Learning of Probability Estimates for System and Network Reliability Analysis by means of Matrix Learning Vector Quantization

Klaus Dohmen[1], Mandy Lange-Geisler[1] and Thomas Villmann[1,2]

1- Mittweida University of Applied Sciences - Dept. of Mathematics
Technikumplatz 17, 09648 Mittweida - Germany

2 - TU Freiberg, Akademiestraße 6, 09599 Freiberg, Germany

Abstract. We present a new approach for the assessment of the reliability of coherent systems by using a prototype-based classification method. More specifically, reliability levels for consecutive k-out-of-n systems, which serve as a model for a particular type of networks, are classified using Generalized Matrix Learning Vector Quantization, which provides useful information about the impact of the input probabilities on the classified reliability levels. Our approach is not limited to reliability analysis, but is generally applicable for estimating the probability of the union of any finite family of events, based on their individual and pairwise probabilities.

1 Introduction

In our modern world, we are surrounded by highly reliable technical systems made up of individual components that are prone to failure. Examples include electrical power systems, nuclear power plants, pipeline networks, transportation networks as well as computer and communication networks. Evaluation (and optimization) of the reliability of such systems is an important task in the early design process. The goal is to assess the reliability of the system as the probability that the system is operating, given that its individual components are subject to random and independent failure with known failure probabilities. There are a variety of algorithms for computing or estimating the reliability of such systems, see e.g., [1]. However, in the general case, exact algorithms are highly inefficient, as the problem of computing certain network reliability measures is NP-hard (in fact, #P-complete [1]). For this reason, simulation methods or lower and upper bounds are frequently used for estimating reliability [4], but, due to the inherent complexity of the problem, these methods often produce poor results with high computational costs.

Machine learning approaches to system and network reliability assessment have hardly been investigated to date and are usually combined with traditional methods. For instance, in [6], support vector machines (SVM) are combined with optimization methods to estimate the failure probability, where the objective is an efficient and accurate reliability assessment by classifying system states. In [11], graph neural networks (GNN) are used to predict the survival signature of a network, from which reliability estimates are obtained by applying the law of total probability. These existing approaches do not provide interpretable results, which are useful for optimizing the network in order to increase its reliability.

Our approach differs basically from existing approaches in that it provides *interpretable* estimates for the reliability or unreliability in terms of so-called *reliability* or *unreliability levels*. Furthermore, it does not require to classify the states or to predict the survival signature, which can be computationally intensive. Our central idea is to use a classification scheme to identify prototype vectors in feature space representing the *(un-)reliability levels*. For this, we apply the *Generalized Matrix Relevance Learning Vector Quantization* (GMLVQ) algorithm, which, due to its robustness and its ability to produce interpretable results, has proved useful in many practical classification problems [9, 12].

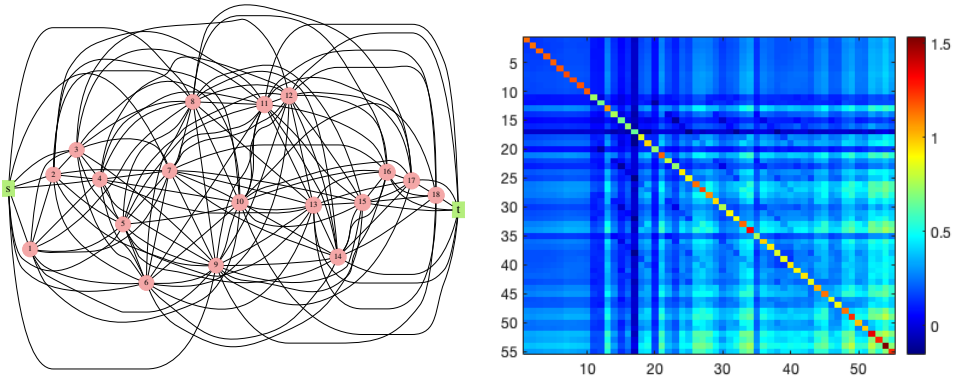
We emphasize that our approach is not limited to reliability analysis, but is generally applicable to estimating the probability of any union of finitely many events, based on their individual and pairwise probabilities. This general problem is usually addressed by applying the inclusion-exclusion principle and related Bonferroni inequalities and has various applications in statistics [5].

Our paper is organized as follows: In Section 2 we review the basic notion of a coherent system from reliability theory, which is general enough to model a wide variety of systems and networks encountered in practice. As a special case, we introduce the consecutive k-out-of-n systems, which are used in the later learning procedure. We proceed by explaining our basic idea, which, as stated above, has applications outside reliability analysis. Section 3 provides a concise overview of the fundamental principles of GMLVQ. The datasets created from consecutive k-out-of-n systems, along with our results, are detailed in Section 4. In Section 5 we summarize our findings and give an outlook on future research.

2 System Reliability Analysis with Machine Learning

A *coherent system* is a pair $\Sigma = (E, \phi)$, where E is a finite set and ϕ is a mapping from the power set of E to $\{0, 1\}$ such that $\phi(\emptyset) = 0$, $\phi(E) = 1$ and $X \subseteq Y \Rightarrow \phi(X) \leq \phi(Y)$ for any $X, Y \subseteq E$. We call E and ϕ the *component set* and the *structure function* of Σ, respectively. It is assumed that each component $e \in E$ is randomly and independently in one of two states, *operating* or *failing*, with probabilities p_e and $q_e = 1 - p_e$, respectively. The set S of operating components of Σ is the *state* of Σ. Σ is said to be *operating* if $\phi(S) = 1$, and *failing* if $\phi(S) = 0$. For $\mathbf{p} = (p_e)_{e \in E}$, the *reliability* of Σ, written as $\mathrm{Rel}_\Sigma(\mathbf{p})$, is the probability that Σ is operating. Analogously, for $\mathbf{q} = (q_e)_{e \in E}$, the *unreliability* of Σ, written $\overline{\mathrm{Rel}}_\Sigma(\mathbf{q})$, is the probability that Σ is failing. Obviously, $\mathrm{Rel}_\Sigma(\mathbf{p}) = 1 - \overline{\mathrm{Rel}}_\Sigma(\mathbf{q})$.

A particular case of a coherent system $\Sigma = (E, \phi)$ is that of a *consecutive k-out-of-n success* resp. *failure system*, where $E = \{1, \ldots, n\}$ and $\phi(X) = 1$ resp. $\phi(X) = 0$ if and only if X contains k consecutive components; that is, the system operates resp. fails whenever k consecutive components operate resp. fail. Figure 1a shows a consecutive 9-out-of-18-failure system, which serves as an exemplary model for a communication network with perfectly reliable terminal nodes s and t (green), unreliable inner nodes labelled from 1 to 18 (red), and perfectly reliable edges. The network is in an operating state if and only if there is a path of operating nodes between s and t. In our example network, this is the case if and only if no nine consecutively labelled nodes simultaneously fail.



(a) consecutive failure network (b) learned relevance matrix Λ

Figure 1: A consecutive 9-out-of-18 failure network with terminal nodes (green) and unreliable inner nodes (red) and the learned relevance matrix of $\mathcal{D}_{2,\mathcal{C}'}$

In general, for any coherent system Σ, its (un-)reliability can be computed from the *minpaths* resp. *mincuts* of Σ, which is any $\subseteq$-minimal $X \subseteq E$ such that $\phi(X) = 1$ resp. $\phi(E \setminus X) = 0$. Thus,

$$\mathrm{Rel}_\Sigma(\mathbf{p}) = \mathrm{Prob}(A_1 \cup \cdots \cup A_n) \quad \text{resp.} \quad \overline{\mathrm{Rel}}_\Sigma(\mathbf{q}) = \mathrm{Prob}(A_1 \cup \cdots \cup A_n) \tag{1}$$

where A_i denotes the event that the i-th pathset resp. cutset of Σ (assuming they are numbered consecutively) exclusively consists of operating resp. failing components ($i = 1, \ldots, n$). To evaluate the right-hand sides in Eq. (1), the inclusion-exclusion principle (the sieve formula) is commonly used:

$$\mathrm{Prob}(A_1 \cup \cdots \cup A_n) = \sum_{k=1}^{n} (-1)^k \sum_{\substack{I \subseteq \{1,\ldots,n\} \\ |I|=k}} \mathrm{Prob}\left(\bigcap_{i \in I} A_i\right).$$

This, however, involves a summation over 2^n terms resulting from the joint probabilities of up to n events. If nothing is known about the independence of the events, it is very difficult or even impossible to determine the joint probabilities of higher order. Even then, the time complexity would be exponential due to the enormous number of terms in the formula. One way of dealing with these problems is to truncate the sum to $k \leq r$, which gives lower bounds if r is even, and upper bounds if r is odd. These so-called Bonferroni bounds (and variants thereof) are often too inaccurate and can take values outside the unit interval, making them useless in this case. Furthermore, it is still necessary to determine the probability of intersections of higher order, which, depending on the specific system or network, can be an enormous hurdle or even be impossible.

This is where our machine learning approach comes into play: Instead of evaluating or computing bounds on the probability of a union, we apply a machine

learning approach to learn these probabilities from the probabilities of intersections of up to two events for different problem instances, that is, by varying the component operation (resp. failure) probabilities and/or the type of system or network under consideration. This is justified as the impact of the mutual probabilities of three or more events, especially in the application domain of system and network reliability, is rather negligible, and since most reliability measures exhibit a similar qualitative behavior, which can be seen, for example, in the S-shaped curve in the case of equal operation or failure probabilities [1].

The basic setup for our machine learning approach are data of the form

$$(\mathbf{t};\tau) = (t_1,\ldots,t_n,t_{1,2},\ldots,t_{n-1,n};\,\tau) \in [0,1]^{n(n+1)/2} \times [0,1], \tag{2}$$

where $t_i = \text{Prob}(A_i)$ $(1 \leq i \leq n)$, $t_{i,j} = \text{Prob}(A_i \cap A_j)$ $(1 \leq i < j \leq n)$, and $\tau = \text{Prob}(A_1 \cup \cdots \cup A_n)$ is to be learned. To avoid misleading permutations, the first n probabilities in each data vector are assumed to be given in ascending order and the pairwise probabilities are assumed to be arranged consistently.

3 Standard GMLVQ

Generalized Matrix Learning Vector Quantization (GMLVQ) is an is an extension of *Learning Vector Quantization* (LVQ) [8, 10]. It improves classification performance through adaptive metric learning by optimizing prototype placement in the feature space to minimize classification errors. More precisely, let $\mathcal{X} = \{\mathbf{x}_j \in \mathbb{R}^n,\, j = 1,\ldots,m\}$ be the set of training data with class labels $c(\mathbf{x}) \in \mathcal{C} = \{1,\ldots,C\}$ and $\mathcal{W} = \{\boldsymbol{w}_k \in \mathbb{R}^n,\, k = 1,\ldots,l\}$ the set of prototypes, each associated with a predefined class $c(\mathbf{w}) \in \mathcal{C} = \{1,\ldots,C\}$. The classification decision is realized via the winner-takes-all rule $\mathbf{w}^* = \arg\min_k d(\mathbf{x},\mathbf{w}_k)$, where d is a dissimilarity measure. Further, we consider the cost function

$$E_{GLVQ}(\mathcal{X},\mathcal{W}) = \textstyle\sum_{\mathbf{x}\in\mathcal{X}} f(\mu) \quad \text{with} \quad \mu = \frac{d(\mathbf{x},\mathbf{w}^+)-d(\mathbf{x},\mathbf{w}^-)}{d(\mathbf{x},\mathbf{w}^+)+d(\mathbf{x},\mathbf{w}^-)} \in [-1,1]\,,$$

where μ represents the classifier function, f is a monotonically increasing function, and $d(\mathbf{x},\mathbf{w}^+)$ and $d(\mathbf{x},\mathbf{w}^-)$ are the dissimilarities to the closest prototypes with matching and non-matching labels, respectively. Training is achieved through gradient-based optimization, maximizing classification margin and ensuring robustness while preserving interpretability [3, 9].

A special feature of GMLVQ is the incorporation of relevance learning by replacing the standard dissimilarity measure with $d(\mathbf{x},\mathbf{w}_k) = (\mathbf{x}-\mathbf{w}_k)^\top\Lambda(\mathbf{x}-\mathbf{w}_k)$ where $\Lambda \in \mathbb{R}^{n\times n}$ is the relevance matrix [10]. The diagonal elements of the learned matrix Λ can be interpreted as the representation of the feature relevance by emphasizing individual feature dimensions in the obtained classification, denoted as classification relevance profile, while the off-diagonal elements capture possible interdependencies between the features. An additional interpretation is provided by the classification influence profile, given by the vector $(\sum_j |\Lambda_{ij}|)_{i=1,\ldots,n}$, which for $i = 1,\ldots,n$ represents the importance of the i-th data feature in connection with all other features for the separation of the dataset [7]. To summarize, this adaptive metric learning can improve the classification accuracy while offering the possibility of valuable interpretability [2].

	$\mathcal{D}_{1,\mathcal{C}}$	$\mathcal{D}_{1,\mathcal{C}'}$	$\mathcal{D}_{2,\mathcal{C}}$	$\mathcal{D}_{2,\mathcal{C}'}$
$\mathsf{acc}_{\mathsf{train}}$	0.965 ($\pm$0.0125)	0.895 ($\pm$0.0075)	0.9575 ($\pm$0.005)	0.895 ($\pm$0.0066)
$\mathsf{acc}_{\mathsf{test}}$	0.9267 ($\pm$0.0058)	0.853 ($\pm$0.0058)	0.9567 ($\pm$0.0208)	0.8633 ($\pm$0.0058)

Table 1: Averaged training and test accuracies and their standard deviations

4 Experimental Design and Results

Following our basic setup in Section 2 we constructed two data sets $\mathcal{D}_1$ and $\mathcal{D}_2$ each containing $N = 500$ samples of the form $(\mathbf{t}; \tau)$ as in Eq. (2) where for $\mathcal{D}_1$ the entries in $\mathbf{t}$ are the individual and pairwise success probabilities of the minpaths of a consecutive 9-out-of-18 success system, whereas for $\mathcal{D}_2$ these are the individual and pairwise failure probabilities of the mincuts of a consecutive 9-out-of-18 failure system as illustrated by the network in Figure 1a. These individual and pairwise probabilities result from the success resp. failure probabilities of the components of the system and their independence, where the success probabilities were chosen uniformly at random from $[0.8, 1]$, and the failure probabilities from $[0.4, 0.7]$. The selection of these intervals is sensible since consecutive k-out-of-n systems are of practical use mainly in case of highly reliable resp. highly unreliable components. Since highly unreliable components are rarely used, we have opted for a medium failure probability in this case.

For each $(\mathbf{t}; \tau)$ in Eq. (2) we consider $\mathbf{t}$ as a training vector with label $c(\mathbf{t})$ resulting from $\tau = \mathrm{Rel}_\Sigma(\mathbf{p})$ resp. $\tau = \overline{\mathrm{Rel}}_\Sigma(\mathbf{q})$ with $\mathbf{p}, \mathbf{q}$ as in Section 2, where τ is computed according to [4, Theorem 5.2]. Special quantiles c_α of τ are used to define the (un-)reliability levels, and thus the labels $c\,(\mathbf{t})$. More precisely, we consider the classes $\mathcal{C} = \{c_{0.25}, c_{0.75}\}$ and $\mathcal{C}' = \{c_{0.25}, c_{0.5}, c_{0.75}\}$, representing the (un-)reliability levels in terms of *low* and *high*, respectively *low*, *middle* and *high*. In this way, from $\mathcal{D}_1$ and $\mathcal{D}_2$ we derive the data sets $D_{1,\mathcal{C}}$, $D_{1,\mathcal{C}'}$, $D_{2,\mathcal{C}}$ and $D_{2,\mathcal{C}'}$, to each of which standard GMLVQ is applied with 3-fold cross-validation for classification and with one prototype per class. The resulting averaged accuracies for training and testing, $\mathsf{acc}_{\mathsf{train}}$ resp. $\mathsf{acc}_{\mathsf{test}}$, and their standard deviations are given in Table 1, which shows that even for a complex system the (un-)reliability levels can be successfully predicted using only partial information on the mutual probabilities associated with the minpaths resp. mincuts of the system, which is a significant advantage over alternative methods. In addition, the learned relevance matrix Λ, shown in Figure 1b, provides a valuable interpretation of the interdependencies between the minpaths resp. mincuts of the system.

5 Conclusion

We demonstrated that the individual and pairwise success resp. failure probabilities of a coherent system's minpaths resp. mincuts can be used to effectively predict its (un-)reliability levels. The method described may be used for larger systems and offers interpretability through the learned relevance matrix. This provides insight into the behavior of the system, which can be useful for reliability optimization, for example. Further applications in other areas may be

addressed in the future as our method is not restricted to reliability analysis and can be used to generically estimate the probability of a union of any finite family of events based on their individual and pairwise intersection probabilities. Such estimates are useful in some applications of statistics, e.g. in multiple comparison procedures, recognition of seasonal trends and outlier detection [5].

References

[1] F. Beichelt and P. Tittmann. *Reliability and Maintenance: Networks and Systems*. Boca Raton: CRC Press, 2012.

[2] M. Biehl et al. "Interpretable Machine Learning in Endocrinology: A Diagnostic Tool in Primary Aldosteronism". In: *Advances in Self-Organizing Maps, Learning Vector Quantization, Interpretable Machine Learning, and Beyond*. Ed. by T. Villmann et al. Lecture Notes in Networks and Systems. Springer, 2024, pp. 96–105.

[3] K. Crammer et al. "Margin analysis of the LVQ algorithm". In: *Advances in Neural Information Processing Systems*. Ed. by S. Becker, S. Thrun, and K. Obermayer. Vol. 15. MIT Press, 2002.

[4] K. Dohmen. "Improved Inclusion-Exclusion Identities and Bonferroni Inequalities with Reliability Applications". In: *SIAM Journal on Discrete Mathematics* 16.1 (2003), pp. 156–171.

[5] J. Galambos and I. Simonelli. *Bonferroni-type Inequalities with Applications*. Berlin: Springer-Verlag, 1996.

[6] J. E. Hurtado. "Filtered importance sampling with support vector margin: A powerful method for structural reliability analysis". In: *Structural Safety* 29.1 (2007), pp. 2–15.

[7] M. Kaden et al. "Learning Vector Quantization as an Interpretable Classifier for the Detection of SARS-CoV-2 Types Based on their RNA-Sequences". In: *Neural Computing and Applications* 34.1 (2021), pp. 67–78.

[8] T. Kohonen. *Self-Organizing Maps*. Berlin: Springer-Verlag, 1997.

[9] S. Saralajew, L. Holdijk, and T. Villmann. "Fast Adversarial Robustness Certification of Nearest Prototype Classifiers for Arbitrary Seminorms". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 13635–13650.

[10] P. Schneider et al. "Adaptive relevance matrices in learning vector quantization". In: *Neural Computation* 21 (2009), pp. 3532–3561.

[11] Y. Shi et al. "Network reliability analysis through survival signature and machine learning techniques". In: *Reliab. Eng. Syst. Saf.* 242 (2024).

[12] R. van Veen et al. "An application of generalized matrix learning vector quantization in neuroimaging". In: *Computer Methods and Programs in Biomedicine* 197 (2020), p. 105708.