Expressiveness and Learning of Hidden Quantum Markov Models

Sandesh Adhikary* University of Washington

Siddarth Srinivasan* MSR Montréal, Georgia Tech

Geoff Gordon MSR Montréal

Byron Boots University of Washington

Abstract

Extending classical probabilistic reasoning using the quantum mechanical view of probability has been of recent interest, particularly in the development of hidden quantum Markov models (HQMMs) to model stochastic processes. However, there has been little progress in characterizing the expressiveness of such models and learning them from data. We tackle these problems by showing that HQMMs are a special subclass of the general class of observable operator models (OOMs) that do not suffer from the *negative probability problem* by design. We also provide a feasible retraction-based learning algorithm for HQMMs using constrained gradient descent on the Stiefel manifold of model parameters. We demonstrate that this approach is faster and scales to larger models than previous learning algorithms.

1 Introduction and Related Work

Classical probabilistic graphical models provide a principled framework for Bayesian reasoning, and there has been much interest in extending this framework by incorporating the mathematical formalism of quantum mechanics (Leifer and Poulin, 2008; Yeang, 2010; Leifer and Spekkens, 2013; Warmuth and Kuzmin, 2014). Hidden quantum Markov models (HQMMs) (Monras et al., 2010; Clark et al., 2015; Srinivasan et al., 2018b) have been some of the more well-investigated models; recent work by Srinivasan et al. (2018b) showed that every finite-dimensional hidden Markov model (HMM) can also be modeled by a finite-dimensional HQMM, and empirically demonstrated some theoretical advantages of HQMMs over HMMs. A major motivation for investigating such 'quantum models' has been the promise of a more general and expressive class of probabilistic models. Yet, a clear characterization of the expressiveness of these models and a practical learning algorithm has remained lacking. These are precisely the problems we tackle in this paper.

Our theoretical exploration of HQMMs is primarily centered around their relationship to the observable operator models (OOMs) developed by Jaeger (2000). OOM-equivalents have been independently developed and are also referred to in the literature as uncontrolled predictive state representations (PSRs) (Singh et al., 2004), linearly dependent processes (Ito et al., 1992), and stochastic weighted automata (Balle et al., 2014; Thon and Jaeger, 2015). OOMs can be seen as a generalization of the well-known hidden Markov models (Rabiner, 1986), but despite their generality they lack a constructive definition. A valid OOM must never produce a negative probability for a sequence of observations, yet it is *undecidable* (Wiewiora, 2007) whether or not a candidate set of OOM parameters will yield negative probabilities. This is known as the *negative probability problem* (NPP) of OOMs, and must be handled with heuristics in practice (Cohen et al., 2013). An alternative approach is to construct models that avoid the NPP by design, such as norm-observable operator models (NOOMs) (Zhao and Jaeger, 2010) or quadratic weighted automata (Bailly, 2011). While NOOMs can simulate processes that no finite-dimensional HMM could model (such as the 'probability clock' (Zhao and Jaeger, 2010)), it is unclear whether they have the broad expressiveness of OOMs; it isn't even known if they contain HMMs as a subclass. In this context, we make three main theoretical contributions in this paper: (i) we show how HQMMs can be seen as a generalization of NOOMs, (ii) we formulate the Liouville representation of HQMMs which uniquely characterizes the model and allows for direct comparison between HQMMs, and (iii) we show that every finite-dimensional HQMM is equivalent to a finite-dimensional OOM, with the special property that we can characterize the valid initial states as the spectraplex of Hermitian PSD matrices with trace 1.

We also present results on learning these models from data. We use the Kraus operator parameterization of HQMMs using matrices $\{\mathbf{K}_i\}$ that satisfy the constraint

Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108.

* denotes equal contribution

$\sum_i \mathbf{K}_i^\dagger \mathbf{K}_i = \mathbb{I}$. Stacking the operators $\mathbf{K}_i$ vertically to form a matrix $\boldsymbol{\kappa}$, the constraint can be re-written as $\boldsymbol{\kappa}^\dagger \boldsymbol{\kappa} = \mathbb{I}$. The existing approach to learning HQMMs (Srinivasan et al., 2018b) yields feasible parameters by starting with an initial guess $\boldsymbol{\kappa}$ and iteratively finding unitary transformations that increase the likelihood of the data. However, this method is inefficient, often gets trapped in poor optima, and can only handle a small number of hidden states. The absence of a practical learning algorithm has been a bottleneck in the development of these models (Schuld et al., 2015). Our primary experimental contribution in this paper is the application and analysis of a viable approach to the learning problem: since $\boldsymbol{\kappa}$ lies on the Stiefel manifold (Stiefel, 1936; Edelman et al., 1998), we can directly learn feasible parameters by constraining gradient updates to lie on the manifold using a well-known retraction-based algorithm (Wen and Yin, 2013). We show that this approach is faster, finds better optima, and can handle more hidden states than the previous method.

2 The Expressiveness of HQMMs

In general, the models we discuss are used to model sequential data and assume an evolving latent state that emits discrete observations at each time-step. Further intuition on these models can be found in Appendix A.

2.1 Hidden Markov Models

Definition 1 (HMMs)**.** *An n-dimensional Hidden Markov Model with a set of discrete observations $\mathcal{O}$ is a tuple $(\mathbb{R}^n, \mathbf{A}, \mathbf{C}, \vec{x}_0)$ where initial state $\vec{x}_0$, transition matrix $\mathbf{A}$, and emission matrix $\mathbf{C}$ satisfy the following conditions:*

(i) Non-negative parameters: $\vec{x}_0 \in \mathbb{R}^n_{\geq 0}$, $\mathbf{A} \in \mathbb{R}^{n \times n}_{\geq 0}$, $\mathbf{C} \in \mathbb{R}^{|\mathcal{O}| \times n}_{\geq 0}$,

(ii) Normalized initial state: $\vec{1}^T \vec{x}_0 = 1$,

(iii) Column-stochastic operators: $\vec{1}^T \mathbf{A} = \vec{1}^T \mathbf{C} = \vec{1}^T$.

HMM belief states are always interpretable as probability distributions over hidden system states. At each time-step, we update the belief state and condition on observation using the column-stochastic matrices $\mathbf{A}$ and $\mathbf{C}$ respectively:

$$\vec{x}'_t = \mathbf{A}\vec{x}_{t-1} \qquad \vec{x}_t = \frac{\mathrm{diag}(\mathbf{C}_{(y,:)})\vec{x}'_t}{\vec{1}^T \mathrm{diag}(\mathbf{C}_{(y,:)})\vec{x}'_t}, \qquad (1)$$

where $\mathrm{diag}(\mathbf{C}_{y,:})$ places the row y of matrix $\mathbf{C}$ in a diagonal matrix. We can also compute the probability of a sequence of observations $\bar{y} = y_1, ..., y_t$ from a given belief state $\vec{x}$ as follows:

$$P(\bar{y}) = \vec{1}^T \mathrm{diag}(\mathbf{C}_{(y_t,:)})\mathbf{A} \cdots \mathrm{diag}(\mathbf{C}_{(y_1,:)})\mathbf{A}\vec{x} \qquad (2)$$

2.2 Observable Operator Models

We describe OOMs as a generalization of HMMs. Observe that the operations above can be equivalently represented by defining observable operators $\mathbf{T}_y = \mathrm{diag}(\mathbf{C}_{(y,:)})\mathbf{A}$ for each observation y:

$$\vec{x}_t = \frac{\mathbf{T}_y \vec{x}_{t-1}}{\vec{1}^T \mathbf{T}_y \vec{x}_{t-1}} \qquad P(\bar{y}) = \vec{1}^T \mathbf{T}_{y_t} \cdots \mathbf{T}_{y_1} \vec{x} \qquad (3)$$

We can arrive at OOMs by relaxing constraint (i) in Definition 1 (so entries in $\vec{x}$, $\mathbf{A}$, $\mathbf{C}$ can be negative) and requiring only that the model always assign non-negative probabilities to observations. This allows us to define a standard OOM as follows:

Definition 2 (Standard OOMs (Jaeger, 2000))**.** *An n-dimensional standard Observable Operator Model with a set of discrete observations $\mathcal{O}$ is a tuple $(\mathbb{R}^n, \{\mathbf{T}_y\}_{y \in \mathcal{O}}, \vec{x}_0)$ where initial state $\vec{x}_0 \in \mathbb{R}^n$ and observable operators $\{\mathbf{T}_y\}_{y \in \mathcal{O}} \in \mathbb{R}^{n \times n}$ satisfy the following constraints:*

(i) Normalized initial state: $\vec{1}^T \vec{x}_0 = 1$,

(ii) Normalized marginal over observations: $\vec{1}^T \sum_{y \in \mathcal{O}} \mathbf{T}_y = \vec{1}^T$,

(iii) Non-negative probabilities: $\vec{1}^T \mathbf{T}_{y_t} ... \mathbf{T}_{y_1} \vec{x}_0 \geq 0$ for all sequences $y_1 ... y_t$.

Note that the above definition is non-constructive since it does not tell us what constraints we could place on model parameters or initial states to satisfy condition (iii) – this is the cost of relaxing the non-negativity constraint.

In fact, it is *undecidable* whether a given candidate OOM $(\mathbb{R}^n, (\mathbf{T}_y)_{y \in \mathcal{O}}, \vec{x}_0)$ satisfying conditions (i)-(ii) will violate condition (iii) (Wiewiora, 2007). This is the root of the infamous negative probability problem (NPP) in OOMs, since we cannot identify whether a learned model will assign negative probabilities to observations.

Jaeger (2000) further showed that HMM $\subset$ OOM using the 'probability clock' OOM which requires an infinite-dimensional HMM to model. The non-negativity constraint (i) from Definition 1 forces the largest eigenvalue of an observable operator $\mathbf{T}_y$ of an HMM to be real (by the Perron-Frobenius theorem). However, negative entries in OOMs allow the largest eigenvalue to be complex, which allows the latent states (and hence conditional probabilities) to display oscillatory behaviour. Jaeger (2000) uses this property in their probability clock example.

A useful conceptual characterization of a candidate OOM with parameters $\{\mathbf{T}_y\}_{y \in \mathcal{O}}$ is the convex cone of valid initial states it admits, i.e., the initial states for which the model will never assign a negative probability for observations. If there is no such cone, the model is invalid. Indeed, Jaeger (2000) present the following alternative to condition (iii):

Proposition 1 (Jaeger (2000))**.** *A tuple $(\mathbb{R}^n, (\mathbf{T}_y)_{y \in \mathcal{O}}, \vec{x}_0)$ satisfying conditions (i)-(ii) of Definition 2 is an OOM if and only if there exists a pointed convex cone K such that:*

(i) Initial state is in the cone: $\vec{x}_0 \in K$,

(ii) Cone is closed under the operators: $\mathbf{T}_y\vec{x}\in K$ *for all* $\vec{x}\in K$ *and* $y\in\mathcal{O}$,

(iii) The sum of entries for any point in the cone is non-negative: $\vec{1}^T\vec{x}\geq 0$ *for all* $\vec{x}\in K$.

Conditions (i) and (ii) guarantee that any initial state inside such a cone will stay inside the cone under action of $\mathbf{T}_y$, and condition (iii) guarantees that any state inside the cone will evaluate to a non-negative probability. This characterization can also tell us which OOMs have equivalent HMMs: a finite-dimensional OOM has an equivalent finite-dimensional HMM if and only if K is a k-polyhedral cone for some k, i.e., it is generated by some finite set of vectors (Jaeger, 2000). Proposition 1 also gives us a recipe to find OOMs that do not suffer from the NPP: select a desired convex cone of valid initial states and construct operators such that the cone is closed under their action.

General OOMs The standard OOMs given in Definition 2 are the original formulation by Jaeger (2000), which is stricter than necessary. Various equivalent formulations have been proposed, including as Sequential Systems (SS) by Thon and Jaeger (2015), uncontrolled predictive state representations (PSRs), or stochastic weighted automata (Balle et al., 2014). In this paper, we refer to the these as 'general OOMs'. The main difference is that the model parameters are no longer constrained to be real, and we don't force the state entries to sum to one; instead the state can be any vector as long as we can use a linear functional $\vec{\sigma}$ (which for standard OOMs was fixed to be $\vec{1}^T$) to recover the probabilities. While the model parameters can be defined over arbitrary fields, we define general OOMs over the complex field as this allows us to eventually recover HQMMs.

Definition 3 (General OOMs (Thon and Jaeger, 2015))**.** *An n-dimensional general Observable Operator Model with a set of discrete observations $\mathcal{O}$ is a tuple $(\mathbb{C}^n,(\boldsymbol{\tau}_y)_{y\in\mathcal{O}},\vec{x}_0,\vec{\sigma})$ where initial state $\vec{x}_0\in\mathbb{C}^n$, observable operators $\{\boldsymbol{\tau}_y\}_{y\in\mathcal{O}}\in\mathbb{C}^{n\times n}$, and a linear evaluation functional $\vec{\sigma}\in\mathbb{C}^n$ satisfy the following constraints:*

(i) Normalized Initial State: $\vec{\sigma}^\dagger\vec{x}_0=1$,

(ii) Normalized marginal over observations: $\vec{\sigma}^\dagger\boldsymbol{\tau}_{y_t}...\boldsymbol{\tau}_{y_1}x_0 = \sum_{y\in\mathcal{O}}\vec{\sigma}^\dagger\boldsymbol{\tau}_y\boldsymbol{\tau}_{y_t}...\boldsymbol{\tau}_{y_1}\vec{x}_0$ *for all sequences* $y_1...y_t$,

(iii) Non-negative probabilities: $\vec{\sigma}^\dagger\boldsymbol{\tau}_{y_t}...\boldsymbol{\tau}_{y_1}\vec{x}_0\in[0,1]$ *for all sequences* $y_1...y_t$.

For such a model, the state update after observing $y\in\mathcal{O}$ and computing the probability of that observation are carried out as follows:

$$\vec{x}_t=\frac{\boldsymbol{\tau}_y\vec{x}_{t-1}}{\vec{\sigma}^\dagger\boldsymbol{\tau}_y\vec{x}_{t-1}} \qquad P(\bar{y})=\vec{\sigma}^\dagger\boldsymbol{\tau}_{y_t}\cdots\boldsymbol{\tau}_{y_1}\vec{x} \quad (4)$$

As shown in Proposition 13 of Thon and Jaeger (2015), every n-dimensional general OOM has an equivalent standard OOM that is a similarity transform away, i.e., we can find a similarity transform $\mathbf{S}$ such that $(\mathbb{C}^n,(\mathbf{S}\boldsymbol{\tau}_y\mathbf{S}^{-1})_{y\in\mathcal{O}},\mathbf{S}\vec{\omega}_0,\vec{\sigma}\,\mathbf{S}^{-1}) = (\mathbb{C}^n,(\mathbf{T}_y)_{y\in\mathcal{O}},\vec{v}_0,\vec{1}^T)$. We will use this equivalence to show that NOOMs and HQMMs are special cases of OOMs. Finally, we note that finite dimensional OOMs are the most expressive class of linear models capable of modeling any stochastic process whose 'system-dynamics' matrix (Singh et al., 2004) has finite rank (Zhao and Jaeger, 2010). Hence these models are extremely powerful, although the NPP makes it challenging to use them in practice.

† is the complex conjugate transpose

2.3 Norm-observable Operator Models

NOOMs represent a class of models designed to avoid the NPP by construction. The central idea is to wrap the output of the model with the non-linear function $\|\cdot\|^2$ so that it always returns non-negative values.

Definition 4 (NOOMs (Zhao and Jaeger, 2010))**.** *An n-dimensional Norm Observable Operator Model with a set of discrete observations $\mathcal{O}$ is a tuple $(\mathbb{R}^n,(\boldsymbol{\phi}_y)_{y\in\mathcal{O}},\vec{v}_0)$ where initial state $\vec{v}_0\in\mathbb{R}^n$ and observable operators $\{\boldsymbol{\phi}_y\}_{y\in\mathcal{O}}\in\mathbb{R}^{n\times n}$ satisfy the following constraints:*

(i) Normalized initial state: $\|\vec{v}_0\|_2^2=1$,

(ii) Normalized marginal over observations: $\sum_{y\in\mathcal{O}}\boldsymbol{\phi}_y^\dagger\,\boldsymbol{\phi}_y=\mathbb{I}$.

The updated state after observing $y\in\mathcal{O}$ and the probability of that observation can be computed as

$$\vec{v}_t=\frac{\boldsymbol{\phi}_y\vec{v}_{t-1}}{\|\boldsymbol{\phi}_{y_t}\ ...\ \boldsymbol{\phi}_{y_1}\vec{v}\|} \qquad P(\bar{y})=\|\boldsymbol{\phi}_{y_t}\ ...\ \boldsymbol{\phi}_{y_1}\vec{v}\|^2 \quad (5)$$

Although any stochastic process can be represented as a NOOM in some inner product space, this space may be infinite dimensional (Zhao and Jaeger, 2010). For practical purposes, we care about the expressiveness of finite-dimensional NOOMs. Zhao and Jaeger (2010) showed that NOOM $\subseteq$ OOM, and once again used the ability of a real-valued NOOM operator to have complex eigenvalues in a NOOM probability clock to show that there are finite-dimensional NOOMs that cannot be modeled exactly by finite-dimensional HMMs.

Zhao and Jaeger (2010) show that despite their non-linear form, NOOMs are equivalent to n^2-dimensional OOMs, and indeed we will build upon this approach to re-derive HQMMs. Zhao and Jaeger (2010) use Kronecker product relationships for the 2-norm (where $\vec{\mathbb{I}}$ is a vectorized identity matrix that implements a matrix trace operation) to show that sequence probabilities in a NOOM from Equation 5 can also be evaluated as:

$$P(\bar{y})=\vec{\mathbb{I}}_{n^2}^T\ (\boldsymbol{\phi}_{y_t}\otimes\boldsymbol{\phi}_{y_t})\ ...\ (\boldsymbol{\phi}_{y_1}\otimes\boldsymbol{\phi}_{y_1})(\vec{v}_0\otimes\vec{v}_0), \quad (6)$$

Now, if we define $\vec{\sigma}=\vec{\mathbb{I}}_{n^2}$, $\boldsymbol{\tau}_y=\boldsymbol{\phi}_y\otimes\boldsymbol{\phi}_y$, and the initial state $\vec{\omega}_0\in\mathbb{R}^{n^2}$ as $\vec{\omega}_0=\vec{v}_0\otimes\vec{v}_0$, we get a general OOM

$(\mathbb{C}^n,\ (\boldsymbol{\tau}_y)_{y\in\mathcal{O}},\ \vec{\omega}_0,\ \vec{\sigma})$. As shown by Zhao and Jaeger (2010), this is a similarity transform of a standard OOM, with $\mathbf{S}=\mathbb{I}_{n^2}+\frac{1}{n^2}\vec{\mathbf{1}}_{n^2}(\vec{\sigma}^T-\vec{\mathbf{1}}_{n^2}^T)$. Thus, NOOMs are not any more expressive than OOMs, i.e., NOOM $\subseteq$ OOM.

2.4 Hidden Quantum Markov Models

Previous work by Srinivasan et al. (2018b) derived HQMMs by generalizing HMMs using system-environment interactions (illustrated using a quantum circuit), and showed that every n-dimensional HMM can be modeled by an HQMM with no more than an n^2-dimensional hidden states. Here, we take a different approach; we will show how HQMMs can be defined through a series of natural generalizations of NOOMs in such a way that they also end up containing finite-dimensional HMMs. We do so by allowing parameters to be complex and expanding the concepts of NOOM states and operators using the representation in Equation 6.

Generalizing NOOM States We know from Equation 6 that the initial state $\vec{\omega}_0$ can viewed as a vectorized rank-1 Hermitian matrix $\boldsymbol{\rho}_0$, i.e., $\vec{\omega}_0=\text{vec}\left(\vec{v}_0\vec{v}_0^\dagger\right)=\text{vec}(\boldsymbol{\rho}_0)$. A natural generalization would be to let the initial state be a vectorized matrix of arbitrary rank, i.e., $\boldsymbol{\rho}_0=\sum_i p_i\vec{v}_i\vec{v}_i^\dagger$ instead. The normalization condition on the initial state can then be restated as $1=\vec{\sigma}^\dagger\vec{\rho}_0=\vec{\mathbb{I}}_{n^2}^T\vec{\rho}_0=\text{tr}(\boldsymbol{\rho}_0)=\sum_i p_i$.

As a linear combination of outer products of vectors with themselves, $\boldsymbol{\rho}_0$ must be Hermitian. We additionally assume that the constituent eigenvectors live in a Hilbert space $\mathcal{H}$, so that $\boldsymbol{\rho}_0$ lives in a Liouville space, i.e., the outer product of two Hilbert spaces. Further, in the NOOM, $\vec{v}_0\vec{v}_0^\dagger$ had a single eigenvalue of 1. If we impose no further constraints, we could allow p_i to be complex-valued or negative as long as the normalization condition above was satisfied. However, this could once again lead to negative probabilities when applying the evaluation $\vec{\sigma}$, and hence a non-constructive model. Thus, we impose a positive semi-definiteness (PSD) constraint on the initial state to guarantee that $p_i\in\mathbb{R}_{\geq 0}$ so that $\text{tr}(\boldsymbol{\rho}_0)$ is real and non-negative. Essentially, we are now considering a model whose initial states $\vec{\rho}_0$ are vectorized arbitrary-rank Hermitian PSD matrices, which constitute a pointed convex cone. Such matrices are called *density matrices* in quantum mechanics (Nielsen and Chuang, 2010), and the imposition of the PSD constraint on the states is what allows these models to avoid the NPP.

Generalizing NOOM operators Having defined a convex cone of valid states, we now derive operators that ensure that the state always evolves inside the cone. We refer to such operators acting on our states in Liouville space as *Liouville superoperators* $\{\mathbf{L}_y\}_{y\in\mathcal{O}}$. Condition (ii) in Definition 4 ensured that probabilities of observations computed by the NOOM were normalized, and the equivalent condition in the OOM representation in Equation 6 is that $\vec{\sigma}^\dagger\left(\sum_{y\in\mathcal{O}}\boldsymbol{\tau}_y\right)=\vec{\sigma}^\dagger$. We impose a similar constraint (trace preservation or TP) on the superoperators to ensure we get a normalized distribution over observations. In addition to this, we further need to ensure that the probabilities assigned to observations are real and non-negative, i.e., the operators must always preserve the Hermitian PSD condition of the state. Finding a constructive way to impose these restrictions on Liouville superoperators is challenging, and it is easier to do so on the 'reshuffled' version of it called its *Choi matrix* (Wood et al., 2015). The reshuffle operation involves reshaping the n^2-dimensional columns of the Liouville superoperator into $n\times n$ matrices. Going across the columns of $\mathbf{L}_y$ from left to right, we fill up the blocks of the Choi matrix column-first with these reshaped matrices (see Appendix B and Życzkowski and Bengtsson (2004) for further details). In the context of Hermitian preserving (HP) maps, there is no elegant way to also impose a simple PSD-preserving 'positivity' constraint (Choi, 1975; Pillis, 1967). Therefore, we must impose a slightly more restrictive complete positivity (CP) constraint which guarantees that the map $\mathbf{L}_y\otimes\mathbb{I}$ is PSD-preserving for identity matrices of any dimension. In fact, Choi (1975) suggest that a CP map is the natural constructive generalization of 'positivity' for a linear HP map. We define L-HQMMs as a generalization of NOOMs with these constraints:

Definition 5 (L-HQMMs)**.** *An n^2-dimensional Liouville-Hidden Quantum Markov Model with a set of discrete observations $\mathcal{O}$ is a tuple $(\mathbb{C}^{n^2},(\mathbf{L}_y)_{y\in\mathcal{O}},\vec{\rho}_0,\vec{\mathbb{I}})$ where the initial state $\vec{\rho}_0\in\mathbb{C}^{n^2}$ and Liouville superoperators $\{\mathbf{L}_y\}_{y\in\mathcal{O}}\in\mathbb{C}^{n^2\times n^2}$ with corresponding Choi matrices $\{\mathbf{C}_y\}_{y\in\mathcal{O}}$ satisfy the following constraints:*

(i) $\vec{\rho}_0$ *is a vectorized Hermitian PSD matrix of arbitrary rank,*

(ii) Normalized initial state: $\vec{\mathbb{I}}^T\vec{\rho}_0=1$,

(iii) CP: $\mathbf{C}_y\geq 0$ *(Choi matrix is PSD).*

(iv) TP: $\vec{\mathbb{I}}^T\left(\sum_{y\in\mathcal{O}}\mathbf{L}_y\right)=\vec{\mathbb{I}}^T$,

(v) HP: $\mathbf{C}_y=\mathbf{C}_y^\dagger$,

For such a model, the state update after observing $y\in\mathcal{O}$ and computing the probability of that observation are:

$$\vec{\rho}_t=\frac{\mathbf{L}_y\vec{\rho}_{t-1}}{\vec{\mathbb{I}}^T\mathbf{L}_y\vec{\rho}_{t-1}}\qquad\qquad P(\bar{y})=\vec{\mathbb{I}}^T\mathbf{L}_{y_t}\ ...\ \mathbf{L}_{y_1}\vec{\rho}\qquad(7)$$

The exact relationship between HQMMs and OOMs was previously unknown, but this formulation of HQMMs allows us to state an important result:

Theorem 1. *HQMM $\subseteq$ OOM, and the set of valid initial states for HQMMs is a spectraplex.*

Proof. Setting $\vec{\sigma}=\vec{\mathbb{I}}$, L-HQMMs satisfy condition (i) of General OOMs laid out in Definition 3 by construction. Condition (ii) of Definition 3 is satisfied by the TP constraint on L-HQMMs. Next, the HP and CP constraints on L-HQMMs guarantee that $\mathbf{L}_{\bar{y}}\vec{\rho}$ always yields

a vectorized Hermitian PSD matrix. The trace of this matrix is always real and non-negative, i.e., $\vec{\mathbb{I}}^T\mathbf{L}_{\bar{y}}\vec{\rho} \geq 0$. We also have $\vec{\mathbb{I}}^T\mathbf{L}_y\vec{\rho}_0 \leq \vec{\mathbb{I}}^T\left(\sum_{y\in\mathcal{O}}\mathbf{L}_y\right)\vec{\rho}_0 = \vec{\mathbb{I}}^T\vec{\rho}_0 = 1$, satisfying condition (iii) of Definition 3.

The valid initial states of L-HQMMs are Hermitian PSD matrices with unit trace. Hermitian PSD matrices form a convex cone, and the intersection of this cone with the linear affine subspace of trace 1 matrices is a spectrahedron known as a spectraplex. □

Using the same similarity transform that we used for NOOMs $\mathbf{S} = \mathbb{I}_{n^2} + \frac{1}{n^2}\vec{\mathbf{1}}_{n^2}(\vec{\sigma}^T - \vec{\mathbf{1}}_{n^2}^T)$, we can transform any n^2-dimensional L-HQMM into an equivalent standard OOM. It is still an open question whether HQMMs are a proper subset of OOMs.

An alternate formulation of HQMMs Prior work on HQMMs have represented these models in the so-called operator-sum representation (Srinivasan et al., 2018b; Monras et al., 2010). While operations on vectorized matrices are fairly common in quantum information (and was implicitly used for HQMMs in Srinivasan et al. (2018a)), L-HQMMs are a novel formulation of HQMMs. We now derive the operator-sum representation of HQMMs from L-HQMMs, showing that the two are equivalent.

From Definition 5, we know that any model equivalent to L-HQMMs must have CP, TP, and HP operators. From Choi's theorem (Choi, 1975), we know that any map which can be expressed in the operator-sum representation $\mathcal{K}(\boldsymbol{\rho}) = \sum_w \mathbf{K}_w\ \boldsymbol{\rho}\ \mathbf{K}_w^\dagger$ is guaranteed to be CP, and will preserve the PSD nature of any input matrix. In the context of CP maps, the operator matrices $\mathbf{K}_w$ are commonly called Kraus operators (Kraus, 1971). The quadratic application of operators preserves the Hermiticity of $\boldsymbol{\rho}$. Thus, the operator-sum representation is particularly appealing because it guarantees the CP and HP constraints by construction. Note that this representation of CP maps is merely the inverse vectorization of the Liouville form

$$\text{vec}\left(\sum_w \mathbf{K}_w\ \boldsymbol{\rho}\ \mathbf{K}_w^\dagger\right) = \sum_w (\mathbf{K}_w^* \otimes \mathbf{K}_w)\vec{\rho} = \mathbf{L}\vec{\rho}$$

Thus, the action of a Liouville superoperator $\mathbf{L}_y$ corresponding to the observable y on $\vec{\rho}$ can can be equivalently represented by a set of Kraus operators $\{\mathbf{K}_{y,w_y}\}$ acting on the density matrix $\boldsymbol{\rho}$, where the cardinality of this set $|w_y|$ is determined by the Schmidt-rank (or Kraus-rank, as we soon explain) of $\mathbf{L}_y$. The Schmidt-rank is analogous to the rank revealed by an SVD, but for a decomposition into a Kronecker product of two vector spaces.

Finally, the operator-sum representation also provides a convenient way of ensuring the TP constraint: the full set of Kraus operators across all observables must satisfy $\sum_{y,w_y}\mathbf{K}_{y,w_y}^\dagger\mathbf{K}_{y,w_y} = \mathbb{I}$ (Nielsen and Chuang, 2010). This condition essentially generalizes condition (ii) for NOOMs in Definition 4 to allow multiple operators per observable. We can now define HQMMs using the Kraus operator-sum representation, as given in Srinivasan et al. (2018b).

Definition 6 (K-HQMMs)**.** *An n-dimensional Kraus-Hidden Quantum Markov Model with a set of discrete observations $\mathcal{O}$ is a tuple $(\mathbb{C}^{n\times n}, \{\mathbf{K}_{y,w_y}\}_{y\in\mathcal{O}}, \boldsymbol{\rho}_0, tr(\cdot))$ where initial state $\boldsymbol{\rho}_0 \in \mathbb{C}^{n\times n}$ and Kraus operators $\{\mathbf{K}_{y,w_y}\}_{y\in\mathcal{O},w_y\in\mathbb{N}} \in \mathbb{C}^{n\times n}$ satisfy the following constraints:*

(i) $\boldsymbol{\rho}_0$ is a Hermitian PSD matrix of arbitrary rank,

(ii) Normalized Initial State: $tr(\boldsymbol{\rho}_0) = 1$,

(iii) Normalized marginal over observations (TP): $\sum_{y,w}\mathbf{K}_{y,w}^\dagger\mathbf{K}_{y,w} = \mathbb{I}$.

The state update after observing y is computed as

$$\boldsymbol{\rho}_t = \frac{\sum_{w_y}\mathbf{K}_{y,w_y}\ \boldsymbol{\rho}_{t-1}\ \mathbf{K}_{y,w_y}^\dagger}{\text{tr}\left(\sum_{w_y}\mathbf{K}_{y,w_y}\ \boldsymbol{\rho}_{t-1}\ \mathbf{K}_{y,w_y}^\dagger\right)}, \tag{8}$$

and probability of a given sequence is given by:

$$P(\bar{y}) = \text{tr}\left(\sum_{w_{y_t}}\mathbf{K}_{y_t,w_{y_t}}\ldots\left(\sum_{w_{y_t}}\mathbf{K}_{y_1,w_{y_1}}\boldsymbol{\rho}_0\mathbf{K}_{y_1,w_{y_1}}^\dagger\right)\ldots\mathbf{K}_{y_t,w_{y_t}}^\dagger\right) \tag{9}$$

The K-HQMM representation was used by Srinivasan et al. (2018b) to show that any n dimensional HMM can be written as an equivalent n^2 dimensional K-HQMM, while there were HQMMs like the NOOM probability clock (trivially an HQMM) that required infinite-dimensional HMMs; hence HMM $\subset$ HQMM.

Uniqueness of L-HQMMs Note that the Kraus operator sum formulation of K-HQMMs does not uniquely define a CP map; it can be equivalently defined using different sets of Kraus operators (with possibly different cardinalities). Thus, it is not evident how one might compare two K-HQMMs. On the other hand, the Liouville superoperator is the unique representation of a CP map, and can be factorized into a canonical set of Kraus operators (Wood et al., 2015; Miszczak, 2011). The minimal number of Kraus operators in such a factorization is known as the Kraus-rank of the CP map. We describe this unique factorization further in Appendix B.

HQMMs & NOOMs We have shown that n-dimensional NOOMs form a subset of n-dimensional HQMMs through generalization. Prior work by Srinivasan et al. (2018b) used 'HQMMs' and 'NOOMs' somewhat ambiguously, differentiating them primarily by the field over which they are defined ($\mathbb{R}$ or $\mathbb{C}$). In this paper we have used the original formulation of NOOMs (Zhao and Jaeger, 2010) to draw a clearer distinction, whereby NOOMs are simply HQMMs with rank-1 vectorized initial states and Kraus-rank 1 operators. Particularly, for a fixed latent dimension n^2 of the vectorized density matrix, an HQMM allows for a greater diversity of both states and dynamics.

First, note that the valid states of HQMMs are Hermitian PSD matrices with unit trace, also known as mixed density matrices in quantum mechanics (Nielsen and Chuang, 2010). By contrast, the valid states for NOOMs correspond to the set of pure density operators (with rank 1). Since these operators encode the probability distribution of the latent state, we see that HQMM states can represent mixture distributions of NOOM states. Formally, the set of rank-1 density matrices are extremal points of the spectraplex defined by arbitrary rank density matrices. This gives us some geometric intuition for why HQMMs have a richer state space than NOOMs.

Second, HQMMs can have an arbitrary number of Kraus operators per observable while NOOMs are restricted to one to preserve rank-1 states. This indicates that the evolution associated with individual observations in an n-dimensional NOOM is restricted to dynamics corresponding to rank-1 Choi matrices. Thus, an n-dimensional HQMMs with arbitrary Kraus rank can encode richer dynamics than an n-dimensional NOOM.

3 Learning HQMMs

Having characterized the expressiveness of HQMMs, we now turn to the task of learning them from data.

The Learning Problem We use the negative log-likelihood of the data as our loss function, which can be written as a function of the set of Kraus operators $\{\mathbf{K}_{y,w}\}$ as follows (Srinivasan et al., 2018b):

$$\mathcal{L} = -\ln \operatorname{tr}\left(\sum_{w}\mathbf{K}_{y_n,w}...\left(\sum_{w}\mathbf{K}_{y_1,w}\ \boldsymbol{\rho}_0\ \mathbf{K}_{y_1,w}^{\dagger}\right)...\mathbf{K}_{y_n,w}^{\dagger}\right) \tag{10}$$

Note that the learned Kraus operators must satisfy the TP constraint $\sum_{y,w}\mathbf{K}_{y,w}^{\dagger}\mathbf{K}_{y,w} = \mathbb{I}$. The problem of learning a set of N trace-preserving $n \times n$ Kraus operators can equivalently be framed as one of learning a matrix $\boldsymbol{\kappa} \in \mathbb{C}^{nN \times n}$ on the Stiefel manifold i.e., that satisfy $\boldsymbol{\kappa}^{\dagger}\boldsymbol{\kappa} = \mathbb{I}$, where $\boldsymbol{\kappa}$ can be block-partitioned row-wise into the N Kraus operators that parameterize the HQMM. Both the previous and this paper's approach begin with an initial guess $\boldsymbol{\kappa}_0$ with a pre-determined partitioning into the Kraus operators we wish to learn, and iteratively make changes to the guess to maximize the log-likelihood.

The Previous Approach Since $\boldsymbol{\kappa}$ is a matrix with orthonormal columns, any initial guess $\boldsymbol{\kappa}_0$ is a unitary transformation away from the true $\boldsymbol{\kappa}^*$ that maximizes the log-likelihood. The existing method (Srinivasan et al., 2018b) iteratively finds a series of Givens rotations that locally increase the log-likelihood. However, a Givens rotation only changes two rows of $\boldsymbol{\kappa}$ at a time, making this approach prohibitively slow for learning large $\boldsymbol{\kappa}$ matrices. Furthermore, since these two rows are picked at random, this approach is not guaranteed to step towards the optimum at every iteration.

Retraction Based Optimization We propose directly learning $\boldsymbol{\kappa}$ using a gradient-based algorithm. Note that since $\mathcal{L}$ is a function of complex matrices, the direction of steepest descent corresponds to the gradient with respect to the complex conjugate of the Kraus operators (Hjørungnes and Gesbert, 2007). We use the algorithm proposed by Wen and Yin (2013) to constrain our parameter updates on the Stiefel manifold. Given a gradient $\mathbf{G}$ of $\mathcal{L}$ with respect to parmeters $\boldsymbol{\kappa}$, the constrained update $\gamma(\tau)$ from some initial feasible solution $\boldsymbol{\kappa}_0$ is

$$\gamma(\tau) = \boldsymbol{\kappa}_0 - \tau\mathbf{U}\left(\mathbb{I} + \frac{\tau}{2}\mathbf{V}^{\dagger}\mathbf{U}\right)^{-1}\mathbf{V}^{\dagger}\boldsymbol{\kappa}_0, \tag{11}$$

where $\mathbf{U} = [\mathbf{G} \mid \boldsymbol{\kappa}_0]$, $\mathbf{V} = [\boldsymbol{\kappa}_0 \mid -\mathbf{G}]$, $\mathbf{G}$ is the gradient at $\boldsymbol{\kappa}_0$, and τ the step size. The trajectory $\gamma(\tau)$ is a smooth *retraction* (Adler et al., 2002) of the gradient onto the manifold, and is the direction of steepest descent to feasibly optimize Equation 10. We combine this retraction with a simple gradient descent scheme to ensure that $\boldsymbol{\kappa}^{\dagger}\boldsymbol{\kappa} = \mathbb{I}$ after every update. In Appendix C, we provide an algorithm box and further details on the Wen-Yin algorithm, and also compare it against some alternative approaches.

4 Experimental Results

To show the superior performance of the retraction-based algorithm for constrained optimization on the Stiefel manifold (**COSM**) over the previous Givens Search (**GS**) method in learning HQMMs, we evaluate their accuracy and run-time on two datasets. The first is the synthetic dataset used by Srinivasan et al. (2018b) that was generated by an HMM. The second is a real-world dataset, on which the GS approach is prohibitively slow; demonstrating the scalability of COSM. In Appendix D, we also present results where COSM outperforms GS on the synthetic data used by Srinivasan et al. (2018b) that was generated by an HQMM representing a quantum mechanical process.

Training For all our HQMMs, we use the log-likelihood loss function from Equation 10. We initialize the latent state $\boldsymbol{\rho}_0$ as a random Hermitian PSD matrix using the QETLAB toolbox (Johnston, 2016), and $\boldsymbol{\kappa}$ as a random orthonormal matrix. Except for very small models, COSM is fairly robust to random initializations (see Appendix E). We compute the gradient of the loss function with respect to the complex conjugate of the Kraus operators using the Autograd package (which can handle complex differentiation), and vertically stack the gradients of the Kraus operators to construct the gradient $\mathbf{G}$ of the matrix $\boldsymbol{\kappa}$. To smoothen the trajectory we apply momentum with $\beta = 0.9$ (Rumelhart et al., 1986; Qian, 1999), and re-normalize the gradient before and after the momentum update, making

A preliminary version of these experimental results appeared in Adhikary et al. (2019). Code available at `https://github.com/sandeshAdhikary/learning-hqmms-stiefel-manifold`

the magnitude of updates entirely dependent on step-size. We refer to HQMMs using the tuple $(n,s,w)-$HQMM, where n is the number of hidden states, s is the number of possible outputs (earlier denoted $|\mathcal{O}|$), and w is the number of Kraus operators per output, also referred to as the dimension of the 'environment' variable (Srinivasan et al., 2018b). Consequently, for an $(n,s,w)-$HQMM we have $\boldsymbol{\kappa} \in \mathbb{C}^{nsw \times n}$. Note that when $w=1$, we are effectively learning a NOOM. We also provide the performance of HMMs trained using the Expectation-Maximization (**EM**) algorithm (with 5 random restarts) for reference. Details of our hyperparameter tuning procedure and computing infrastructure are described in Appendix F.

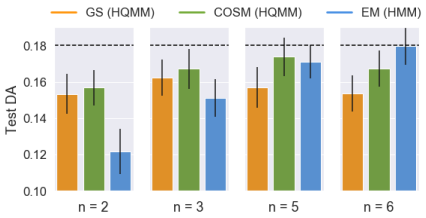
Metrics On the synthetic HMM dataset, we use a scaled log-likelihood (M. Zhao, 2007; Srinivasan et al., 2018b) independent of sequence length called description accuracy: $DA = f\left(1 + \frac{\log_s P(Y|\mathbb{D})}{\ell}\right)$, where $f(\cdot)$ squishes the log-likelihood from $(-\infty,1]$ to $(-1,1)$ (with $f(x)=\tanh(x/8)$ for $x \leq 0$, and $f(x)=x$ for $x>0$). When $DA=1$, the model predicted the sequence with perfect accuracy, and when $DA>0$, the model performed better than random. The error bars represent one standard deviation of the DA scores across many test samples. On the real-world dataset, we report the average accuracy for a classification problem.

4.1 Synthetic HMM Data

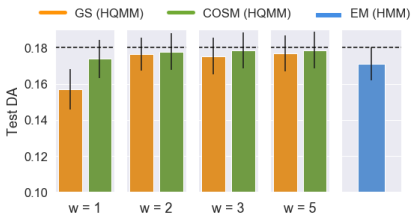
For our first experiment, we generated data using the same synthetic HMM as Srinivasan et al. (2018b), with 6 hidden states and 6 possible outputs. We show two things with the experiments on this dataset: 1) COSM finds better optima than GS, and 2) COSM is much faster than GS – so much so that we could train larger HQMMs than were previously possible. We also investigate the effects of increasing model size by adding latent states (n) versus increasing the Kraus-rank (w).

We used the same 20 training and 10 validation sequences of length 3000 used by Srinivasan et al. (2018b), splitting up each sequence into 300 sequences and use a burn-in of 100. We trained HQMMs using the COSM approach for 60 epochs, and evaluated the model with the highest validation DA score on the test set. The results for this model are shown in Figure 1a and Figure 1b.

COSM finds better optima than GS As shown in Figure 1a, HQMMs (with $w=1$) learned using COSM achieve better optima than HQMMs learned using GS for all n. As described in Section 2.4, these models are essentially complex-valued NOOMs. We also confirm that as noted in Srinivasan et al. (2018b), small HQMMs ($n \leq 5$) can model this data better than small HMMs, although this doesn't hold for $n=6$. However, we can take advantage of the additional Kraus-rank hyperparameter w available to HQMMs to further improve performance, as shown in Figure 1b for $(5,6,w)-$HQMMs (varying w). Also note that the number of parameters for an HQMM scales faster than for an HMM.



(a) $w=1$



(b) $n=5$

Figure 1: **Test Set Performances on the Synthetic HMM Data**: The dashed line represents the test set performance of the true model (a (6,6)-HMM) that generated the data.

COSM is much faster than GS In Figure 2, we plot the test set DA versus CPU training time for the smallest and largest models trained. To ensure a fair comparison, we train both approaches on sequences of length 300 and a batch size of 30. Note that we pre-tune hyperparameters on the validation set, and the graphs show the changing test DA as the models are trained with these hyperparameters (test DAs were not used to tune hyperparameters). For all models, we see that COSM converges much faster than GS, and the difference in both speed and accuracy is especially pronounced for the larger models; COSM quickly approaches convergence within a few hundred seconds, while GS yields very poor solutions even after 2000 seconds. As the GS method can take days to converge for large models, we did not directly calculate a precise speedup but provide an estimate in Appendix G.

Srinivasan et al. (2018b) proved that a (6,6,6)-HQMM should be sufficient to fully model a (6,6)-HMM, but the GS method was too slow to train this model. With COSM, we are able to show that this theoretical guarantee holds in practice. In fact, we find that in practice a $(5,6,3)-$HQMM is sufficient to model our $(6,6)-$HMM.

4.2 Splice Dataset

For our second experiment, we use the real-world splice dataset (Dheeru and Karra Taniskidou, 2017; Towell et al., 1991) consisting of DNA sequences of length 60, each element of which represents one of four nucleobases: Adenine (A), Cytosine (C), Guanine (G), and Thyamine (T). A DNA sequence typically consists of information encoded in sub-sequences (exons), that are separated by superfluous sub-sequences (introns). The task associated with this dataset is to classify sequences as having an

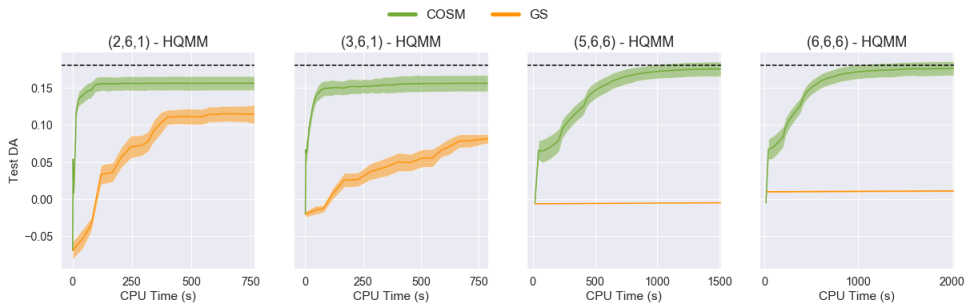


Figure 2: **COSM Learns More Accurate Models Faster than GS**: Test DA versus training time for various (n,s,w)-HQMMs trained on the synthetic HMM data. COSM converges to a better optimum faster than GS for all models; the dashed line represents the DA of the true data generating model.

exon-intron (EI) splice, an intron-exon (IE) splice, or neither (N), with 762, 765, and 1648 labeled examples for each label respectively. In addition to A, C, T and G, the raw dataset also contains some ambiguous characters, which we filter out prior to training. We demonstrate that COSM can be used to train HQMMs on real-world datasets which would have been too slow to train using GS.

We train a separate model for each of the three labels, and during test-time, choose the label corresponding to the model that assigned the highest likelihood to the given sequence. We train HQMMs using the COSM method and HMMs with the EM algorithm (with 5 random restarts) for reference. In Figure 3, we report the average classification accuracies across all labels obtained with 5-fold cross validation. For reference, a random classifier achieves around 33.3% accuracy.

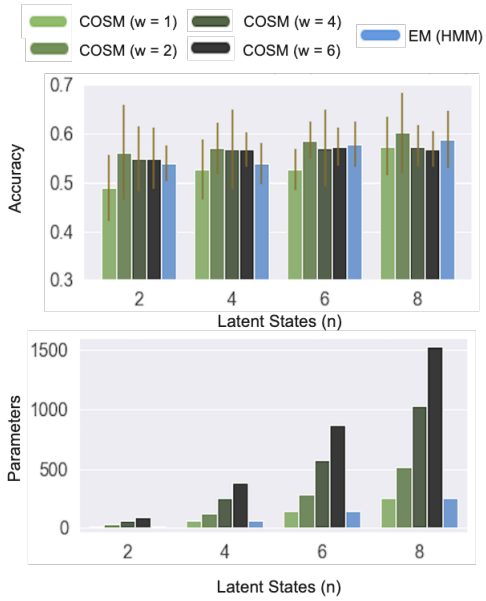


Figure 3: **Average 5-fold Test Set Performance on the Splice Dataset** Test set accuracies (top) and number of parameters (bottom) for various HQMMs and HMMs trained using the COSM and EM algorithms respectively. Errorbars in the top graph represent the mean standard deviation across labels over the 5 folds.

Note that 5-fold cross-validation is prohibitively time consuming for GS, even for models with a modest number of parameters. However, we are able to learn these HQMMs with COSM. We also see that (as before) there is a sizable marginal gain in DA when going from $w=1$ to $w=2$, with the benefits of increasing w further being less clear. However unlike the previous experiment, we still see persistent gains by increasing n. Interpreting this in conjunction with the results in the previous section suggests that we have to tune both n and w depending on the dataset. We also find that for a given number of hidden states, COSM is able to learn an HQMM that outperforms the corresponding HMM, although this comes at the cost of a rapid scaling in the number of parameters.

5 Conclusion

We showed that HQMMs are OOMs that generalize NOOMs, and that unlike prior approaches that avoid the NPP by design, HQMMs are able to model arbitrary HMMs as well. HQMMs expand the convex cone of valid states from rank-1 PSD matrices in (complex valued) NOOMs to arbitrary rank Hermitian PSD matrices. We also formulated the unique Liouville representation of a HQMMs, which allows direct comparison between models, and also simplifies theoretical analysis connecting them to general OOMs. Future work could focus on identifying the exact relationship between NOOMs and HMMs, and whether arbitrary OOMs can be converted to HQMMs.

We also used a retraction-based learning algorithm that directly constrains gradient updates to the Stiefel manifold to learn feasible HQMMs, and presented experimental results on a synthetic and a real-world dataset. Our proposed algorithm outperforms the prior approach in terms of both accuracy and speed, allowing us to train HQMMs that were previously too large to train. Future work could investigate approximation strategies that reduce the parameters of HQMMs at minimal cost to performance, or learning schemes that dynamically learn the Kraus-rank w instead of tuning it as a hyperparameter. Other future work could develop new QGMs defined via Kraus operators, which can be learned using our approach.

References

Absil, P.-A., Mahony, R., and Sepulchre, R. (2007). *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, USA.

Adhikary, S., Srinivasan, S., and Boots, B. (2019). Learning quantum graphical models using constrained gradient descent on the stiefel manifold. *arXiv preprint arXiv:1903.03730*.

Adler, R. L., Dedieu, J. P., Margulies, J. Y., Martens, M., and Shub, M. (2002). Newton's method on Riemannian manifolds and a geometric model for the human spine. *IMA Journal of Numerical Analysis*.

Bailly, R. (2011). Quadratic weighted automata: Spectral algorithm and likelihood maximization. In *Asian Conference on Machine Learning*, pages 147–163.

Balle, B., Hamilton, W., and Pineau, J. (2014). Methods of moments for learning stochastic languages: Unified presentation and empirical comparison. In *International Conference on Machine Learning*, pages 1386–1394.

Choi, M.-D. (1975). Completely positive linear maps on complex matrices. *Linear Algebra and its Applications*, 10(3):285 – 290.

Clark, L. A., Huang, W., Barlow, T. M., and Beige, A. (2015). Hidden quantum markov models and open quantum systems with instantaneous feedback. In *ISCS 2014: Interdisciplinary Symposium on Complex Systems*, pages 143–151. Springer.

Cohen, S. B., Stratos, K., Collins, M., Foster, D. P., and Ungar, L. (2013). Experiments with spectral learning of latent-variable pcfgs. In *Proceedings of the 2013 conference of the North American chapter of the Association for Computational Linguistics: human language technologies*, pages 148–157.

Dheeru, D. and Karra Taniskidou, E. (2017). UCI machine learning repository. Available at: http://archive.ics.uci.edu/ml.

Edelman, A., Arias, T., and Smith, S. (1998). The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353.

Gerlach, W. and Stern, O. (1922). Der experimentelle nachweis der richtungsquantelung im magnetfeld. *Zeitschrift für Physik*, 9(1):349–352.

Hjørungnes, A. and Gesbert, D. (2007). Complex-valued matrix differentiation: Techniques and key results. *IEEE Transactions on Signal Processing*, 55:2740–2746.

Ito, H., Amari, S.-I., and Kobayashi, K. (1992). Identifiability of hidden markov information sources and their minimum degrees of freedom. *IEEE transactions on information theory*, 38(2):324–333.

Jaeger, H. (2000). Observable operator models for discrete stochastic time series. *Neural computation*, 12(6):1371–1398.

Jiang, B. and Dai, Y.-H. (2013). A framework of constraint preserving update schemes for optimization on stiefel manifold. *Math. Program.*, 153:535–575.

Johnston, N. (2016). QETLAB: A MATLAB toolbox for quantum entanglement, version 0.9. `http://qetlab.com`.

Kraus, K. (1971). General state changes in quantum theory. *Annals of Physics*, 64(2):311–335.

Leifer, M. S. and Poulin, D. (2008). Quantum graphical models and belief propagation. *Annals of Physics*, 323(8):1899–1946.

Leifer, M. S. and Spekkens, R. W. (2013). Towards a formulation of quantum theory as a causally neutral theory of Bayesian inference. *Physical Review A*, 88(5):052130.

Li, L., Jamieson, K. G., DeSalvo, G., Rostamizadeh, A., and Talwalkar, A. S. (2017). Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18:185:1–185:52.

M. Zhao, H. J. (2007). Norm observable operator models. Technical report, Jacobs University.

Miszczak, J. A. (2011). Singular value decomposition and matrix reorderings in quantum information theory. *International Journal of Modern Physics C*, 22(09):897–918.

Monras, A., Beige, A., and Wiesner, K. (2010). Hidden Quantum Markov Models and non-adaptive read-out of many-body states. *arXiv preprint arXiv:1002.2337*.

Nielsen, M. A. and Chuang, I. L. (2010). *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press.

Pillis, J. (1967). Linear Transformations which Preserve Hermitian and Positive Semidefinite Operators. *Pacific Journal of Mathematics*.

Qian, N. (1999). On the momentum term in gradient descent learning algorithms. *Neural Netw.*, 12(1):145–151.

Rabiner, L. R. (1986). An introduction to hidden markov models. *ieee assp magazine*, 3(1):4–16.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536.

Schuld, M., Sinayskiy, I., and Petruccione, F. (2015). An introduction to quantum machine learning. *Contemporary Physics*, 56(2):172–185.

Singh, S., James, M. R., and Rudary, M. R. (2004). Predictive state representations: A new theory for modeling dynamical systems. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*,

UAI '04, pages 512–519, Arlington, Virginia, United States. AUAI Press.

Srinivasan, S., Downey, C., and Boots, B. (2018a). Learning and Inference in Hilbert space with Quantum Graphical Models. In *Advances in Neural Information Processing Systems 31*.

Srinivasan, S., Gordon, G., and Boots, B. (2018b). Learning hidden quantum markov models. In *International Conference on Artificial Intelligence and Statistics*, pages 1979–1987.

Stiefel, E. (1935-1936). Richtungsfelder und fernparallelismus in n-dimensionalem mannig faltigkeiten. *Commentarii Math. Helvetici*, 8(305-353).

Thon, M. and Jaeger, H. (2015). Links between multiplicity automata, observable operator models and predictive state representations — a unified learning framework. *Journal of Machine Learning Research*, 16:103–147.

Towell, G. G., Noordewier, M. O., and Shavlik, J. W. (1991). Molecular biology (splice-junction gene sequences) data set. "Available at `https://archive.ics.uci.edu/ml/datasets`".

Warmuth, M. K. and Kuzmin, D. (2014). A Bayesian probability calculus for density matrices. *arXiv preprint arXiv:1408.3100*.

Wen, Z. and Yin, W. (2013). A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434.

Wiewiora, E. (2007). *Modeling probability distributions with predictive state representations*. PhD thesis, University of California, San Diego.

Wood, C. J., Biamonte, J. D., and Cory, D. G. (2015). Tensor networks and graphical calculus for open quantum systems. *Quantum Info. Comput.*, 15(9-10):759–811.

Yeang, C.-H. (2010). A probabilistic graphical model of quantum systems. In *Machine Learning and Applications (ICMLA), 2010 Ninth International Conference on*, pages 155–162. IEEE.

Zhao, M.-J. and Jaeger, H. (2010). Norm Observable Operator Models. *Neural Computation*, 22(7):1927–1959.

Życzkowski, K. and Bengtsson, I. (2004). On Duality between Quantum Maps and Quantum States. *Open Systems & Information Dynamics*, 11(1):3–42.