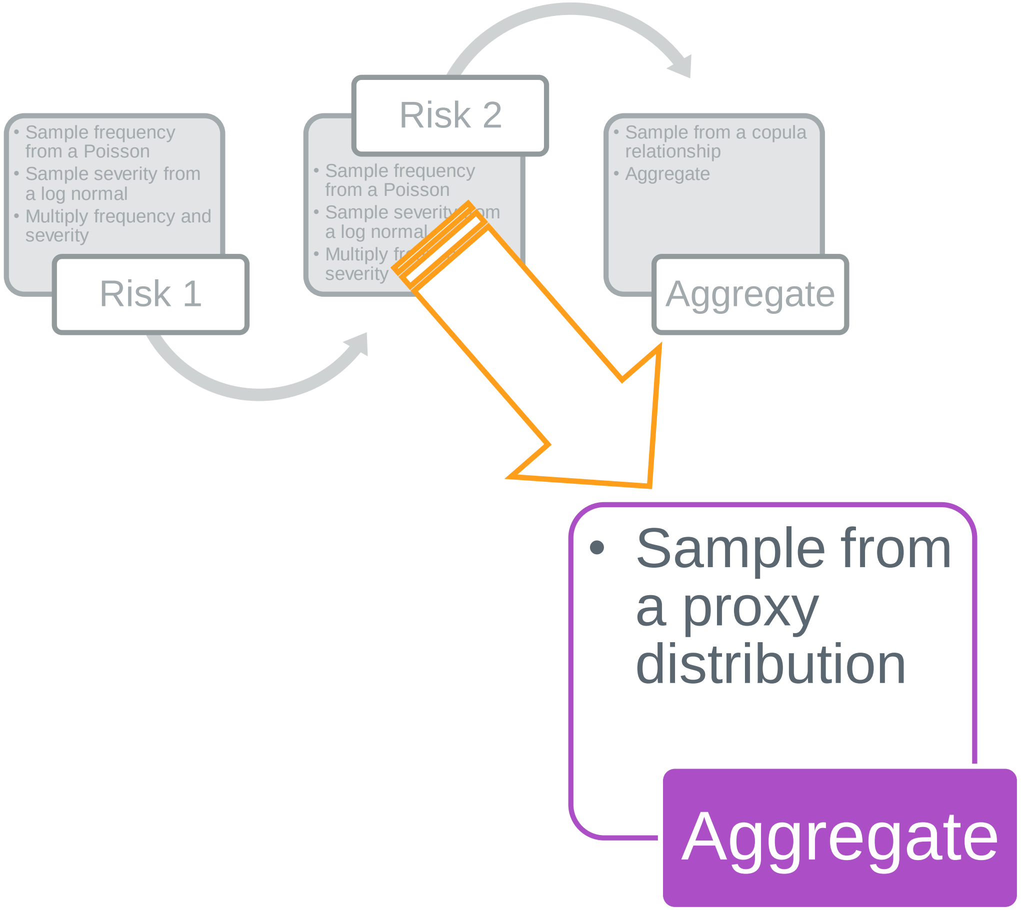
- Consider a proxy
  - 500 simulations
  - Polynomial not a good fit
  - Lognormal a reasonable fit



SUNGARD®

Simple Example

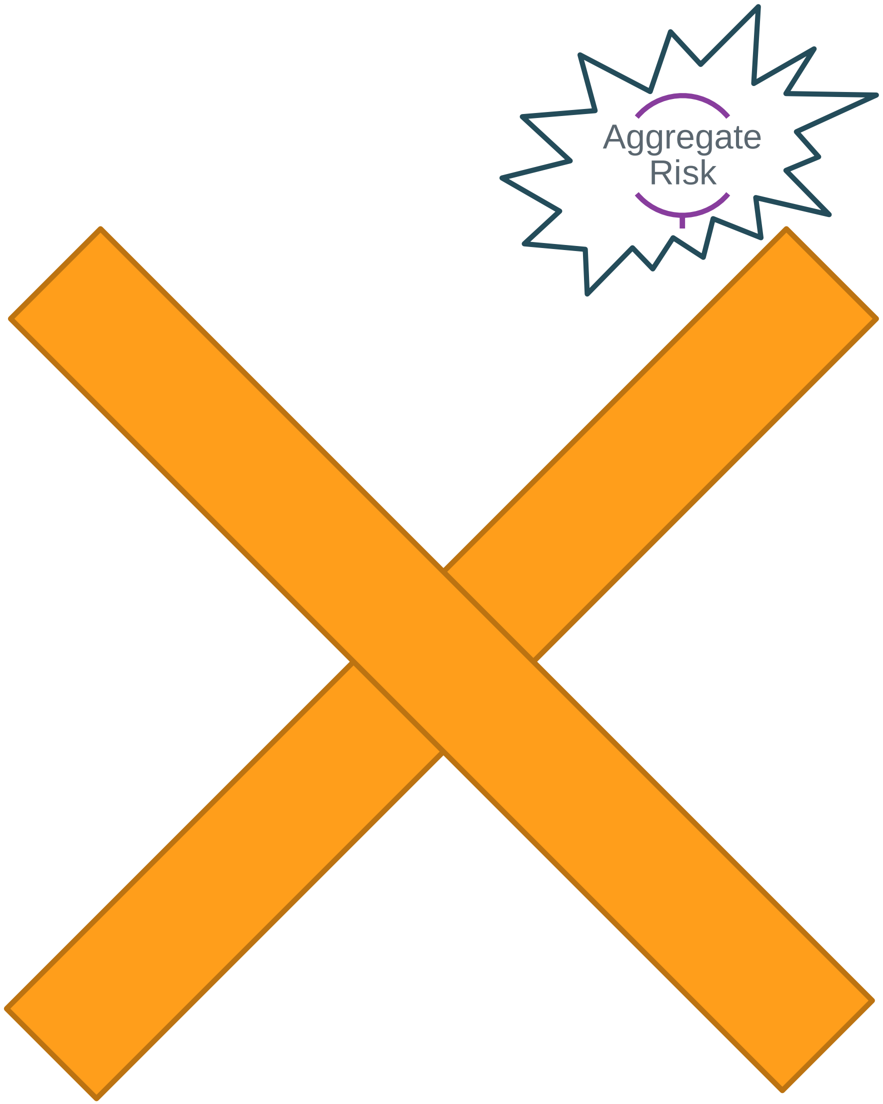
Proxy simulation



SUNGARD®

Reality

Potentially replace with a single model



SUNGARD®

Potential Applications

Applications

- Capital modelling
- Pricing analyses
- Predictive analytics
- Reserving

Examples

- Capital model simulations results
- Burn cost pricing models
- Ultimate claim reserve development

Benefits

Fast recalculation of model results

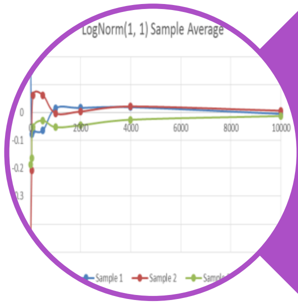
Full distribution from small number of scenarios

Aggregate multiple sources easier

05

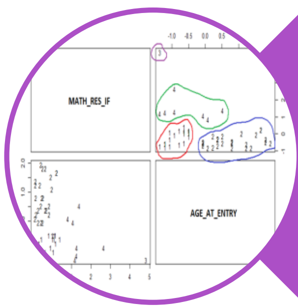
Summary

Summary



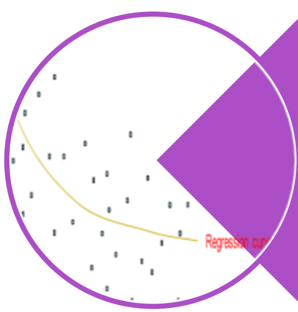
Stratified Sampling

- Produce the true distribution quicker, with fewer simulations



Cluster Modelling

- Use fewer pieces of data to reasonably produce the same result



Proxy fitting

- Produce a formula to generate similar results quicker and simpler

SUNGARD®

SUNGARD®

DOING MORE WITH LESS: GETTING BETTER VALUE OUT OF YOUR CURRENT DATA

Neil Covington
Director of Solutions Management P&C

Questions.