Quantum Neural Networks and Topological Quantum Field Theories

Antonino Marcianò [a,b], Deen Chen [c], Filippo Fabrocini [c,d,*], Chris Fields [e], Enrico Greco [f,*], Niels Gresnigt [g], Krid Jinklub [c], Matteo Lulli [h], Kostas Terzidis [c], Emanuele Zappala [i]

[a] Center for Field Theory and Particle Physics & Department of Physics, Fudan University, Jingwan campus, Jingsan Rd, 200433 Shanghai, China
[b] Laboratori Nazionali di Frascati INFN, Via Enrico Fermi, 54, 00044, Frascati (Rome), Italy
[c] College of Design and Innovation, Tongji University, 281 Fuxin Rd, 200092 Shanghai, China
[d] Institute for Computing Applications "Mario Picone", Italy National Research Council, Via dei Taurini, 19, 00185, Rome, Italy
[e] Allen Discovery Center, Tufts University, 200 College Avenue, 02155 Medford, MA, USA
[f] Department of Chemical and Pharmaceutical Sciences, University of Trieste, Via Giorgieri, 1, 34127 Trieste, Italy
[g] Department of Physics, Xi'an Jiaotong-Liverpool University, 111 Ren'ai Rd, 215123 Suzhou, China
[h] Department of Mechanics and Aerospace Engineering, Southern University of Science and Technology, 1088 Xueyuan Avenue, 518055 Shenzhen, China
[i] Yale School of Medicine, Yale University, New Haven, 06510 CT, USA

ARTICLE INFO

Keywords:
Topological quantum field theory
Topological quantum neural networks
Graph neural networks
Quantum amplitude classifiers
Quantum perceptron

ABSTRACT

Our work intends to show that: (1) Quantum Neural Networks (QNNs) can be mapped onto spin-networks, with the consequence that the level of analysis of their operation can be carried out on the side of Topological Quantum Field Theory (TQFT); (2) A number of Machine Learning (ML) key-concepts can be rephrased by using the terminology of TQFT. Our framework provides as well a working hypothesis for understanding the generalization behavior of DNNs, relating it to the topological features of the graph structures involved.

1. Introduction

A paradoxical result in Zhang et al. (2017, 2021) according to which DNNs memorize the training samples by brute force leaves unexplained where the generalization capabilities of DNNs come from. This "apparent" paradox, as it has been dubbed in Kawaguchi et al. (2017), has led to active investigations by many scholars; see for example Arpit et al. (2017), Dinh et al. (2017), Dziugaite and Roy (2017), Hoffer et al. (2017), Keskar et al. (2017), Krueger et al. (2017), Li et al. (2020), Lin et al. (2017), Neyshabur et al. (2017a, 2017b), Shwartz-Ziv and Tishby (2017), Wu et al. (2017) and Wang et al. (2017). In our vision, the overall discussion has empirically demonstrated how far the ML community is from building a principled model of DNNs and, therefore, understanding their generalization capabilities.

Quantum Machine Learning (QML) and quantum algorithms have been employed successfully to obtain significant computational advantages over classical artificial intelligence methods (Aïmeur et al., 2013). The opposite approach, i.e. that of applying classical ML techniques to improve quantum algorithms, is also frequently used e.g. Carleo and Troyer (2017), Lovett et al. (2013) and Tiersch et al. (2015). Quantum Computing (QC) has provided a deep theoretical background to apply quantum algorithms to quantum computers, and quantum approaches to machine learning have recently found profound applications (Paparo et al., 2014; Schuld et al., 2014; Wiebe et al., 2016). In the present article, we are interested in developing a new theoretical background for ML that is based on mathematical notions derived from quantum topology and traditionally applied in theoretical physics. Specifically, we aim at using Topological Quantum Field Theory (TQFT) to construct a topological notion of a neural network, a Topological Quantum Neural Network (TQNN), whose corresponding quantum algorithms provide an algebraic/geometric background to explain the issue of generalization in DNNs. We emphasize that such TQNNs are more general than QNN models employing fixed arrays of quantum gates, as in e.g. Beer et al. (2020) and Farhi and Neven (2018). Our TQNN structure potentially provides a computational advantage

* Corresponding authors.
E-mail addresses: marciano@fudan.edu.cn (A. Marcianò), deenchen@tongji.edu.cn (D. Chen), fabrocini@tongji.edu.cn (F. Fabrocini), fieldsres@gmail.com (C. Fields), enrico.greco@live.it (E. Greco), niels.gresnigt@xjtlu.edu.cn (N. Gresnigt), jinklub_krid@tongji.edu.cn (K. Jinklub), lulli@sustech.edu.cn (M. Lulli), kostas@tongji.edu.cn (K. Terzidis), emanuele.zappala@yale.edu, zae@usf.edu (E. Zappala).

as a consequence of the fact that the projectors used in Noui and Perez (2005) naturally implement arbitrarily deep topological neural networks. We will also show that the semi-classical limit of the objects hereby considered can be interpreted as classical DNNs.

2. Motivations and theoretical background

The main problem addressed in this article is that, despite their excellent performance in many different domains, the source of DNNs' success and the reason for them being powerful ML models remain elusive. DNNs are still analytically opaque in the sense that they miss a principled model of their operation. This issue has a theoretical relevance and, at the same time, it is extremely urgent from an applicative point of view as well. Indeed, if we wish to trust any application making use of Deep Learning technology, we need to open the "black box" of these architectures. Moreover, the problem is worsened by uncertainty coming from the data (data uncertainty) and/or the model (model uncertainty). As a consequence, the predictive reliability of DNNs can be hardly quantified as soon as the model is deployed in the real world. Ideally, ML methods including DNN approaches are used to predict data from the same distribution on which they were trained. Yet, real world domains can lead to shifts between the distribution of data on which the model was trained and on which the model is applied (covariance shift). Many scholars are showing an increased interest in developing approaches for estimating uncertainty in DNNs (as concerns a comprehensive overview, Gawlikowski et al. (2022); concerning recent approaches in estimating uncertainty see, Blundell et al. (2015), Gal and Ghahramani (2016), Lakshminarayanan et al. (2017), Malinin and Gales (2018), Ramalho and Miranda (2020), Van Amersfoort et al. (2020), Wu et al. (2019) and Zhao et al. (2019)). In this sense, a solution to problems of this kind is also going to have a social impact to the extent that it will improve the trustworthiness of AI systems. It has been empirically shown (Zhang et al., 2017, 2021) that successful DNNs can achieve zero training error or very small error when trained on completely random labeling of the true data. On the other side, the test error is not better than random chance insofar as there is no correlation between the training labels and the test labels. However, as the authors of Zhang et al. (2017, 2021) underline, in this case learning should have been impossible to the extent that the semantics of the training samples has been completely corrupted by the randomization of the labels, with the consequence that training should not converge or slow down substantially. Surprisingly, the training process remains largely unaffected by the transformation of the labels. This result seems to leave unexplained the generalization capabilities of DNNs. How to explain that DNNs are actually able to achieve more than good generalization performances, even though the results of learning a function that maps an input to an output based on example input–output pairs show that the training set has been memorized by brute force?

Moreover, the results of Zhang et al. (2017, 2021) have posed a challenge to Computational Learning Theory (CoLT) as well. The experimental results emphasize that the effective capacity of several successful DNNs is large enough to shatter the training data. In other words, the capacity of these models is in principle rich enough to memorize the training data (with or without the use of regularizers). In particular, the classical measures of ML model expressivity (VC-dimension, Rademacher complexity, etc.) seem to fail when explaining the capabilities of DNNs. Specifically, they do not explain the good generalization behavior achieved by DNNs, which are typically over-parametrized models that often have substantially less training data than model parameters (Goodfellow et al., 2016). It is usually understood that good generalization is obtained when a ML model does not memorize the training data, but rather learns some underlying rule associated with the data generation process, therefore being able to extrapolate that rule from the training data to new unseen data. Overfitting and, even more, brute force memorization should exclude generalization by definition, even as concerns human beings. It is not by chance that the concepts of capacity (Cowan, 2001; Feldman, 2000; Lewis, 1996; Miller, 1956; Wattenmaker et al., 1986; Zhu et al., 2009), bias (Griffiths, 2010; Griffiths et al., 2008), overfitting (O'Reilly & McClelland, 1994; Vong et al., 2016), and generalization (Kemp & Jern, 2014; Shepard, 1987) have been widely explored in cognitive psychology as well. This scenario has prompted us towards considering a different framework, the TQNNs framework, for revising a number of traditional ML concepts in light of the perspective of TQFT.

3. Topological Quantum Neural Networks

The mathematical structure used to define TQNNs is that of TQFT. Formally, a TQFT is a functor from the category of cobordisms, which we denote by $\mathcal{C}ob$, to the category of vector spaces. See Fig. 1 for a concise description of cobordisms. Roughly speaking, what the definition of TQFT means, is that to each closed $(n-1)$-manifold we associate a vector space (of arbitrary dimension) on some fixed base field, usually $\mathbb{C}$, and to each n-manifold M between two $(n-1)$-manifolds N_1 and N_2, we associate a linear map between the vector spaces corresponding to N_1 and N_2. What is functorially encoded in this context is the coherence of the composition of manifolds (i.e. gluing manifolds along their boundaries) with respect to the composition of linear maps. With respect to Fig. 1, the manifolds N_1 and N_2 in the top drawing of the figure are associated by a TQFT to vector spaces V_1 and V_2, while M becomes a linear map between V_1 and V_2. In the two drawings in the middle and bottom of Fig. 1, the linear maps corresponding to M_1 and M_2 are composed, through the vector space associated with N_j, which we call V_j. In the case of the bottom drawing, further, V_j is the tensor product of two vector spaces, corresponding to the two connected components of N_j.

The composition rule of $\mathcal{C}ob$ is translated into the composition of linear maps between vector spaces. We can, in particular, think of any linear map $f : N_1 \mapsto N_2$ as an arbitrary finite composition $f = f_m \circ f_{m-1} \circ \ldots f_2 \circ f_1$, where each of these m maps is associated to some n-manifold M_k, subject only to constraint that the M_k can be successively glued together. Hence, we can equally well think of each f_k as an element of an equivalence class of smooth paths through M_k, paths to which amplitudes will be assigned in the construction below.

The typical elementary example of TQFT is in dimension 2, i.e. one dimension lower than the TQFTs considered in this article. We have a fixed vector space V for each copy of the circle (i.e. 1-manifolds), and the vector space $V^{\otimes r}$ is associated with 1-manifolds that consists of multiple copies of circles. Then, let N_1 consist of r circles and N_2 of s circles. To a surface connecting N_1 and N_2 we associate a linear map $V^{\otimes r} \longrightarrow V^{\otimes s}$. It is a "folklore" result in quantum topology that TQFTs in dimension 2 are classified by Frobenius algebras. Observe, in particular, that in the previous scheme we have that to a closed manifold (i.e. without boundaries N_1 and N_2) is associated a linear map between two copies of $V^{\otimes 0} \cong \mathbb{C}$. This is nothing but a complex number that is an invariant of the manifold. In practice, such invariants arise as *partition functions*, i.e. sums over all possible colorings with certain Lie group representations, see Appendix B. The class of TQFT relevant to this article makes use of the spin-network representation in gauge theory, where we have that the boundary vector spaces are Hilbert spaces whose bases are given by cylindrical functions corresponding to spin-networks. A

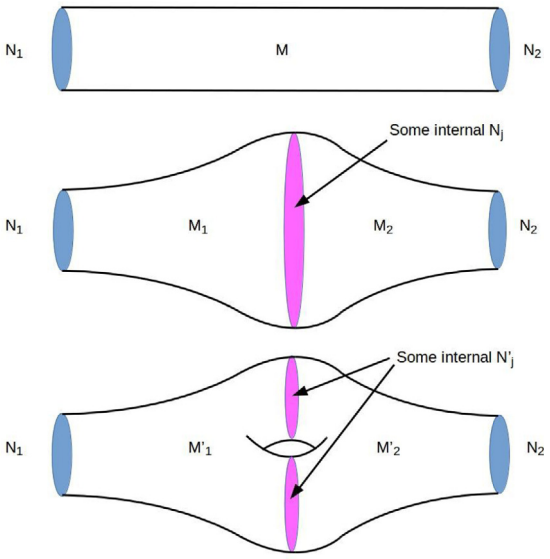


Fig. 1. Schematic representation of $\mathcal{C}ob$. The top drawing shows a manifold M whose boundary consists of two manifolds N_1 and N_2. While N_1 and N_2 are objects in $\mathcal{C}ob$, the manifold M is a morphism. In the middle and bottom, two cobordisms are glued along their common boundaries (where the orientation of N_j in M_2 is taken with opposite sign). This provides a composition rule for morphisms having the same target and source objects.

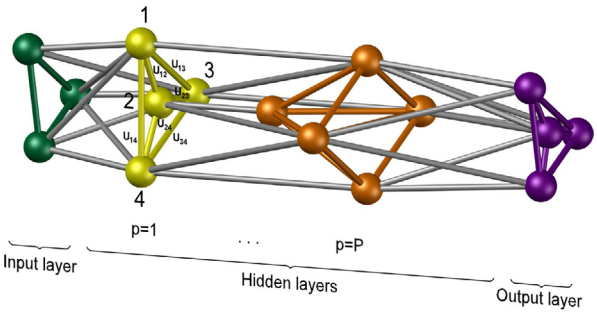


Fig. 2. A functorial evolution among two spin-network states.

detailed description of the definitions is provided in Appendix A. We define a TQNN to be a TQFT whose target vector spaces are tensor products of the Hilbert space of cylindrical functions, taken with the (regularized) Ashtekar–Lewandowski metric.

In this setting, therefore, we can take an input spin-network associated with the dual cubulation of a boundary manifold, and map this to another output spin-network. Associated with such a mapping there arises a scalar that is geometrically derived by "capping" the boundary components to obtain a closed manifold. This scalar is interpreted as being a probability amplitude for a transition between two spin-networks. This is the outcome of applying a TQNN between input and output states. In practice, given two spin-networks $(\Gamma_{in}, \Gamma_{out})$, a TQNN returns the transition amplitude from Γ_{in} to Γ_{out}, which in turn can be used for a binary classification problem, e.g. a transition amplitude whose modulus square is higher than a predefined "confidence" number between 0 and 1 implies, when used as an indicator, that the input is classified as the output. See Appendix A for details.

A tight texture of analogies provided by the equivalence between this categorical approach to quantum field theory and deep machine learning specifies the theoretical perspective through which we progress. Following the recent literature (Farhi & Neven, 2018), these states can be considered as part of a QNN machine, and their state transitions as implementing quantum computations. The former is supported on 1-complexes (graph Γ), and are endowed with a functorial evolution supported on 2-complexes. This 2-complex evolution is, in turn, a cobordism acting at an internal boundary (an $n-1$-manifold) that is effectively a "hidden layer" of the TQNN; however unlike in a QNN architecture with fixed layers, in a TQNN each "layer" can be further decomposed into an arbitrarily long sequence of intermediate evolution operators (n-manifolds glued by further cobordisms) and hence into a further nested sequence of "hidden layers" as schematized in Fig. 2. This functorial evolution on 2-complexes is amenable to a training algorithm specifically adapted to our TQNN framework, as detailed in the Appendix.

We consider, in the present article, the case of a TQFT with a local non-abelian Lie group, which we assume for the sake of simplicity to be SU(2). This specific choice, in particular, allows us to parallel the example of QNN provided above. Then, square $(2j+1)\times(2j+1)$ matrices depending on the Euler angles turn out to constitute the representations of the group elements $U \in \mathrm{SU}(2)$. At the vertices of the graph supporting the quantum state, tensors saturating the matrix indices are specified by the intertwiners of SU(2). In our setting, these are initial and final states of the TQNN, rather than the network itself. The functor, as an operator the action of which is supported on the disjoint boundary states, corresponds to the classifier, i.e. the overall map $f : N_1 \mapsto N_2$ implemented by the TQNN as described above.

We conclude this section with a few remarks about TQNN. First, we notice that the definition of TQNN does not generally fix the geometry of the network, but it rather determines a "preferred" geometry to detect certain (equivalence classes of) states by considering the highest transition amplitudes. Moreover, we naturally implement the superposition principle, as a sum (of sorts) over all possible histories between boundary states, i.e. paths through the intervening n-manifold M. This might be compared to utilizing classical networks of arbitrary layer widths and depths simultaneously, as different histories present in general a different number of single vertex transitions that are composed in order to evolve from one boundary state to another. Following this line of interpretation, it is reasonable to expect that ideally, a TQNN "implements all input/output equivalent DNNs in parallel" (Deutsch, 2002) and hence would provide considerably higher computational performance with respect to a classical neural network.

Interestingly, while as noted above the most straightforward interpretation of QNN as spin-networks assumes that the quantum machine corresponds to a given spin-network, in TQNN an appropriate functor determines the transition between two spin-networks that are associated to single states. This functor represents, in effect, a superposition of quantum machines implementing the chosen function $f : N_1 \mapsto N_2$ from the input to the output state. Replacing single maps with functors representing appropriate equivalence classes of maps in this way is commonly referred to as *categorification* in mathematics.

We notice that the functor that defines a TQNN derives its form from integration on the possible geometries that determine a transition between boundary states. More specifically, it is known (see for example Rovelli (2011) and references therein) that the partition function explicitly defined in Appendix B approximates the Einstein–Hilbert action in the semi-classical limit, and the integration variables can be interpreted as living in the moduli space of (equivalent up to diffeomorphism) metrics over the base manifold. Rovelli (2011) compares this approximation to a "concrete implementation" of the Misner–Hawking integral. In the setting of the present article, this is interpreted as the learning rule itself. A TQNN computes transition amplitudes between states by obtaining a partition function determined by

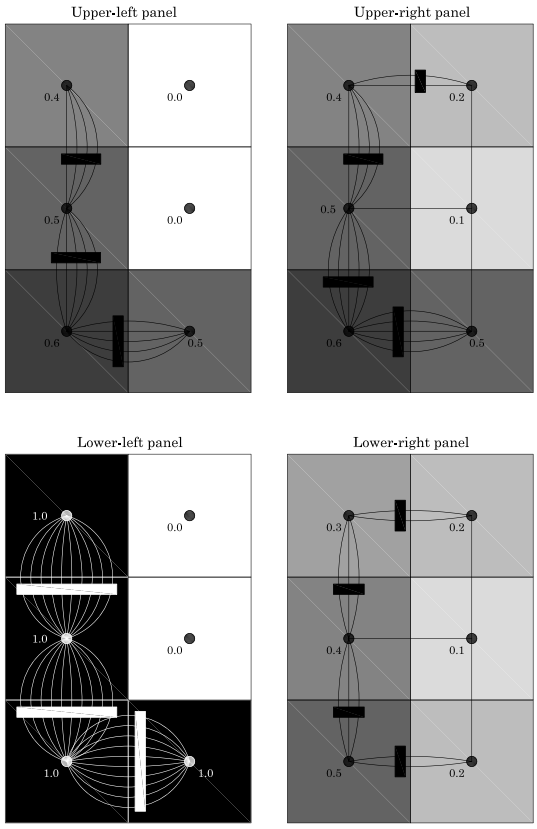


Fig. 3. Superimposed to four different images are the associated graphs, endowed with assigned SU(2) irreducible representations. The bottom left panel encloses an image that corresponds exactly, i.e. with probability 1, to a "L".

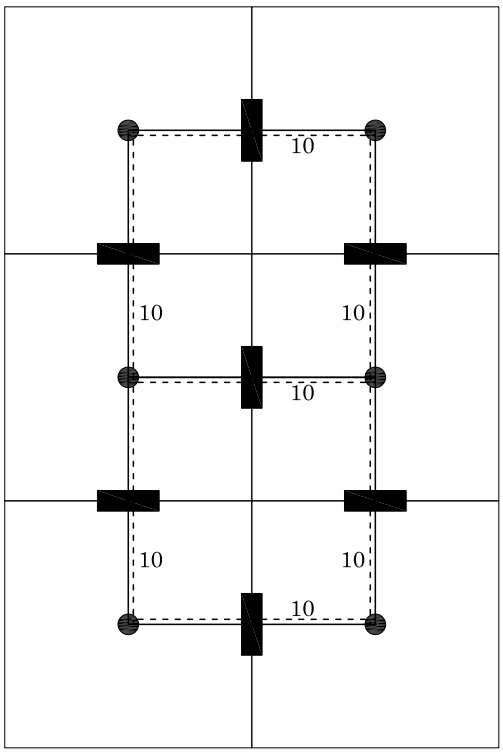


Fig. 4. The maximal graph, which encloses all the possible sub-graphs supporting the training samples' cylindrical functions, where 10 denotes the maximal value for the spin color j_a.

the topology of the system, and infers this by integrating over the geometries of the system, therefore selecting a geometry that extremizes the transition amplitude.

4. Associating spin-networks to images

A fundamental feature of the definition of TQNNs is that input and output states are spin-networks and, more generally, cylindrical functions of the Hilbert space in the holomorphic representation. It is, therefore, crucial to have well-determined rules to associate spin-networks to the input data. We suppose to have a pixeled image whose shades of gray vary in [0, 1]. We let the nodes of our spin-network coincide with the centers of the pixels. For each node N, we let j_a denote twice the spin j representation of SU(2), i.e. $j_a = 2j$, where j_a is the integer part of ten times the shade of gray of the pixel whose center is N — then there will be 11 different integer values, when including 0, j_a with $a = 0, \ldots, 10$, and 11 semi-integer values of j, from $j = 1/2$ to 5. Then, we consider the von Neumann neighborhood of a node N, and for a node N' in the neighborhood we join the two nodes by $j_{ab} = \min\{j_a, j_b\}$, where a and b are the associated (re-scaled) shade of gray of the pixels of N and N', respectively. We apply the Jones–Wenzl projector (Kauffman et al., 1994) to the representation corresponding to j_{ab} in order to symmetrize it, so to provide all the possible spin irreducible representations with $0 \leq j \leq 5$.

To better elucidate the previous scheme we consider the specific situation of (handwritten) letters with 3×2 pixels and the shades of gray ranging in the interval [0, 1] in decimals, where 0 corresponds to white, while 1 corresponds to black. By construction, the associated spin-networks obtained will have six nodes, each centered in one of the pixels. For example, four instances of the letter "L" and their corresponding spin-networks are given in Fig. 3, where we use rectangular boxes to denote the Jones–Wenzl projector applied to the edges (corresponding to SU(2) representations) joining two nodes. In the case of the top left panel in Fig. 3, proceeding counterclockwise from the left top pixel, the encountered set of shades of gray is set to be {0.4, 0.5, 0.6, 0.5, 0.0, 0.0}. A slightly different case is represented in the top right panel of Fig. 3 for which the string of numbers is {0.4, 0.5, 0.6, 0.5, 0.1, 0.2}. The ideal case, corresponding to the spin-network state that perfectly captures the letter L, with a probability $|\mathcal{A}|^2 = 1$, is given by {1.0, 1.0, 1.0, 1.0, 0.0, 0.0} $\equiv$ L, and is represented on the bottom left panel of Fig. 3. Finally, the bottom right panel represents an undetermined case captured by the string of numbers {0.3, 0.4, 0.3, 0.2, 0.1, 0.2}. We shall notice that these are all nothing but "colored" sub-graphs that can be recovered from a maximally connected graph, the one pictured in Fig. 4, by removing fundamental representation strands along with the links.

5. The perceptron in the semi-classical limit

We consider now our topological version of the notion of perceptron and show that in the semi-classical limit we obtain an object that resembles traditional perceptrons closely. A detailed description of the semi-classical limit derived from the general TQFT formalism used in Section 3 is provided in Appendix C, and a dictionary mapping DNN concepts to TQNNs is given in Appendix D. The first step toward adapting TQNNs to the setting of perceptrons is to define an algorithmic way to associate spin-networks to the input vectors in $\mathbb{R}^n$ constituting the dataset. Let N be a natural number that is large compared to the magnitudes of the entries of the vectors of the dataset. Given a vector $\bar{x}$, we

construct a spin-network $\Gamma_{\bar{x}}$ associated to $\bar{x}$ as follows. We introduce a node which is labeled by 0, and for each $i = 1, 2, \ldots, n$ we add a node, labeled by the index i of the corresponding entry of $\bar{x}$. As in the case of Section 4, we color the node labeled by 0 with the spin representation j_N, while each node i is colored by $[x_i]$, the closest integer rounding x_i. Then, for each i we introduce an edge connecting 0 and i, which is labeled by a spin $j_{0i} = N + [x_i]$ representation. Finally, we symmetrize the edges by applying the Jones–Wenzl projector, indicated diagrammatically by placing a black box on the connecting edges. Observe that we do not introduce, in this context, links between nodes i and j with $i, j \neq 0$. Now, the weights of the perceptron are vectors $\bar{w} \in \mathbb{R}^n$ similarly to the inputs $\bar{x}$ of the dataset. We follow the same procedure above to introduce a spin-network $\Gamma_{\bar{w}}$ of weights.

Since we have chosen N much larger than the actual range of the data entries $\bar{x}$ (i.e. the hypercube $[-M, M]^n$ where M is the maximum magnitude that the entries of the dataset reach, has $M \ll N$), it follows that we can adopt the large spin j_{0i} limit as described in Appendix C, for which transition amplitudes are computed as

$$\mathcal{A}_{\prod_i H_{0i},\bar{w}} = \langle \Psi_{\Gamma_{\bar{x}},H_{0i}} | \Psi_{\Gamma_{\bar{w}},j_{\bar{w}},\iota_n} \rangle = \prod_i \Delta_{j_{0i}} e^{-\frac{(j_{0i}-\bar{j}_{0i})^2}{2\sigma_{0i}^2}} e^{-\iota \xi_{0i} j_{0i}}, \tag{1}$$

where H_{0i} denote $\mathrm{SL}(2,\mathbb{C})$ elements labeling boundary data in the asymptotic limit — see e.g. Appendix C.

The analogy with classical perceptrons is as follows. A perceptron trains a function f whose weight vector $\bar{w}$ determines the output according to the rule $f(\bar{x}) = 1, 0$ depending on whether $\bar{w} \cdot \bar{x} > \theta$ or not, respectively, for some threshold θ, and where $\cdot$ indicates the inner product of $\mathbb{R}^n$. In fact, usually, a bias appears in the perceptron formulas, but this can be encoded among the weights as well, so we will omit to refer to it. In our topological version above, the amplitude $\mathcal{A}_{\prod_i H_{0i},\bar{w}}$ is obtained by the inner product of spin-network states associated to inputs $\bar{x}$ and weights $\bar{w}$. The transition amplitude $\mathcal{A}_{\prod_i H_{0i},\bar{w}}$ is a complex number whose modulus square is between 0 and 1, so that by applying a Heaviside step function H, centered at some threshold value θ, to $|\mathcal{A}_{\prod_i H_{0i},\bar{w}}|^2$ we obtain a TQNN implementation of the concept of perceptron. Training a topological perceptron would amount to optimizing weights $\bar{w}$, and $\mathrm{SL}(2,\mathbb{C})$ elements H_{0i} with respect to a predetermined target.

Similar reasoning applied to feedforward neural networks (i.e. multilayer perceptrons) can be implemented as well, by using the fact that TQFTs are defined via functorial constructions that allow us to compose an arbitrary number of H which are the basic computational units in feedforward neural networks. Note that, in this setting, the "semi-classical" nature of QNNs with fixed layers and fixed connections, and hence classical constraints on the entanglement between qubits, also becomes clear: such systems are effectively designed only on particular paths through the input/output equivalent TQNN. We see, therefore, that TQNNs are versatile objects that can be trained and utilized for classification problems in different ways. Moreover, through the notion of semi-classical limit, they provide a way of interpreting artificial neural networks in the context of TQNN theory.

6. Handwritten letter recognition

We may discuss possible experimental applications of the new framework we introduced. First, we consider the theory introduced in this article, applied to a concrete example, where we work in the semi-classical limit. It is worth mentioning that we take into account hidden layers as in the "Feedforward" step of the algorithm of Appendix C. This consists of interpolating among

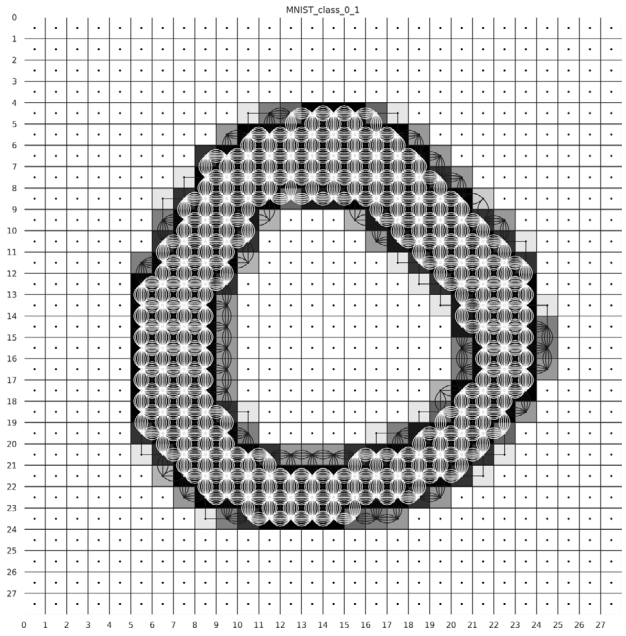


Fig. 5. A specific graph, representing the number 0, within the case employing 28×28 pixels.

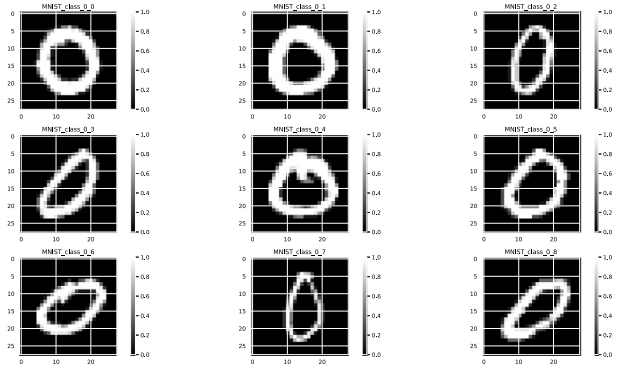


Fig. 6. Several samples of the number 0, extracted from the MNIST database, to be used during the training process.

intermediate states, on which a complete summation is taken into account through Eq. (B.8), and which are supported only on a restricted set of sub-graphs. The functoriality of TQNN in this sense is here fundamental, as Eq. (B.8) encodes precisely the composition property of cobordisms, preserved by TQFTs. We can imagine the hidden layers acting as filtering specific patterns over others. Indeed, what the hidden layers do is to impose a selection over the intermediate graphs $\partial\mathcal{C}_n$, and hence the 2-complexes that interpolate among these latter ones. Internal summation over the irreducible representations of SU(2), namely variation of the metric properties of the QNN states, then individuates all the possible sub-graphs contained in $\partial\mathcal{C}_n$, i.e. this corresponds to a variation of the topological features of the 1- and 2-complex structures (Fig. 5). Applying the definition of cobordisms and functoriality implicit in the definition of a TQNN as a type of TQFT, implementing different layers as described above simply coincides with computing transition amplitudes through middle steps in the computation, as prescribed by Eq. (B.7).

The experiment utilizes the MNIST database (Fig. 6) which is the machine learning benchmark for hand-written digit recognition. The dataset contains the gray-scale images of hand-written digits. The fact that all images in the dataset have identical

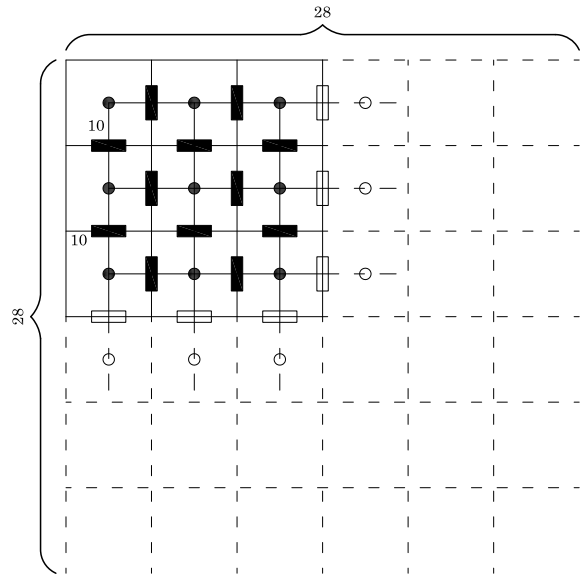


Fig. 7. The maximal graph, which encloses all the possible sub-graphs supporting the training samples' cylindrical functions for the case 28 × 28 pixels.

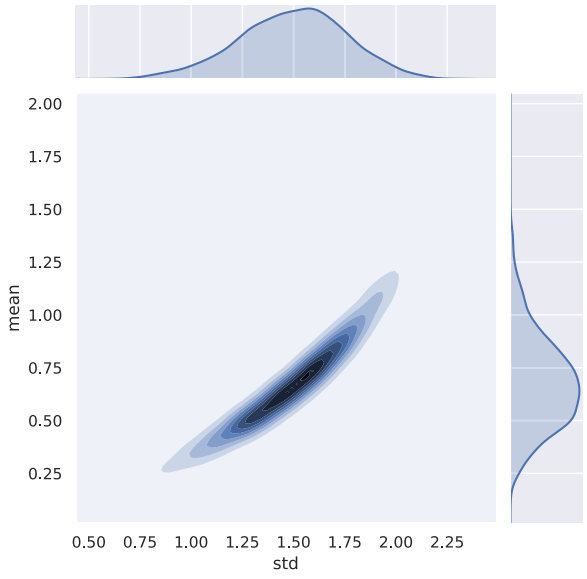


Fig. 8. Marginalized plots for the estimated mean values and standard deviation of the irreducible representations associated to the links of the spin-networks states. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

dimensions, which is 28 × 28 pixels, see Fig. 7, implies that the knowledge representation graph (Scarselli et al., 2009; Sowa, 2006), encoding both raw image data and the relationships between the values of each pixel, can be constructed from any image in the dataset. After the translation of the knowledge representation graph, the parameters for each class of digits are obtained using class prototyping (Rosch, 1983, 1988). The latter consists of averaging the spin colors appearing in the training set of MNIST, in order to determine a representative spin-network whose transition with respect to input data provides the classification probability (hence the label), and it is coherent with the choice of utilizing the semi-classical formalism. The constructed input array, namely the knowledge graph encoding topological information, is helping us to determine the likelihood of "input". Alternatively, any optimization technique like gradient descent can be applied to learning the class prototype of a specific set of spin-networks states. In Fig. 8 we report the mean values of the standard deviations of the j-spin colorings corresponding to irreducible representations associated with the spin-networks.

7. A new working hypothesis

As a consequence of the previous discussions, we propose a working hypothesis that the learning process of DNNs shall be interpreted within an extended framework, which follows the very same axioms of quantum mechanics and quantum topology, through the formulation of TQFT. In other words, we see a TQNN as a quantization of a DNN whose $\hbar \to 0$ limit recovers the classical case. In the learning process of a TQNN, the substantial feature that a TQNN learns is the selection of relevant geometries in the partition function that determines transition amplitudes utilized to classify. The main idea that constitutes the backbone of the present framework is that DNNs should be addressed at the TQNNs level. Training examples or tests samples will be captured by the spin representations of the TQNN quantum states, which are superpositions of the boundary Hilbert space elements.

Transition amplitudes will return the probability of a state as being in a certain spin-network basis state. The generic boundary states are characterized by two classes of parameters, which we refer to as topological and metric parameters: the former is captured by the topology of the graph, hence by the topological invariant (linking and knotting) quantum numbers, while the latter is captured by the spin of the representation itself. Pertaining to the topological parameters, the information provided by the training samples, together with the definition of training error in terms of the internal product of boundary quantum states, substantially determines the structure of the bulk, and therefore the functor that determines transition amplitudes, in the learning process. We argue that the topological parameters are enough to learn the classifier, namely the TQNN 2-complex that provides the functorial structure of the TQNN, playing a similar role to the frequency threshold in the photoelectric effect. Whenever not enough information about the topology is specified by the training data, any TQNN 2-complex with enough topological internal structure will be selected. This might be considered as a TQNN counterpart of a similar phenomenon in the theory of TQFT, and its relations to Chern–Simons theory and the Jones polynomial. In fact, celebrated results of Witten (1989) have shown that the partition function associated with the action corresponding to Chern–Simons theory is independent of the metric, although the action itself is not. We have incurred in a similar situation, and we argue that the notion of generalization in TQNN theory and, as a limit, in DNN theory, lies precisely here. Although the partition functions that are used to determine the transition amplitudes are topological (hence the name TQNN), what is learned during the learning process is what geometries to associate to given classified patterns.

8. Conclusions

Moving from the perspective of TQFT, we have defined the concept of "Topological Quantum Neural Network" and show that classical DNNs can be seen as a subcase of TQNNs and emerge in a coherent group theoretical sense as a limit of TQNNs. This allowed us to establish a dictionary translating a number of ML key concepts into the terminology of TQFT. More importantly, we have proposed a framework that provides a working hypothesis for understanding the generalization behavior of DNNs.

The novelty of our approach, particularly when compared to recent studies in the literature (Beer et al., 2020; Farhi & Neven,

2018), stands in taking into account fully, for the first time, the truly topological structure of graphs and 2-complexes on which the TQNN states are supported. Indeed, ours is not only a pictorial representation, in terms of graphs, of product states belonging to the total Hilbert space (Fock space) of the theory. Instead, what we have developed is a scheme that allows associating ML concepts to topologically invariant features of the graphs (inter-connectivity of edges, linking and knotting numbers, topological invariants on 2-complexes) and 2-complexes involved in the TQNN construction.

A number of further lines of research could be pursued starting from our approach:

1. Providing empirical results concerning the working hypothesis previously described so to corroborate the claim that the notion of generalization introduced in this article is consistent;
2. Defining new complexity measures more appropriate to the framework we described and adequate to explain the behavior of over-parametrized models such as DNNs. It would also be of interest to pursue deeper experimentation with a variety of benchmark data sets, so to relate complexity measures to concrete examples;
3. Introducing the notion of "time" into the architecture by modeling phenomena of the cortical plasticity such as firing rate or spike timing, see Sjöström et al. (2001). In particular, this perspective implies the necessity of using TQFTs that have one extra dimension with respect to the concrete ones that have been used in this article. The basic theory does not change, in that the notion of TQNNs does not require fixing a specific dimension in the cobordism category, but the corresponding algebraic/analytical machinery certainly becomes heavier.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Antonino Marciano reports financial support was provided by National Science Foundation of China. Antonino Marciano reports financial support was provided by Shanghai Municipal Government. Niels Gresnigt reports financial support was provided by Foundation of the Jiangsu Higher Education Institutions of China Programme. Matteo Lulli reports financial support was provided by National Science Foundation of China. Emanuele Zappala reports financial support was provided by Estonian Research Council.

Acknowledgments

A. Marcianò acknowledges support by the National Science Foundation of China, through the Grant No. 11875113, the Shanghai Municipality, through the Grant No. KBH1512299, and by Fudan University, through the Grant No. JJH1512105. N. Gresnigt acknowledges support through Foundation of the Jiangsu Higher Education Institutions of China Programme Grant No. 19KJB140018 and Xi'an Jiaotong-Liverpool University Grant No. REF-18-02-03. M. Lulli acknowledges the support from National Science Foundation of China Grant No. 12050410244. E. Zappala was supported by the Estonian Research Council through the grant MOBJD679.

Appendix A. Topological Quantum Field Theory

We provide in this section of the appendix a deeper introduction to Topological Quantum Field Theory (TQFT), spin-network (boundary) states, and (bulk) 2-complexes functorial evolution of boundary states.

A.1. Classical phase-space and spin-network states

The theory is the principal SU(2)-bundle over a D-dimensional base manifold $\mathcal{M}$. The SU(2)-connection A realizes the parallel transport among infinitesimally closed fibers of the principal bundle. The parallel transport along a finite path γ connecting any two points of $\mathcal{M}$ is individuated by

$$H_\gamma[A] = \mathrm{P}\, e^{\int_\gamma A}\,, \tag{A.1}$$

which denotes the path ordered exponential P of the integrated flux of A along γ. The holonomy then provides a group element $g \in \mathrm{SU}(2)$. The trace of the holonomy along a closed path (a loop α) can be expanded, taking into account a squared loop of infinitesimal edge ϵ, as

$$\lim_{\|\alpha\|\to 0} W_\alpha[A] = \mathbb{1} - \epsilon^2 F[A] + \dots\,, \tag{A.2}$$

where $\|\alpha\|$ denotes the measure of the loop α, and $F[A] = dA + A \wedge A$ is the field strength, or curvature, of the connection A. The connection A is both a *1*-form on $\mathcal{M}$ — indeed, its curvature is a *2*-form over $\mathcal{M}$, since the differential d is one-form — and an element of the $\mathfrak{su}(2)$ algebra. Thus, it admits the decomposition over the generators τ^a, with $a = 1, 2, 3$ indices in the adjoint representation of the algebra. Consequently, the connection A and its curvature $F[A]$ acquire the dependence on the internal indices, respectively $A = A^a\tau^a$ and $F^a[A] = dA^a + \epsilon^{abc}A^b \wedge A^c$, the Levi-Civita symbol ϵ^{abc} providing the structure constants of SU(2) and the Einstein convention of summing repeated indices is intended.

A TQFT can be introduced considering the topological action associated to the Lagrangian density function

$$\mathcal{L}[A] = B^a \wedge F^a[A] = \mathrm{Tr}[B \wedge F[A]]\,, \tag{A.3}$$

where the B field denotes a $\mathfrak{su}(2)$ algebra valued D-form, which is the canonically conjugated momentum to the connection A, and the trace over the generators of the algebra is normalized to the identity and yields $\mathrm{Tr}[\tau^a\tau^b] = \delta^{ab}$. The phase-space variables A and B can be then paired in a symplectic construction, imposing the Poisson brackets

$$\{A^i_a(x_1), B^b_j(x_2)\} = \delta^b_a\,\delta^i_j\,\delta(x_1, x_2)\,, \tag{A.4}$$

with $i = 1, \dots, D$ space indices over the dimensions of $\mathcal{M}$.

Holonomies realize the smearing of the configuration space variables, i.e. the connections A, along the paths γ.

Similarly, the smearing of the frame fields B can be implemented by substituting their fluxes calculated through the surfaces Σ of co-dimension 1 that crosses the paths γ at least in one point, namely

$$B_\Sigma = \int_\Sigma B \cdot n\,, \tag{A.5}$$

where n is normal to the surface Σ and the dot denotes contraction of indices. For example, since the dimension of the path γ is 1, its co-dimension 1 surface in a *3D* ambient space will be a *2D* surface.

The theory we just introduced retains what is called a gauge symmetry, namely a symmetry under internal transformations, which individuates an equivalence class that describes an observer. These are instantiated by transformations involving generic group elements $g \in \mathrm{SU}(2)$, i.e.

$$A \to A_g = g^{-1}Ag + g^{-1}dg\,, \tag{A.6}$$

and

$$B \to B_g = g^{-1}Bg\,. \tag{A.7}$$

It is trivial to check that the action (A.3) is invariant under the joined action of (A.6)–(A.7). The infinitesimal expansion of finite

transformation rules (A.6)–(A.7) can be cast at the $\mathfrak{su}(2)$ algebraic level, through the infinitesimal expansion of a group element around the identity, i.e. $g \simeq \mathbb{1} + \alpha^a \tau^a + \dots$. This individuates an infinitesimal transformation

$$\delta_\alpha B = [B, \alpha]\,, \qquad \delta_\alpha A = \mathcal{D}_A \alpha\,, \tag{A.8}$$

where the commutators $[\,,\,]$ denote the adjoint action of the algebra. The generators of the algebra appear in $B = B^a \tau^a$ and $\alpha = \alpha^a \tau^a$, while $\mathcal{D}_A$ denotes the covariant SU(2) derivative $\mathcal{D}_A := d + A$.

Another symmetry, which is relevant for the definition of TQFT, is shift symmetry. This is actually ensuring the theory under consideration to be topological, as it is straightforward to recognize by looking at

$$B \to B + \delta_\eta B\,, \qquad \delta_\eta B = \mathcal{D}_A \eta\,, \tag{A.9}$$

and

$$A \to A + \delta_\eta A\,, \qquad \delta_\eta A = 0\,, \tag{A.10}$$

where η is any arbitrary infinitesimal 0-form (a function). Under the infinitesimal transformations (A.9)–(A.10), the variation of the action of the theory $\mathcal{S}[A] = \int_{\mathcal{M}} \mathcal{L}[A]$, namely

$$\delta_\eta \mathcal{S}[A, B] = \mathcal{S}[A + \delta_\eta A, B + \delta_\eta B] - \mathcal{S}[A, B]\,, \tag{A.11}$$

vanishes, due to the Bianchi identity $\mathcal{D}_A F[A] = 0$. This latter identity appears in the variation of the action due to an integration by part:

$$\begin{aligned} \int_{\mathcal{M}} \mathrm{Tr}[(B + \delta_\eta B) \wedge F[A + \delta_\eta A]] = \\ \int_{\mathcal{M}} \mathrm{Tr}[(B + \mathcal{D}_A \eta) \wedge F[A]] = \\ \int_{\mathcal{M}} \mathrm{Tr}[B \wedge F[A]] - \int_{\mathcal{M}} \mathrm{Tr}[B + \wedge \eta \mathcal{D}_A F[A]] = \\ \int_{\mathcal{M}} \mathrm{Tr}[B \wedge F[A]]. \end{aligned} \tag{A.12}$$

This symmetry is often referred to as a "gauge symmetry" of the *BF* theory, which individuates a class of equivalence among physical solutions that differ by this transformation.

On the other hand, the equation of motions are specified by the variation of the action with respect to the phase-space fields:

$$\mathcal{D}_A B = 0\,, \qquad F[A] = 0\,. \tag{A.13}$$

Solutions are then "flat", or with zero curvature, i.e. $F[A] = 0$, while the frame fields satisfy the Gauß constraint $\mathcal{D}_A B = 0$, which generates the gauge transformations. Locally, by the topological shift symmetry, any frame field B that satisfies the Gauß constraint can be recast as $\mathcal{D}_A \eta$, for some η. This is true as locally closed forms are exact, and continue to satisfy the Gauß-constraint. This implies that locally the solutions of the equations of motion belong to the same equivalence class, modulo gauge transformations, and shift symmetry transformations. Since these can be mapped into vanishing configurations, this argument finally shows that there are no propagating degrees of freedom in *BF* theories, namely that these theories are topological.

A.2. Graph-kinematics

As the last step before proceeding to the definition of the 1- and 2-complexes, we introduce the irreducible representations of the group, the so-called "spin" numbers, and the inter-twiner numbers, depending on the SU(2) recoupling theory. For this purpose, we remind that in this case holonomies over a path γ are group elements of SU(2), and thus undergo the transformations

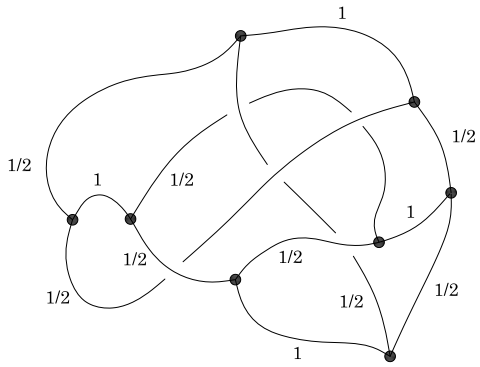


Fig. A.1. A graph with tri-valent nodes colored under SU(2).

$$H_\gamma[A] \to g^{-1}_{s(\gamma)} H_\gamma[A] g_{t(\gamma)}, \tag{A.14}$$

where $g_{s(\gamma)}$ and $g_{t(\gamma)}$ are group elements assigned respectively to the source and the target of an oriented path γ. For SU(2), irreducible representation of holonomies are provided by the Wigner matrices and labeled by the semi-integer j-spin numbers, namely

$$D^{(j_\gamma)}(U_\gamma)\,, \qquad U_\gamma \equiv H_\gamma[A]\,, \tag{A.15}$$

SU(2) intertwiners are expressed as the (group elements) integrals of a number of copies of irreducible representations (Wigner matrices). As a compact group, SU(2) is endowed with a Haar measure (invariant under gauge transformations and coordinate reparametrizations) that enables the definitions of the intertwiner invariant tensors. These latter quantities can be thought to be associated with the nodes where endpoints (target points) and origins (source points) of the paths γ intersect. A collection of n path $\gamma_1, \gamma_2 \dots \gamma_n$ intersecting at their target and source points (nodes) provides a graph Γ. The internal indices of the Wigner matrices integrated ensure gauge-invariance through the contraction with the holonomies flowing across the node. Integrating with the Haar measure the irreducible representations of the holonomies, the target or source points of which cross at the node, and which are labeled by the spin $j_{\gamma_1}, j_{\gamma_2} \dots j_{\gamma_n}$, provided the expression for the inter-twiner

$$v_\iota = \int_{\mathrm{SU}(2)} dU\, D^{(j_{\gamma_1})}(U) D^{(j_{\gamma_2})}(U) \dots D^{(j_{\gamma_n})}(U)\,, \tag{A.16}$$

having again suppressed all the (intertwiner and Wigner matrices) representation indices.

A collection of holonomies, the internal indices of which are contracted with the intertwiners defined by integration of the group elements at the nodes, defines a spin-network state. In terms of its constituents, the holonomies and the intertwiners, a spin-network state cast as

$$\psi_{\Gamma,\{j_l\},\{\iota_n\}}[A] = \left(\bigotimes_{n \in \Gamma} v_{\iota_n} \right) \cdot \left(\bigotimes_{\gamma_l \in \Gamma} {}^{j_{\gamma_l}}D\,(U_{\gamma_l}[A]) \right)\,, \tag{A.17}$$

where the dot denotes the contraction of internal indices, and $l = 1, \dots n$ label the n paths γ that compose the graph Γ (Fig. A.1).

SU(2) spin-network states are equipped with a Haar measure, which ensures invariance under gauge transformations and diffeomorphisms (coordinate reparametrizations) on the base manifold $\mathcal{M}$, of the internal product

$$\langle \Psi_{\Gamma', j'_\gamma, \iota'_n}[A] |\, \Psi_{\Gamma, j_\gamma, \iota_n}[A] \rangle = \delta_{\{\Gamma'\},\{\Gamma\}} \delta_{j'_\gamma, j_\gamma} \delta_{\iota'_n, \iota_n} \tag{A.18}$$

Invariance under diffeomorphisms, which is expressed by the Kronecker delta between classes of equivalence of graphs endowed with the same topology, namely $\{\Gamma\}$, instantiates the symmetry under elastic transformations, rendering the graph structure truly topological. In this study, graphs Γ are also referred to as 1-complexes.

A.3. Graph-dynamics

A concept of dynamics requires the definition of boundary states (1-complexes), the quantum evolution of which is provided by relative transition amplitudes. These are captured by the path integral (realizing the vacuum-vacuum transition, with no underlying graph structure) and the expectation values in its measure. It is convenient to introduce the mathematical concept of 2-complex $\mathcal{C}$. A 2-complex $\mathcal{C}$ is composed by edges e departing or ending either at nodes $n \in \Gamma$ or at vertices v internal to $\mathcal{C}$, by faces f bounded by either links γ or internal edges e, and vertices v where edges cross. We are going to show how to associate a functor — either the partition function $Z_{\mathcal{C}}[U_{\gamma_l}]$, or the expectation value of boundary state in the path-integral associated to the topological theory — to a 2-complex $\mathcal{C}$ endowed with boundary group elements U_{γ_l}.

The partition function for the BF model over a SU(2)-bundle is specified by the expression

$$\mathcal{Z} = \int \mathcal{D}A\mathcal{B}\, e^{\imath \int_{\mathcal{M}} \mathrm{Tr}[B\wedge F]} = \int \mathcal{D}A'' \delta(F)''. \tag{A.19}$$

where in the last equality we introduced a Dirac delta measure on the space of flat connections. This is understood (Baez, 2000) from smearing the phase-space variables and then casting the partition function as

$$\mathcal{Z}(\Delta) = \int_{\mathfrak{su}(2)^E} \prod_{e\in E} dB_e$$
$$\times \int_{\mathrm{SU}(2)^{E^*}} \prod_{e\in E^*} dU_e\, e^{\imath \sum_{e\in\Delta} \mathrm{Tr}[\mathrm{B_e F_e}]}, \tag{A.20}$$

where Δ denotes the triangulation of the manifold $\mathcal{M}$ — this allows to introduce a simplicial complex Δ^* that is dual to the triangulation $\Delta - E$ denotes the set of edges e of the triangulation Δ, and E^* the set of edges $e*$ of the dual simplicial complex Δ^*. Furthermore, in the expression (A.20) we have been using the natural definition of the curvature, which is expressed by the product of group elements U_{e*} associated to the links around the boundary $\partial f*$ of a dual face $f*$ (thus associated with the dual face itself):

$$U_{f*} = \prod_{e^*\in\partial f*} U_{e*}\,. \tag{A.21}$$

where $F_e = \ln U_{f^*}$, namely individuates a Lie algebra element that entails the discretization of the connection field curvature on the edges e of Δ. Integration over the algebra elements B_e provides the expression for the Dirac delta on the product of group elements that realizes the smearing of the curvature, namely

$$\int_{\mathfrak{su}(2)^E} \prod_{e\in E} B_e\, e^{\imath \sum_{e\in\Delta} \mathrm{Tr}[\mathrm{B_e F_e}]} = \delta(e^{F_e}) = \delta(U_{f*})\,. \tag{A.22}$$

The partition function then casts

$$\mathcal{Z}(\Delta) = \int_{\mathrm{SU}(2)^{E^*}} \prod_{e\in E^*} dU_{e*} \prod_{f*} \delta(U_{f*})\,. \tag{A.23}$$

This formula finally admits a re-manipulation in terms of the irreducible representation of SU(2), which thanks to the Peter–Weyl expansion, is provided by Plancherel formula

$$\delta(U_{f*}) = \sum_{j_{f*}} \Delta_{j_{f*}} \chi^{j_{f*}}(U_{f*})\,, \tag{A.24}$$

where j_{f*} denote half-integer numbers that label SU(2) irreducible representations, $\Delta_j = (2j+1)$ the dimension of these latter, and $\chi^j(U) = D^j(U)^\alpha_\alpha$ is the character of the group element $U \in SU(2)$, i.e. the trace of a Wigner matrix over the internal indices α in the representation Hilbert space. Then the partition function recasts

$$\mathcal{Z}(\Delta) = \sum_{j_{f*}} \int_{\mathrm{SU}(2)^{E^*}} \prod_{e\in E^*} dU_e \prod_{f*} \mathrm{Tr}[D(\prod_{e^*\in\partial f*} U_{e*})]\,, \tag{A.25}$$

which depends only on the recoupling theory of SU(2), and retains a dependence on the dimension of the manifold $\mathcal{M}$, in which both the graphs Γ and the 2-complex $\mathcal{C}$ are merged. Thus, we can identify the no-boundary path-integral amplitude $\mathcal{Z}(\Delta)$ with the no-boundary functor $\mathcal{Z}_{\mathcal{C}}$, i.e.

$$\mathcal{Z}_{\mathcal{C}} = \mathcal{Z}(\Delta) \tag{A.26}$$

where there is no dependence on the boundary group elements.

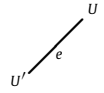
Appendix B. Topological Quantum Neural Networks

We provide a detailed account of the formulation of Topological Quantum Neural Networks (TQNN), their functorial dynamics, and their training algorithm.

B.1. Transition amplitudes

The scheme of computing the transition amplitude between initial and final states is obtained following an association path (Rovelli, 2011). This is schematically described as follows.

- We integrate either twice over each internal edge in the bulk,[1] or once over an adjacent couple of group elements, assigned to either internal edges or vertices:

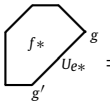


$$\Longrightarrow \quad \int_{\mathrm{SU}(2)} dU_{s_e} \int_{\mathrm{SU}(2)} dU_{t_e}\,; \tag{B.1}$$

- We integrate over each couple of adjacent group element, assigned to either to a face or to an internal edge:

$$\Longrightarrow \quad \int_{\mathrm{SU}(2)} dU_{e*}\ \chi^{j_f}(U_f)\,; \tag{B.2}$$

- We sum over each face $f*$ and associate the element



$$\Longrightarrow \quad \sum_{j_{f*}} \Delta_{j_{f*}}\, \chi^{j_{f*}}\Big(\prod_{e*\in\partial f} U_{e*}\Big)\,; \tag{B.3}$$

- We drop, at each vertex, an integral $\int_{\mathrm{SU}(2)} dU_{v(e)}$, which appears as redundant in (B.1).

The functor $\mathcal{Z}(U_l)$ provides the transition operator between boundary states, and gives the algebraic counterpart of cobordisms between boundary manifolds. It clearly depends on the boundary group elements and it is written as

$$\mathcal{Z}_{\mathcal{C}}(U_l) = \int_{\mathrm{SU}(2)^{2(E-L)-V}} dU_{v(e)} \int_{\mathrm{SU}(2)^{\mathcal{V}-L}} dU_f$$
$$\times \prod_f \mathcal{K}_{f*}(U_{e*}, U_f)\,, \tag{B.4}$$

[1] For bulk we intend any 2-complex structure, without boundary. Therefore $Z_{\mathcal{C}}[U_{\gamma_l}]$ acts in a functorial way on the boundary states, which are 1-complexes, i.e. colored graphs Γ composed by a collection of paths γ and nodes where the paths intersect, to which are assigned respectively holonomies and intertwiners.

where $\mathcal{K}_{f*}(U_{e*}, U_f)$ denotes the "face amplitude"

$$\mathcal{K}_{f*}(U_{e*}, U_f) \equiv \sum_{j_{f*}} \Delta_{j_{f*}} \chi^{j_{f*}}\Big(\prod_{e*\in\partial f} U_{s(e)} U_{e*} U_{t(e)}^{-1}\Big) \prod_{e*\in\partial f} \chi^{j_{f*}}(U_f)\,. \tag{B.5}$$

Taking into account a 2-complex without boundary, (B.4) reduces to the partition function

$$\mathcal{Z}_{\mathcal{C}} = \int_{\mathrm{SU}(2)^{2E-V}} dU_{v(e)} \int_{\mathrm{SU}(2)^{\mathcal{V}}} dU_f \sum_{j_{f*}} \prod_f \Delta_{j_{f*}} \times \chi^{j_{f*}}\Big(\prod_{e*\in\partial f} (U_{s(e)} U_{e*} U_{t(e)}^{-1})\Big) \prod_{e*\in\partial f} \chi^{j_{f*}}(U_f)\,, \tag{B.6}$$

where $\mathcal{V}$ is the sum of the valences of the faces of $\mathcal{C}$. Differently than in (B.4), the expression in (B.6) provides the amplitudes of probability for the output of the transition among states. This coincides to the process of "capping" the boundaries described before, and gives a partition function which is a topological invariant of manifolds. As observed before, for the example of TQFT in dimension 2, this is an endomorphism of the ground field $\mathbb{C}$.

B.2. Training algorithm

We are finally able to specify the training algorithm of the model as follows.

1. **Initialize**:
   Associate, between boundary states that are supported on disjoint graphs
   $\{\Gamma_{\mathrm{in}},\ \Gamma_{\mathrm{out}}\,;\, \partial\mathcal{C} = \Gamma_{\mathrm{out}} \cup \Gamma_{\mathrm{in}}\}$, the functorial evolution
   $\mathcal{Z}_{\mathcal{C}}(\{U_l\,;\, l \in \mathcal{C}\}, \{\bar{j}_l\})$,
   where $\{\bar{j}_l\}$ denote a set of parameters to be fitted in the learning process.
2. **Feedforward**:
   2a compose a functor $\mathcal{Z}_{\mathcal{C}}(\{U_l\,;\, l \in \mathcal{C}\}, \{\bar{j}_l\})$, which is supported on a 2-complex $\mathcal{C}$, with a series of 2-complexes interpolating among either the intermediate hidden layers graphs or the boundary states' graphs. For P hidden layers, labeled by $p \in P$, we have the decomposition $\mathcal{C} = \mathcal{C}_1 \cdots \cup \mathcal{C}_p \cup \mathcal{C}_{P+1}$. Therefore
   $$\mathcal{Z}_{\mathcal{C}}(\{U_l\,;\, l \in \mathcal{C}\}, \{\bar{j}_l\}) = \mathcal{Z}_{\mathcal{C}_1}(\{U_{l_{\mathrm{in}}}\,;\, l_{\mathrm{in}} \in \Gamma_{\mathrm{in}}\}, \{\bar{j}_{l_{\mathrm{in}}}\}) \cdots \times \mathcal{Z}_{\mathcal{C}_1}(\{U_{l_{\mathrm{out}}}\,;\, l_{\mathrm{in}} \in \Gamma_{\mathrm{out}}\}, \{\bar{j}_{l_{\mathrm{out}}}\})\,, \tag{B.7}$$
   where the dot denotes the integration over the group elements assigned to the interpolating graphs supporting the hidden layer structures. This, in fact, encodes functoriality of $\mathcal{Z}$, since it respects the composition of intermediate manifolds.
   2b integrate over the group elements U assigned to the hidden layer graphs, so to trace them out:
   $$\mathcal{Z}_{\mathcal{C}_1}(\{G\}) \cdot \mathcal{Z}_{\mathcal{C}_2}(\{H\}) = \int_{\mathrm{SU}(2)} \prod dU\ \mathcal{Z}_{\mathcal{C}_1}(\{U\}, \{G\})\ \mathcal{Z}_{\mathcal{C}_2}(\{U\}, \{H\}) = \mathcal{Z}_{\mathcal{C}_1 \cup \mathcal{C}_2}(\{G\}, \{H\})\,. \tag{B.8}$$
   This property is often referred to as a cobordism of the functorial structure.
3. **Classify**:
   Introduce $H_l \in \mathrm{SL}(2, \mathbb{C})$, encoding the information on the set of parameters $\{\bar{j}_l\}$; by the aforementioned combinatorics, associate to the 2-complex $\mathcal{C}$ the transition amplitude
   $$\mathcal{Z}_{\mathcal{C}}(H_l) = \int_{\mathrm{SU}(2)^{2(E-L)-V}} dU_{v(e)} \int_{\mathrm{SU}(2)^{\mathcal{V}-L}} dU_f \times \prod_f \mathcal{K}_{f*}^{t_{f*}}(U_{e*}, U_f)\,, \tag{B.9}$$
   where the heat kernel propagator, encoding the information about the parameter $\{j_l\}$ through the SU(2) coherent group elements,[2] see Bianchi et al. (2010b), acquires the expression
   $$\mathcal{K}_{f*}^{t_{f*}}(U_{e*}, U_f) \equiv \sum_{j_{f*}} \Delta_{j_{f*}}\, e^{-j_{f*}(j_{f*}+1)\frac{t_{f*}}{2}} \times \chi^{j_{f*}}\Big(\prod_{e*\in\partial f} (U_{s(e)} U_{e*} U_{t(e)}^{-1}) H_{e*}^{-1}\Big) \times \prod_{e*\in\partial f} \chi^{j_{f*}}(U_f)\,, \tag{B.10}$$
   $\{t_{f*}\}$ being a set of positive real numbers.
4. **Estimate**:
   Estimate the parameters $\{\bar{j}_l\}$, maximizing the probability derived from the amplitude $\mathcal{Z}_{\mathcal{C}}$, in a feedforward approach.
5. **Repeat**:
   Repeat the previous steps 1–4 for different choices of the boundaries $\partial\mathcal{C}$.

Appendix C. The semi-classical limit

We have so far considered spin-network basis states represented by cylindrical functionals of the holonomies, contracted with the intertwiner invariant tensors. A different representation involves coherent spin-network states (Bianchi et al., 2010a), which is obtained as the gauge-invariant projection of the product over links of heat kernels. Namely

$$\Psi_{\Gamma, H_{ab}}(h_{ab}) = \int \Big(\prod_a dg_a\Big) \prod_{ab} \mathcal{K}^{t_{ab}}(h_{ab}, g_a H_{ab} g_b^{-1}), \tag{C.1}$$

where a, b label the nodes of the maximal graph where the spin-networks live, pairs ab correspond to links, $g_a \in SU(2)$ are group elements at the nodes, $h_{ab} \in SU(2)$ label group elements over the links, and H_{ab} are group elements of $\mathrm{SL}(2, \mathbb{C})$, assigned to each link ab. Notice that elements of $\mathrm{SL}(2, \mathbb{C})$ can be expressed in terms of a positive real number η_{ab} and two independent SU(2) group-element g_{ab} and g_{ab}^{-1}, namely

$$H_{ab} = g_{ab} e^{\eta_{ab}(\sigma_3/2)} g_{ba}^{-1}\,. \tag{C.2}$$

The two SU(2) group elements cast uniquely in terms of an angle $\tilde{\phi}$ and a unit vector identified by its inclination and azimuth $\vec{n} = (\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)$. The associated SU(2) group element reads

$$n = \exp(-\imath\phi\sigma_3/2)\exp(-\imath\theta\sigma_2/2)\,, \tag{C.3}$$

and the SU(2) group elements g recast $g = n\exp(\imath\tilde{\phi}\sigma_3/2)$. Thus we get

$$H_{ab} = n_{ab} e^{-\imath z_{ab}(\sigma_3/2)} n_{ba}^{-1}\,. \tag{C.4}$$

[2] The coherent group elements of SU(2) are defined by $|\vec{n}, j\rangle := D^j(U_{\vec{n}})\, D^j(e)$, with e unit element of the group, $\vec{n}$ direction on S^3 that generically individuates $U \in SU(2)$ and $D^j(e) \equiv |j, \pm\hat{z}j\rangle$.

having introduced $z_{ab} = \xi_{ab} + \imath\eta_{ab}$, with $\xi_{ab} = \tilde{\phi}_{ba} - \tilde{\phi}_{ab}$. This finally allows to identify the set of parameters associated to each link, namely $(\vec{n}_{ab}, \vec{n}_{ba}, \xi_{ab}, \eta_{ab})$. These parameters give weight vectors that determine the transition amplitudes that the TQNN associates to input and output states. The learning process, therefore, consists of obtaining the weights that produce the maximal transition amplitudes with respect to a ground truth. For example, in the case of spin-networks associated to hand-written letters "L" given above, the weights have to maximize the transition amplitude corresponding to the lower bottom panel of Fig. 3.

The state in Eq. (C.1) can be expanded on the spin-network basis $\Psi_{\Gamma, j_{ab}, \iota_a}$,

$$\Psi_{\Gamma, H_{ab}}(h_{ab}) = \sum_{j_{ab}} \sum_{\iota_a} f_{j_{ab}, \iota_a} \Psi_{\Gamma, j_{ab}, \iota_a}(h_{ab}) , \tag{C.5}$$

with coefficients f_{j_{ab}, ι_a} individuated by

$$f_{j_{ab}, \iota_a} = \left(\prod_{ab} \Delta_{j_{ab}} e^{-j_{ab}(j_{ab}+1)t_{ab}} D^{j_{ab}}(H_{ab}) \right) \cdot \left(\prod_{a} v_{\iota_a} \right) . \tag{C.6}$$

In the large j_{ab} limit, the coherent states $\Psi_{\Gamma, H_{ab}}(h_{ab})$ undergo the expansion

$$\Psi_{\Gamma, H_{ab}}(h_{ab}) \simeq \sum_{j_{ab}} \left(\prod_{ab} \Delta_{j_{ab}} e^{-\frac{(j_{ab}-\bar{j}_{ab})^2}{2\sigma_{ab}^2}} e^{-\imath \xi_{ab} j_{ab}} \right) \times \Psi_{\Gamma, j_{ab}, \Phi_a(\vec{n}_{ab})}(h_{ab}) , \tag{C.7}$$

where the coherent intertwiners $\Phi_a(\vec{n}_{ab})$ can be decomposed on the intertwiner space v_{ι_a} by

$$\Phi_a(\vec{n}_{ab}) = \sum_{\iota_a} \Phi_{\iota_a}(\vec{n}_{ab}) v_{\iota_a} , \tag{C.8}$$

with

$$\Phi_{\iota_a}(\vec{n}_{ab}) = v_{\iota_a} \cdot \left(\bigotimes_b |j_{ab}, \vec{n}_{ab}\rangle \right) , \tag{C.9}$$

the variance of the Gaussian distribution per each link is inversely proportional to the diffusion time t_{ab}, namely $\sigma_{ab} \equiv 1/(2\, t_{ab})$, and finally the parameters $\bar{j}_{ab}$ over which the coherent state is peaked, which correspond to the estimated parameters we refer to through the paper, are related to the η_{ab}, the real numbers entering the parametrization of SL(2, $\mathbb{C}$) group elements, at each link by $\Delta_{\bar{j}_{ab}} \equiv \eta_{ab}/t_{ab}$.

The partition function of Section 3 is therefore changed in the semi-classical limit by the use of the approximations in Eq. (C.7) and the corresponding transition amplitudes between initial and final states $\Psi_{\Gamma, j_{\gamma}, \iota_n}$, $\Psi_{\Gamma, H_{ab}}$, respectively, are therefore computed according to the formula:

$$\begin{aligned} \mathcal{A}_{\prod_{ab} H_{ab}} &= \langle \Psi_{\Gamma, H_{ab}} | \Psi_{\Gamma, j_{\gamma}, \iota_n} \rangle \simeq \\ &\sum_{j_{ab}} \left(\prod_{ab} \Delta_{j_{ab}} e^{-\frac{(j_{ab}-\bar{j}_{ab})^2}{2\sigma_{ab}^2}} e^{-\imath \xi_{ab} j_{ab}} \right) \\ &\times \int dh_{ab} \overline{\Psi}_{\Gamma, j_{ab}, \Phi_a(\vec{n}_{ab})}(h_{ab}) \Psi_{\Gamma', j'_{ab}, v_{\iota' a}}(h_{ab}) \\ &= \sum_{j_{ab}} \left(\prod_{ab} \Delta_{j_{ab}} e^{-\frac{(j_{ab}-\bar{j}_{ab})^2}{2\sigma_{ab}^2}} e^{-\imath \xi_{ab} j_{ab}} \right) \\ &\times \delta_{\Phi_a(\vec{n}_{ab}), v_{\iota' a}} \delta_{j_{ab} j'_{ab}} \\ &= \left(\prod_{ab} \Delta_{j_{ab}} e^{-\frac{(j_{ab}-\bar{j}_{ab})^2}{2\sigma_{ab}^2}} e^{-\imath \xi_{ab} j_{ab}} \right) . \end{aligned} \tag{C.10}$$

Using the transition amplitudes above, between states in the semi-classical limit, we can apply the fundamental idea of the algorithm of Section 3 in the semi-classical limit to obtain:

1. **Initialize**:
   Associate spin-networks to images as in Section 4. This is done in two steps:
   1a associate to each training sample a 1-complex (i.e. a graph), where each node corresponds to the center of a pixel, and the edges connect pixels in the von Neumann neighborhoods;
   1b assign to each link of the 1-complex SU(2) irreducible representations, where the spin j representation label is determined by the pixel colors.
2. **Feedforward**:
   2a estimate the parameters entering the feedforward pattern through the functorial functional $\mathcal{Z}_{\mathcal{C}}(h_l)$, by maximizing the internal product $\mathcal{A}$ between this latter and the QNN boundary states supported on $\partial\mathcal{C}$. The geometric supports for QNN boundary states are graphs resulting from the disjoint union of any Γ', on which training samples are constructed, and 1-complexes supporting output states;
   2b for hidden layer approaches: compute the functorial composition (cobordism properties) to take place accordingly to Eq. (B.8), and consistently with the filtering process that is implemented by the selection of the sub-graph structure at each hidden layer.
3. **Classify**:
   3a introduce $H_l \in \mathrm{SL}(2, \mathbb{C})$, encoding the information on the set of parameters to be determined, namely $(\vec{n}_{ab}, \vec{n}_{ba}, \xi_{ab}, \eta_{ab})$;
   3b associate to each link of the 1-complex a set of parameters, the string $(\vec{n}_{ab}, \vec{n}_{ba}, \xi_{ab}, \eta_{ab})$, to be fitted in the learning process. This identifies the functional $\Psi_{\Gamma, H_{ab}}$;
   3c compute the internal product to associate probability amplitudes to the training samples:
   $$\mathcal{A}_{\prod_{ab} H_{ab}} = \langle \Psi_{\Gamma, H_{ab}} | \widetilde{\Psi}_{\Gamma, j_{\gamma}, \iota_n} \rangle , \tag{C.11}$$
   the $\Psi_{\Gamma, H_{ab}}$ denoting the functionals of the training samples, and $\widetilde{\Psi}_{\Gamma, j_{\gamma}, \iota_n}$ the functional associated to the image to be recognized.
4. **Estimate**:
   Estimate, for each training sample, the parameters $(\vec{n}_{ab}, \vec{n}_{ba}, \xi_{ab}, \eta_{ab})$, maximizing the probability derived from the amplitude $\mathcal{A}_{\prod_{ab} H_{ab}}$.
   These parameters individuate a rotation group element Eq. (C.3), which acting on a reference vector, e.g. the identity element of the SU(2) group, individuates the weight vector.
5. **Repeat**:
   Repeat the previous steps for different cylindrical functions, corresponding to different training samples, by using the estimated parameters, and the corresponding weight vectors.

Observe that the topological structure of the graph, and the related extended information that is encoded by its links and intertwiners, are captured by the combinatorial summation of the a, b indices, and by the information stored in the Kronecker delta on the projected coherent intertwiners at each node. On the other hand, metric properties are encoded in the Gaussian weights at each link, capturing the relevant quantitative information concerning the recognition of the specific digit. It is clear that the case

in which, at the link γ_{ab}, both the mean value $\bar{j}_{ab}$ and its dispersion $(j_{ab} - \bar{j}_{ab})^2/\sigma^2_{ab}$ are vanishing, no information relative to that link appears anymore in the amplitude, and the specific metric feature affects the topology of the graph, with the consequence that the graph will embed one link less. Finally, we recognize as a remarkable feature of this approach that probability interference terms (while computing $|\mathcal{A}|^2$) will be provided by the ξ_{ab} coefficients.

An implementation of TQNN without employing the semi-classical limit will appear elsewhere. Such an algorithm utilizes the machinery of Section 3 in its generality. We limit ourselves to mentioning that transition amplitudes, in the general setting, use the definition of Jones–Wenzl projector at the links of spin-networks, along with the projector of Noui and Perez (2005) to regularize the inner products.

Appendix D. A dictionary for Quantum Neural Networks

As we have already mentioned, the novelty of our model consists in using the richer structures of graph-supported spin-network states to represent training and test samples. As a matter of fact, as far as we know, it is the first time that graph structures are taken into account, together with their evolution supported on 2-complexes. Instead, within the traditional approach, nodes that are located at each boundary and hidden layer, are taken to evolve along with graphs (1-complexes).

Now we are ready to reformulate notions found in DNN theory in the language of TQNN. We restrict our illustration to the supervised learning scenario consisting, as it is well known, in learning a (typically unknown) function $g : X \to Y$ that maps a (typically large, e.g. all possible images of handwritten characters) input set X to a (typically much smaller, e.g. names of characters) output set Y, based on a training set $X' \subset X$ and hence an explicitly represented function $g' : X' \to Y$ specifying example input–output pairs. If $f : X \to Y$ is the (presumably random) function implemented by the network before training, we can represent the learning algorithm as an operation $\mathcal{L} : (f, g') \mapsto g$ on the initial function f given the training function g'. In particular, we follow the statistical learning framework of supervised learning delineated in Shalev-Shwartz and Ben-David (2014). Let us recall first, some classical definitions for DNN, see Shalev-Shwartz and Ben-David (2014).

- Sample complexity:
  It represents the number of training samples (i.e. $Card(X')$) that a learning algorithm needs in order to learn successfully a family of target functions.
- Model capacity:
  It is the ability of the model to fit a wide variety of functions; in particular, it specifies the class of functions $\mathfrak{H}$ (the hypothesis class) from which the learning algorithm $\mathcal{L}$ can choose the specific function $\mathfrak{h}$.
- Overfitting:
  A model is overfitting when the gap between training error and test error is too large; this phenomenon occurs when the model learns the training function g' but $\mathcal{L}$ incorrectly maps $(f, g') \mapsto h \neq g$, i.e. the trained network generalizes to the wrong function h and fails to predict future observations (i.e. additional samples from X) reliably. The training function g' has been merely "memorized" to the extent that h is random on X outside of the training sample X'.
- Underfitting:
  A model is underfitting when it is not able to achieve a sufficiently low error on the training function g'; this phenomenon occurs when the model does not adequately capture the underlying structure of the training data set and, therefore, may also fail to predict future observations reliably.
- Bias:
  It is the restriction of the learning system towards choosing a classifier or predictor $\mathfrak{h}$ from a specific class of functions $\mathfrak{H}$ (the hypothesis class).
- Empirical Risk Minimization (ERM):
  It consists in minimizing the error on the set of training data (the "empirical" risk), with the hope that the training data is enough representative of the real distribution (the "true" risk).
- Generalization:
  It is conceived as the ability of the learner to find a predictor, i.e. a map $X' \to X$, which is able to enlarge successfully its own predictions from the training samples to the test or unseen samples.

These notions can be translated into the TQNN dictionary as follows:

- Sample complexity:
  It is a measure of the Hilbert-space of the entire spin-network state that is supported on a specific graph Γ. It is then dependent on the connectivity of the graph (nodes and links of each graph, i.e. the multiplicity of connectivity that characterizes the graph Γ) and on the dimensionality of the Hilbert spaces connected to each link and node. In this sense complexity, once extended to the different classes of graphs corresponding to the training set, provides a measure of the entropy of the set. Therefore, in the TQNN framework, the notion of "complexity" has a wider meaning than its counterpart in DNN, for which the sample complexity is nothing but the size of the training set. This is summarized in the expression for the dimension of the Hilbert space $\mathcal{H}_\Gamma$ of the (whole) spin-network supported on Γ, namely
  $$\dim[\mathcal{H}_\Gamma] = \oplus_{j_l} \otimes_n \otimes_{l \in \partial n} \dim[\mathcal{H}_{j_l}]. \tag{D.1}$$
  This directly encodes both the size of the maximal graph where the input/output states live, as well as the algebro/analytical structure used in the TQFT from which the corresponding TQNN arises, as encoded by the dimensionality of the Hilbert spaces $\mathcal{H}_j$, for instance;
- Model capacity:
  It is quantified in terms of the interconnectivity of the graph Γ. It depends on the topological structure of the graphic support Γ of the spin-network states, and neither on the dimensionality of the Hilbert space of the irreducible representations nor on the intertwiner quantum numbers, respectively assigned to each link and node of Γ; in other words, it depends on the total valence V of Γ, defined in terms of the valences v_n of each node of Γ through the expression
  $$V = \sum_n v_n \,; \tag{D.2}$$
- Overfitting:
  As pointed out in Section 3, in the semi-classical limit, the integrals that allow us to compute the transition amplitudes that characterize a TQFT are interpreted as a "sum over all the geometries" of the ground topological manifold, where the integrand is some approximation of the Einstein–Hilbert action. During the learning process, then a TQNN learns how to select certain geometries with respect to certain others in order to maximize certain transition amplitudes corresponding to "a more suitable" classification. The information available to make this selection during the learning process is that given by the connectivity of the input graphs/spin-networks and their given correlation g' with the label set

Y. If g' is insufficiently representative of the target function g, the TQNN may only partially capture the topological structure of the full input set X and therefore be unlikely to classify correctly spin-network states that are not part of, or are significantly dissimilar from those contained in, the training set X';

- Underfitting:
  It represents the converse of the overfitting scenario. The geometries that have been selected in the learning process do not correspond to the graphs Γ at the starting point. Fewer information channels (links) are present and lower dimensionality of the information channels (dimensions of the Hilbert space associated with each holonomy) as well. As a consequence, the QNN cannot fit the training set and may therefore also fail to predict future observations reliably;
- Bias:
  It amounts to the predisposition of the spin-network to account for a specific set of data; it depends on the topological structure of the spin-network states, encoded in the connectivity properties of input Γ's and on the specific realization of the TQNN quantum state, i.e. on the weight of the quantum state on the spin-networks basis elements of the Hilbert space.
- Empirical Risk Minimization (ERM):
  It is the variance of the Gaussian distribution of the irreducible representations assigned to the holonomies on the links in the semi-classical limit, i.e.

$$\mathrm{ERM} := \sum_l \frac{(j_l - \bar{j}_l)^2}{2L}\,, \tag{D.3}$$

  with L equal to the total number of links.
- Generalization:
  It is the behavior of the system in response to test or unseen data analogous to a functor (amplitude) either from a boundary spin-network to another boundary spin-network, or from a boundary spin-network to a complex number. This is determined by the geometries that have been selected as the most representative of a certain training sample during the learning process. This is in practice captured by the parameters that give higher relevance, in the integral computing of the transition amplitudes in a TQNN, to certain boundary transitions, while suppressing others. These parameters are determined by (i) connectivity of 1- and 2-complexes (nodes and links, vertices and edges respectively), (ii) linking and knotting (e.g. for loops in a different Hilbert space representation), and (iii) states' sum (as a global topological charge, invariant under refinement of the triangulation, i.e. invariant under refinement of the data/group elements/intertwiners assigned to the links and the nodes). How the parameters determine the corresponding amplitudes is clear, for the TQNN used in practice in this article, from the formula for the partition function of the model:

$$\mathcal{Z}_{\mathcal{C}}(U_l) = \int_{\mathrm{SU}(2)^{2(E-L)-V}} dU_{v(e)} \int_{\mathrm{SU}(2)^{\mathcal{V}-L}} dU_f \times \prod_f \mathcal{K}_{f*}(U_{e*}, U_f)\,, \tag{D.4}$$

  where the "face amplitude" casts

$$\mathcal{K}_{f*}(U_{e*}, U_f) \equiv \sum_{j_{f*}} \Delta_{j_{f*}}\, \chi^{j_{f*}}\Big(\prod_{e*\in\partial f} U_{e*}\Big) \times \prod_{e*\in\partial f} \chi^{j_{f*}}(U_f)\,. \tag{D.5}$$

Finally, from the definitions of the present article, we can provide the meaning of Learner's input and output in the context of TQNN.

- Learner's input:
  (i) The domain set X: It corresponds to links l and nodes n, and attached holonomies U_l and invariant tensors ι_n respectively along with the links and at the nodes: it is concisely denoted as a state of the Hilbert space of the theory:

$$\Psi_{\Gamma;\{j_l\},\{\iota_n\}}[A] \equiv \Psi_\Gamma(U_l, \iota_n) := |\Gamma; \{j_l\}, \{\iota_n\}\rangle; \tag{D.6}$$

  (ii) The label set Y: It is a set of topological charges and quantum numbers, with which the 2-complex is endowed; for instance, recalling the group-isomorphism $\pi_3(S_3)$, for the mapping individuated by the homotopy group $\pi_3(S_3) = \mathbb{Z}$ the winding number w is defined as the integral over the SU(2) group element

$$w = \frac{1}{24\pi^2}\int_{\mathrm{SU}(2)} dU; \tag{D.7}$$

  (iii) The training data S: It is the union of the (initial) boundary-colored graphs together with the topological invariants associated with them through the QNN functorial action.
- Learner's output:
  It is a prediction rule, i.e. the QNN functor that identifies the topological charges of the boundary states (training/test samples) and thus implements the classifier; for Γ supporting a disjoint boundary state, the classifier is captured by the probability amplitude that results from the internal product

$$\mathcal{A} = \langle \Gamma; \{j_l\}, \{\iota_n\}| \, |\mathcal{Z}_{\mathcal{C},\partial\mathcal{C}=\Gamma}; \{j_l\}, \{\iota_n\}\rangle. \tag{D.8}$$

Lastly, contributions to the topological invariants can be recognized to be of several different types, including the ones associated with the connectivity of the graphs, the linking, and the knotting (e.g. in the loops decomposition of the TQNN boundary and intermediate spin-network states) and the states' sum invariants. The first two classes will be local in the experimental implementation of the TQNN, while the latter represents a global charge, the analytical expansion of which in the deformation parameter might entail an infinite number of momentum expansion of the charge.

Notice that generic boundary states are characterized by two classes of parameters, which we dub as topological and metric parameters: As reminded above, the former ones are captured either by the topology of the graph or by the topological invariant (linking and knotting) quantum numbers, which can be expressed in terms of quantum group representations and are characterized by the deformation parameter of the quantum group, while the latter ones are captured by the spin/label of the representation itself. Whenever not enough information about the topology is specified by the training data, any TQNN 2-complex with enough topological internal structure to account for the classification task will be selected. In other words, if the training data prescribe an effective shrinking of the "measure" of edges and links to zero, any topological feature of the graph, such as the valency of a node, or the knotting or linking of an edge, will cease to be. Metric parameters instead are individuated by the Gaussian weights associated with the coherent group elements assigned to the TQNN states and recovered by the fit on the spin representation set that is assigned to each training state. In this sense, since the parameters fit is achieved considering the whole amplitude $\mathcal{A}$, the resulting topology qualifies as a derivative-free feedforward architecture in which a composition of intermediate evolution operators among the hidden layers does not need to backpropagate the information.

References

Aïmeur, E., Brassard, G., & Gambs, S. (2013). Quantum speed-up for unsupervised learning. *Machine Learning*, *90*(2), 261–287. http://dx.doi.org/10.1007/s10994-012-5316-5.

Arpit, D., Jastrzkebski, S., Ballas, N., Krueger, D., Bengio, E., & Kanwal, M. S. (2017). A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning* (pp. 233–242). PMLR.

Baez, J. C. (2000). An introduction to spin foam models of bf theory and quantum gravity. In *Geometry and Quantum Physics* (pp. 25–93). Springer, http://dx.doi.org/10.1007/3-540-46552-9_2.

Beer, K., Bondarenko, D., Farrelly, T., Osborne, T. J., Salzmann, R., & Scheiermann, D. (2020). Training deep quantum neural networks. *Nature Communications*, *11*(1), 1–6. http://dx.doi.org/10.1038/s41467-020-14454-2.

Bianchi, E., Magliaro, E., & Perini, C. (2010). Coherent spin-networks. *Physical Review D*, *82*(2), 024012. http://dx.doi.org/10.1103/PhysRevD.82.024012.

Bianchi, E., Magliaro, E., & Perini, C. (2010). Spinfoams in the holomorphic representation. *Physical Review D*, *82*(12), 124031. http://dx.doi.org/10.1103/PhysRevD.82.124031.

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight uncertainty in neural network. In *International Conference on Machine Learning* (pp. 1613–1622). PMLR.

Carleo, G., & Troyer, M. (2017). Solving the quantum many-body problem with artificial neural networks. *Science*, *355*(6325), 602–606. http://dx.doi.org/10.1126/science.aag2302.

Cowan, N. (2001). The magical number 4 in short-term memory: a reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, *24*(1), 87–114. http://dx.doi.org/10.1017/S0140525X01003922.

Deutsch, D. (2002). The structure of the multiverse. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, *458*(2028), 2911–2923. http://dx.doi.org/10.1098/rspa.2002.1015.

Dinh, L., Pascanu, R., Bengio, S., & Bengio, Y. (2017). Sharp minima can generalize for deep nets. In *International Conference on Machine Learning* (pp. 1019–1028). PMLR.

Dziugaite, G. K., & Roy, D. M. (2017). *Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data*, arXiv:1703.11008.

Farhi, E., & Neven, H. (2018). *Classification with Quantum Neural Networks on Near Term Processors*, arXiv:1802.06002.

Feldman, J. (2000). Minimization of boolean complexity in human concept learning. *Nature*, *407*(6804), 630–633. http://dx.doi.org/10.1038/35036586.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a bayesian approximation: representing model uncertainty in deep learning. In *International Conference on Machine Learning* (pp. 1050–1059). PMLR.

Gawlikowski, J., Tassi, C. R. N., Ali, M., Lee, J., Humt, M., & Feng, J. (2022). *A Survey of Uncertainty in Deep Neural Networks*, arXiv:2107.03342.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. *Adaptive Computation and Machine Learning series*, MIT Press.

Griffiths, T. L. (2010). Bayesian models as tools for exploring inductive biases. In M. T. Banich, & D. Caccamise (Eds.), *Generalization of Knowledge: Multidisciplinary Perspectives* (1st Edition). (pp. 135–156). Psychology Press.

Griffiths, T. L., Christian, B. R., & Kalish, M. L. (2008). Using category structures to test iterated learning as a method for identifying inductive biases. *Cognitive Science*, *32*(1), 68–107. http://dx.doi.org/10.1080/03640210701801974.

Hoffer, E., Hubara, I., & Soudry, D. (2017). Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In *31st Conference on Neural Information Processing Systems* (pp. 1729–1739). Long Beach, CA, USA.

Kauffman, L., Lins, S., & Lins, S. (1994). Temperley-lieb recoupling theory and invariants of 3-manifolds. *Annals of Mathematics Studies*, Princeton University Press.

Kawaguchi, K., Kaelbling, L. P., & Bengio, Y. (2017). *Generalization in Deep Learning*, arXiv:1710.05468.

Kemp, C., & Jern, A. (2014). A taxonomy of inductive problems. *Psychonomic Bulletin & Review*, *21*(1), 23–46. http://dx.doi.org/10.3758/s13423-013-0467-3.

Keskar, N. S., Nocedal, J., Tang, P. T. P., Mudigere, D., & Smelyanskiy, M. (2017). On large-batch training for deep learning: generalization gap and sharp minima. In *5th International Conference on Learning Representations, ICLR*.

Krueger, D., Ballas, N., Jastrzebski, S., Arpit, D., Kanwal, M. S., & Maharaj, T. (2017). Deep nets don't learn via memorization. In *5th International Conference on Learning Representations (Workshop)* (pp. 1–4).

Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (pp. 6405–6416).

Lewis, R. L. (1996). Interference in short-term memory: the magical number two (or three) in sentence processing. *Journal of Psycholinguistic Research*, *25*(1), 93–115. http://dx.doi.org/10.1007/BF01708421.

Li, J., Sun, Y., Su, J., Suzuki, T., & Huang, F. (2020). Understanding generalization in deep learning via tensor methods. In *International Conference on Artificial Intelligence and Statistics* (pp. 504–515). PMLR.

Lin, H. W., Tegmark, M., & Rolnick, D. (2017). Why does deep and cheap learning work so well?. *Journal of Statistical Physics*, *168*(6), 1223–1247. http://dx.doi.org/10.1007/s10955-017-1836-5.

Lovett, N. B., Crosnier, C., Perarnau-Llobet, M., & Sanders, B. C. (2013). Differential evolution for many-particle adaptive quantum metrology. *Physical Review Letters*, *110*(22), 220501. http://dx.doi.org/10.1103/PhysRevLett.110.220501.

Malinin, A., & Gales, M. (2018). Predictive uncertainty estimation via prior networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* (pp. 7047–7058).

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information.. *Psychological Review*, *63*(2), 81–97. http://dx.doi.org/10.1037/h0043158.

Neyshabur, B., Bhojanapalli, S., McAllester, D., & Srebro, N. (2017). Exploring generalization in deep learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (pp. 5949–5958). Long Beach, CA, USA.

Neyshabur, B., Bhojanapalli, S., & Srebro, N. (2017). A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. In *International Conference on Learning Representations*.

Noui, K., & Perez, A. (2005). Three-dimensional loop quantum gravity: physical scalar product and spin-foam models. *Classical and Quantum Gravity*, *22*(9), 1739. http://dx.doi.org/10.1088/0264-9381/22/9/017.

O'Reilly, R. C., & McClelland, J. L. (1994). Hippocampal conjunctive encoding, storage, and recall: avoiding a trade-off. *Hippocampus*, *4*(6), 661–682. http://dx.doi.org/10.1002/hipo.450040605.

Paparo, G. D., Dunjko, V., Makmal, A., Martin-Delgado, M. A., & Briegel, H. J. (2014). Quantum speedup for active learning agents. *Physical Review X*, *4*(3), 031002. http://dx.doi.org/10.1103/PhysRevX.4.031002.

Ramalho, T., & Miranda, M. (2020). Density estimation in representation space to predict model uncertainty. In *International Workshop on Engineering Dependable and Secure Machine Learning Systems* (pp. 84–96). Springer.

Rosch, E. (1983). Prototype classification and logical classification: the two systems. In E. K. Scholnick (Ed.), *New Trends in Conceptual Representation: Challenges to Piaget's Theory?* (pp. 73–86). Erlbaum.

Rosch, E. (1988). Principles of categorization. In A. Collins, & E. E. Smith (Eds.), *Readings in Cognitive Science* (pp. 312–322). Morgan Kaufmann.

Rovelli, C. (2011). Simple model for quantum general relativity from loop quantum gravity. *Journal of Physics: Conference Series*, *314*, 012006. http://dx.doi.org/10.1088/1742-6596/314/1/012006.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, *20*(1), 61–80. http://dx.doi.org/10.1109/tnn.2008.2005605.

Schuld, M., Sinayskiy, I., & Petruccione, F. (2014). The quest for a quantum neural network. *Quantum Information Processing*, *13*(11), 2567–2586. http://dx.doi.org/10.1007/s11128-014-0809-8.

Shalev-Shwartz, S., & Ben-David, S. (2014). Understanding machine learning: from theory to algorithms. Cambridge University Press.

Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science*, *237*(4820), 1317–1323. http://dx.doi.org/10.1126/science.3629243.

Shwartz-Ziv, R., & Tishby, N. (2017). *Opening the Black Box of Deep Neural Networks via Information*, arXiv:1703.00810.

Sjöström, P. J., Turrigiano, G. G., & Nelson, S. B. (2001). Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. *Neuron*, *32*(6), 1149–1164. http://dx.doi.org/10.1016/s0896-6273(01)00542-6.

Sowa, J. F. (2006). Semantic networks. In L. Nadel (Ed.), *Encyclopedia of Cognitive Science*. Wiley.

Tiersch, M., Ganahl, E., & Briegel, H. J. (2015). Adaptive quantum computation in changing environments using projective simulation. *Scientific Reports*, *5*(1), 1–18. http://dx.doi.org/10.1038/srep12874.

Van Amersfoort, J., Smith, L., Teh, Y. W., & Gal, Y. (2020). Uncertainty estimation using a single deep deterministic neural network. In *International Conference on Machine Learning* (pp. 9690–9700). PMLR.

Vong, W. K., Hendrickson, A., Perfors, A., & Navarro, D. (2016). Do additional features help or harm during category learning? an exploration of the curse of dimensionality in human learners.. In A. Papafragou, D. Grodner, D. Mirman, & J. Trueswell (Eds.), *Proceedings of the 38th Annual Conference of the Cognitive Science Society* (pp. 2471–2476).

Wang, A., Zhou, H., Xu, W., & Chen, X. (2017). *Deep Neural Network Capacity*, arXiv:1708.05029.

Wattenmaker, W. D., Dewey, G. I., Murphy, T. D., & Medin, D. L. (1986). Linear separability and concept learning: context, relational properties, and concept naturalness. *Cognitive Psychology*, *18*(2), 158–194. http://dx.doi.org/10.1016/0010-0285(86)90011-3.

Wiebe, N., Kapoor, A., & Svore, K. M. (2016). Quantum perceptron models. In *Proceedings of the 30th International Conference on Neural Information Processing Systems* (pp. 4006–4014).

Witten, E. (1989). Quantum field theory and the jones polynomial. *Communications in Mathematical Physics*, *121*(3), 351–399. http://dx.doi.org/10.1007/BF01217730.

Wu, A., Nowozin, S., Meeds, E., Turner, R. E., Hernández-Lobato, J. M., & Gaunt, A. L. (2019). Deterministic variational inference for robust bayesian neural networks. In *7th International Conference on Learning Representations, ICLR*. New Orleans, LA, USA.

Wu, L., Zhu, Z., & E, W. (2017). *Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes*, arXiv:1706.10239.

Zhang, C., Bengio, S., Hardt, M., Recht, B., & Vinyals, O. (2017). Understanding deep learning requires rethinking generalization. In *5th International Conference on Learning Representations, ICLR*. Toulon, CS, France.

Zhang, C., Bengio, S., Hardt, M., Recht, B., & Vinyals, O. (2021). Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, *64*(3), 107–115. http://dx.doi.org/10.1145/3446776.

Zhao, X., Ou, Y., Kaplan, L., Chen, F., & Cho, J.-H. (2019). *Quantifying classification uncertainty using regularized evidential neural networks*, arXiv:1910.06864.

Zhu, J., Gibson, B., & Rogers, T. T. (2009). Human rademacher complexity. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems* (pp. 2322–2330).