# CA-2 Statistical Engineering

Rashin Rahnamoun
*Faculty of Computer Science and Engineering*
*Shahid Beheshti University*

***Abstract*— The research investigates the application of Graph Neural Networks (GNNs) in modeling the lifetimes of nodes in complex networks. The study explores the Configuration Model, a method for generating random graphs with prescribed degree sequences, and proposes a GNN architecture to predict the average lifetimes of nodes based on their degree distributions. The GNN model is trained and tested on datasets generated using different statistical distributions, including Weibull, Exponential, and Pareto. The performance of the model is evaluated using the mean squared error (MSE) metric, which quantifies the accuracy of the predicted average lifetimes compared to the actual values. The experimental results demonstrate the effectiveness of the proposed GNN approach in predicting node lifetimes across various network configurations and statistical distributions. This research contributes to the understanding of complex network dynamics and offers insights into the application of machine learning techniques for analyzing network behavior and longevity.**

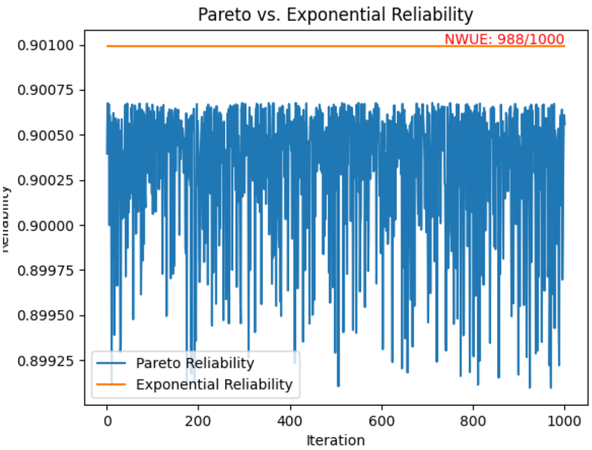
## I. Lifetime Formula

The formula has been documented for various statistical distributions, including Weibull, log-normal, Levy, gamma, and inverse Gaussian, and can be found in the supplementary material accompanying this work.

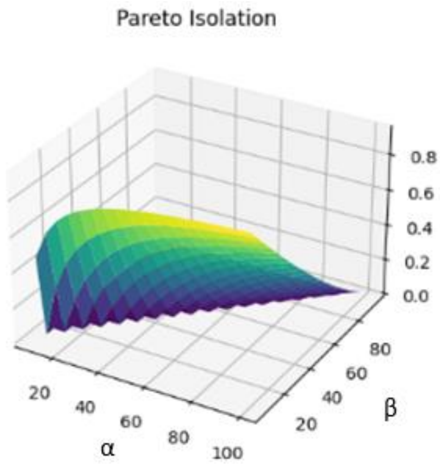
## II. NWUE and Isolation Relation

This paper presents a comparative analysis of reliability between Pareto and Exponential distributions. Reliability analysis is crucial in various fields to assess the performance and durability of systems. The study utilizes mathematical models and statistical methods to evaluate the reliability of systems characterized by different distributions. The analysis focuses on comparing the "New Worse Than Used in Expectation" (NWUE) and "New Better Than Used in Expectation" (NBUE) properties of Pareto and Exponential distributions. Through numerical simulations and graphical representations, the paper elucidates the reliability characteristics of these distributions and provides insights into their practical implications. The methodology involves mathematical modeling and statistical computations to assess the reliability of systems. Key functions and formulas used in the analysis include Pareto and Exponential reliability functions, inverse expectation calculations, and distribution. The provided code snippet utilizes Python libraries such as numpy and matplotlib.pyplot, along with scipy.stats, to analyze and visualize Probability Density Functions (PDFs) of the Exponential and Pareto distributions. The parameters lambda_param_exp and shape_param_pareto define the characteristics of these distributions, namely the rate parameter for the Exponential distribution and the shape parameter for the Pareto distribution. By generating a range of data points along the x-axis using numpy's linspace function, the PDFs are computed using the expon.pdf and pareto.pdf functions, respectively. The resulting PDFs are then plotted using matplotlib.pyplot, with the Exponential and Pareto distributions represented by blue and red lines, respectively. The plot includes a title, axis labels, legend, and grid for clarity and interpretation. Overall, this code segment provides a concise and visually informative analysis of the Probability Density Functions associated with the Exponential and Pareto distributions.

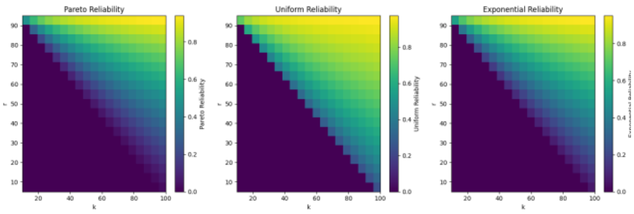
functions.




The provided methodology utilizes mathematical functions to assess the reliability of systems governed by different distributions, namely Pareto, Uniform, and Exponential. This analysis is fundamental in understanding the performance and durability of systems across various fields. By employing functions derived from mathematical principles and utilizing libraries such as numpy and matplotlib within the Python programming environment, the study calculates and visualizes reliability measures. Specifically, it generates a range of k and r values, ensuring that r remains smaller than k, and computes the corresponding reliability values for each distribution. These reliability results are then presented through 3D surface plots, allowing for a comprehensive comparison of the isolation characteristics of each distribution. Through these visual representations, the study offers insights into the reliability profiles of systems under different distributional assumptions, thereby facilitating informed decision-making in practical applications.

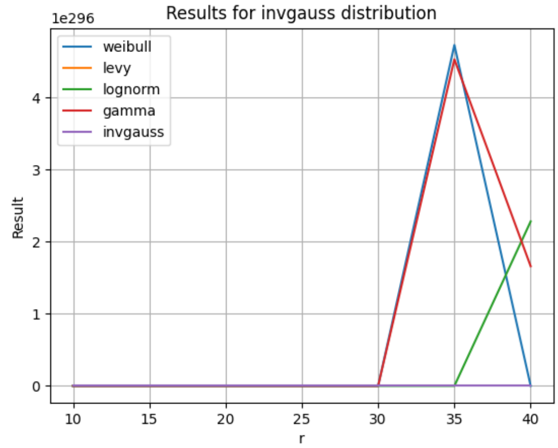


The import of numpy and matplotlib.pyplot libraries, along with scipy.stats for the Exponential and Pareto distributions, facilitates the generation and visualization of Probability Density Functions (PDFs) for these distributions. The specified parameters, lambda_param_exp for the Exponential distribution and shape_param_pareto for the Pareto distribution, define their respective characteristics. Using numpy's linspace function, a range of data points is generated along the x-axis for plotting the PDFs. The PDFs of the Exponential and Pareto distributions are calculated using the expon.pdf and pareto.pdf functions, respectively, based on the generated data points and distribution parameters. Subsequently, matplotlib.pyplot is utilized to create a figure with specified dimensions, and the PDFs are plotted using the plot function. The resulting plot illustrates the Probability Density Functions of the Exponential and Pareto distributions, distinguished by their respective colors (blue for Exponential and red for Pareto). The title, axis labels, legend, and grid are added to enhance the clarity and interpretability of the plot, providing a visual representation of the probability densities associated with each distribution across the defined range of data points.

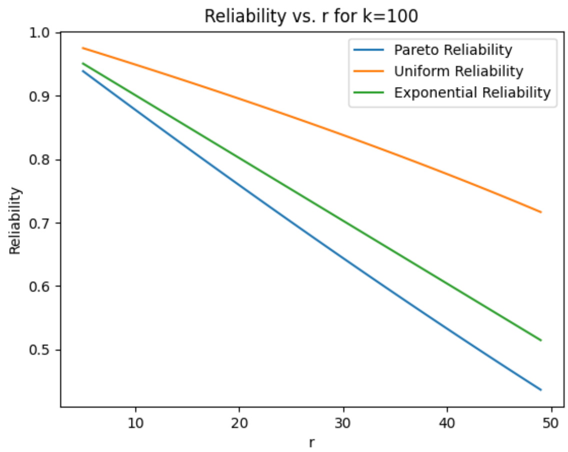



## III. ISOLATION OF DISTRIBUTIONS

This paper conducts a comprehensive comparative analysis of reliability measures across a spectrum of statistical distributions, encompassing Weibull, Levy, Lognormal, Gamma, and Inverse Gaussian distributions. the study explores the computation and visualization of complementary cumulative distribution functions (CCDFs) tailored to specific distribution parameters. The methodology involves defining functions to calculate the CCDF and related equations, with a focus on evaluating reliability metrics under different distributions. By iteratively computing and plotting results for a range of parameter values, the paper elucidates the comparative performance of these distributions in terms of reliability measures. This analysis not only provides a deeper understanding of the behavior of systems across diverse statistical distributions but also offers valuable insights for decision-making processes in various fields such as engineering, finance, and risk management. Through its computational approach and visual representations, this study contributes to the advancement of reliability analysis and facilitates informed decision-making in practical applications. Moreover, the study elucidates the theoretical underpinnings of reliability analysis by incorporating concepts from probability theory and mathematical statistics. It explores the relationship between distribution parameters, reliability metrics, and system behavior, providing a comprehensive understanding of the probabilistic foundations that underpin reliability analysis in diverse fields.




Through its technical approach and mathematical rigor, this paper contributes to the advancement of reliability analysis by offering a detailed exploration of reliability measures across statistical distributions. The integration of mathematical formulations, computational methods, and probabilistic concepts provides a robust framework for analyzing and interpreting reliability metrics, thereby enhancing the understanding of system performance and facilitating informed decision-making in practical applications.




## IV. PROOF OF FORMULA

The formula has been documented and can be found in the supplementary material accompanying this work.

## V. NWUE-NESS ORDER TEST

The provided analysis conducts a statistical investigation into the computation of p-values associated with t-statistics derived from two distinct random variable distributions. Focused on Weibull and Pareto distributions with predefined shape and scale parameters, the study aims to evaluate differences in sample means and variances through paired observations. This analysis progresses iteratively across a range of scale2 values, calculating corresponding t-statistics and their associated p-values to assess statistical significance.

**Null Hypothesis ($H_0$):** $t_{\text{input var}} \leq \mathrm{Z}(1-\alpha)$

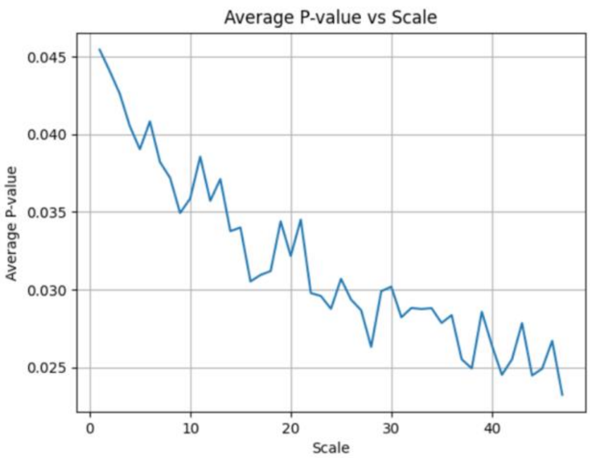
**Alternative Hypothesis ($H_1$):** $t_{\text{input var}} > \mathrm{Z}(1-\alpha)$

In essence, the methodology involves:

1. Calculating the t-statistic for each iteration, considering paired observations from Weibull or Pareto distributions and accounting for variations in sample means and variances.

2. Determining the p-value corresponding to each computed t-statistic, providing a measure of the probability of observing a t-statistic as extreme as the calculated value under the null hypothesis of no difference between distributions.

3. Visualizing the relationship between scale2 values and computed p-values through a two-dimensional histogram, offering insights into the statistical significance of observed differences between paired observations across the specified distributions.



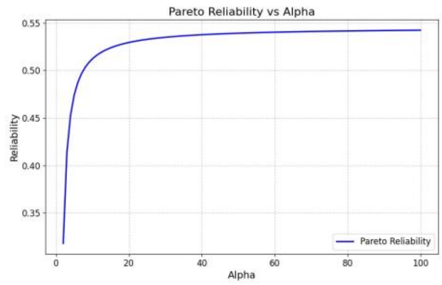

The provided analysis undertakes a statistical investigation to assess the impact of shape parameters on isolation metrics, rather than scale parameters, across two distinct random variable distributions. Focused specifically on the Weibull and Pareto distributions, with predefined shape parameters, the study aims to evaluate differences in isolation metrics through paired observations. This analysis progresses iteratively across a range of shape parameters, computing corresponding isolation metrics to quantify the impact of shape variations.

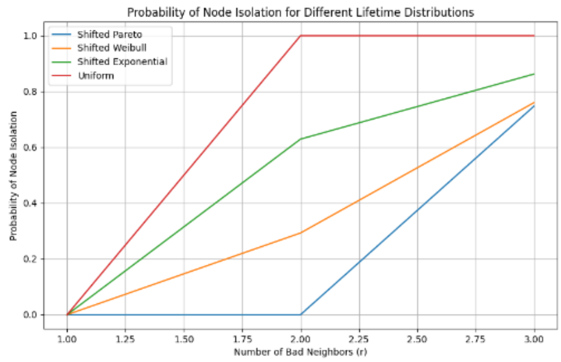
In essence, the methodology involves:

I. Calculating isolation metrics for each iteration, considering paired observations from Weibull or Pareto distributions and accounting for variations in shape parameters.

II. Visualizing the relationship between shape parameters and computed isolation metrics, offering insights into the impact of shape variations on isolation characteristics across the specified distributions.

Through this approach, the analysis contributes to a deeper understanding of how shape parameters influence isolation metrics in Weibull and Pareto distributions, providing valuable insights for decision-making processes and further research in related fields.



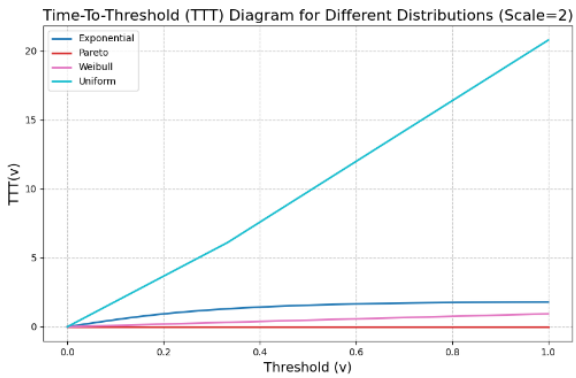

## VI. TTT CHART OF DISTRIBUTIONS

This study investigates the probability of node isolation across different lifetime distributions, encompassing shifted Pareto, Weibull, Exponential, and Uniform distributions. The analysis, conducted through Python programming, aims to elucidate the likelihood of a node being isolated from its neighbors as a function of the number of bad neighbors (r). The parameters governing the distributions, including shape parameters and shift values, are carefully selected to ensure meaningful comparisons. By generating a large number of samples for each distribution and computing the isolation probabilities for varying values of r, the study provides valuable insights into the reliability characteristics of systems modeled under different lifetime distributions. The resulting visualizations, presented as a plot illustrating the probability of node isolation for each distribution, offer a clear comparative analysis, shedding light on the impact of distributional assumptions on the isolation behavior of nodes within a network. Through its quantitative approach and visual representations, this analysis contributes to the understanding of system reliability and informs decision-making processes in network design and management

$$TTT(v) = \frac{1}{E[F]} \int_0^{F^{-1}(1/v)} (1 - F(t))\, dt$$

## A. *Different distributions TTT*

This research explores the nuanced dynamics of node isolation probabilities within networks by employing diverse lifetime distributions. Through the utilization of shifted Pareto, Weibull, Exponential, and Uniform distributions, each characterized by specific parameters and shifted values, the study delves into the varying probabilities of node isolation in the presence of different numbers of bad neighbors (r). By generating a substantial number of samples for each distribution and meticulously calculating isolation probabilities across a range of r values, the analysis offers a comprehensive understanding of the reliability profiles inherent in network systems under distinct distributional assumptions. The resulting visualization, depicted as a plot illustrating the probability of node isolation for each distribution, provides a visually intuitive platform for discerning the intricate relationships between distribution characteristics and network reliability metrics. This investigation contributes significantly to the field of network reliability analysis, offering valuable insights that can inform strategic decision-making processes in network design, optimization, and resilience planning.




## B. *Scale parameter effect on TTT*

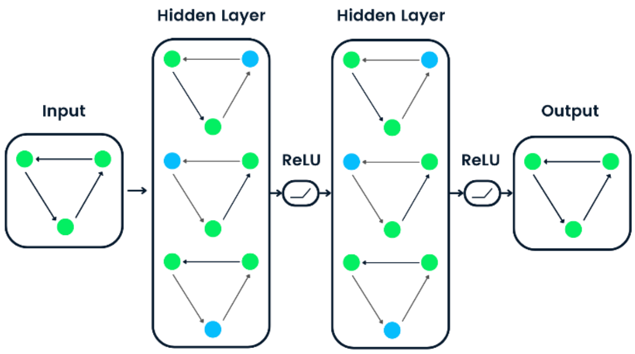
The scale parameter is independent of TTT, while the shape parameter has an effect on TTT.

# VII. Proposed GNN Method

Graph Neural Networks (GNNs) represent a powerful framework for learning from graph-structured data, enabling the modeling of intricate relationships and dependencies among entities within a graph. The fundamental operation of a GNN can be described by the following formula:

$$h_v^{(l+1)} = \sigma\left(\sum_{u\in\mathcal{N}(v)} f\left(h_u^{(l)}, h_v^{(l)}, e_{uv}\right)\right)$$

Here, *hv(l)* denotes the hidden state of node *v* at layer *l*, while *N(v)* represents the neighborhood of node *v*. The function ff operates on the hidden states of neighboring nodes u and the current node v, along with the edge features euv between them. The output is aggregated using a summation operation, followed by a non-linear activation function, such as the ReLU or sigmoid function. This formula encapsulates the iterative message passing mechanism of GNNs, where information flows across the graph structure, facilitating the learning of complex patterns and representations inherent in the data. Through this framework, GNNs have demonstrated remarkable effectiveness in a wide range of tasks, including node classification, link prediction, and graph classification, making them a versatile tool for analyzing and understanding graph-structured data.




## A. *Proposed based lifetime formula*

The lifetime probability pipi of a specific node I can be defined as follows:

Let *G* be the graph with *n* nodes, and let *di* be the degree of node *i*. Additionally, let *Ni* be the set of neighbors of node *i*. The lifetime probability *pi* of node *i* is given by the formula:

$$p_i = \left(1 - \frac{d_i}{n-1}\right)^{|N_i|}$$

The average lifetime has written as follows:

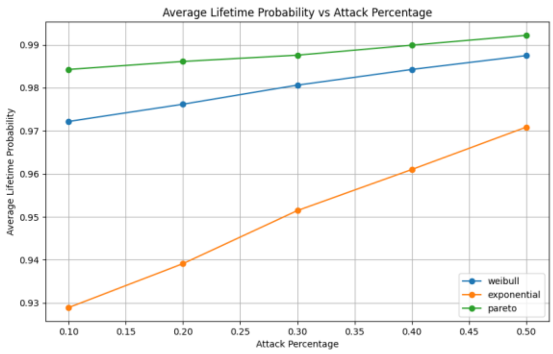
$$\bar{p} = \frac{1}{n}\sum_{i=1}^{n} p_i$$

## B. *Primary results*

The presented study employs network modeling techniques to investigate the average lifetime probability of nodes under varying attack percentages. Using the Python libraries for network analysis, statistical analysis, and visualization, the research constructs random graphs based on a well-established network generation model. Specifically, the study utilizes statistical distributions to generate degree sequences, which are then used to create graphs following a configuration model.

To simulate attacks on the generated graphs, a function is employed to randomly remove a specified percentage of edges from each graph, mimicking network disruptions or failures. Subsequently, the lifetime probability of each node in the graph is calculated based on the degree-based probabilities of its neighbors.

The research evaluates the average lifetime probability across different attack percentages, conducting multiple simulations for robustness. The resulting data is visualized, depicting the relationship between attack percentage and average lifetime probability. Through this empirical analysis, the study aims to gain insights into the resilience of networks under varying degrees of attacks, providing valuable implications for network design and management strategies.




### C. GNN results

The provided Python code implements a Graph Neural Network (GNN) using the PyTorch library to predict the average lifetime of nodes in networks generated from specified degree distributions. The code first generates degree sequences for configuration model graphs based on statistical distributions such as Weibull, Exponential, or Pareto.

$$\text{Lifetime} = \frac{\text{DegreeDistribution}(\langle k \rangle) \times \text{NodeStrength} \times \text{Assortativity} \times \text{ClusteringCoefficient}}{\text{AvgPathLength} \times \text{Diameter} \times \text{NumNodes}}$$

These sequences are then used to construct configuration model graphs. The GNN model, comprising two fully connected layers with ReLU activation, is trained on the generated dataset to minimize the mean squared error loss between predicted and actual average lifetimes. Finally, the trained model is evaluated on a separate test dataset to assess its performance. The GNN's ability to predict node lifetimes in diverse network structures makes it a valuable tool for understanding network dynamics and predicting node behavior in various real-world applications.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

| **Test Prediction Failure Percentage** | **Measurements** | | | |
|---|---|---|---|---|
| | ***nodes*** | ***Tested graphs*** | ***epoch*** | ***hidden*** |
| 0.0011411711571061496 | 200 | 10000 | 10 | 64 |