Manuscript (Survey): Quantum Diffusion Kernels on Graphs, Symmetries, Groups, and Speedups

Jue Xu*

June 21, 2022

Abstract

We discuss the quantum analogue of diffusion kernels on graphs. prove the quantum speedup in terms of groups and symmetries.

Contents

1 Introduction

group theoretical methods in machine learning by Kondor [Kon08]. diffusion kernel on graphs [KL02]

Many insightful and powerful models, like adiabatic quantum computation [Far+00], quantum random walks [Chi04]

*juexu@cs.umd.edu

1.1 Preliminary and Notations

a datum $\mathbf{x} := (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ is a vector where n is the number of *features*; dataset $\left\{\mathbf{x}^{(m)}, y\right\}$ where the *label* $y \in \{-1, 1\}$, Ω; quantum state $|z\rangle$ with $z \in 1, 2, \dots, 2^n$ where n is the number of qubits, computational basis; the binary representation $|\mathbf{x}\rangle = |x_1, x_2, \dots, x_n\rangle, x_i \in \{0, 1\}$. $|0\rangle^n$

1.2 SVM and Kernel trick

supervised learning [1]

Support Vector Machine

training set, test set. training stage, classification stage. exponentially large space; hyperplane $(\mathbf{w}, b)$ parametrized by a normal vector $\mathbf{w} \in \mathbb{R}^n$ and a bias term $b \in \mathbb{R}$. in the (high-dimensional) *feature space*. maximize the margin

$$\arg\max_f L(y, \tilde{y}) + \|\| \tag{1}$$

where the loss function L, slackness. called *support vector*. objective (cost function): *empirical risk* (error rate, loss function)

$$R_{emp}(\theta) = \frac{1}{|T|} \sum_{\mathbf{x} \in T} \mathbb{P}(\tilde{y} \neq y) \tag{2}$$

the dual quadratic program that (only uses access to the kernel) we maximize

$$L_D(\alpha) = \sum_{i=1}^{t} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{t} y_i y_j \alpha_i \alpha_j \mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) \tag{3}$$

subject to $\sum_{i=1}^{t} \alpha_i y_i = 0$ and $\alpha_i \geq 0$ for each i?.

construct the classifier

$$\tilde{m}(\mathbf{s}) = (sign)\left(\sum_{i=1}^{t} y_i \alpha_i^* \mathcal{K}(\mathbf{x}_i, \mathbf{s}) + b\right) \tag{4}$$

Kernel trick (method)

kernel trick: *feature map* the input data to higher dimension such that the data are linearly separatable in this feature space. only depend on the inner product to avoid the expensive (exponential) calculation. a *kernel function* (mapping) $\mathcal{K} : \Omega \times \Omega \mapsto \mathbb{R}$, a *(feature) mapping* $\phi : \Omega \mapsto \mathcal{H}_{\mathcal{K}}$

$$\mathcal{K}(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle \tag{5}$$

inner product (Dirac notation?).

Definition 1 (Feature map)**.** *feature map*;

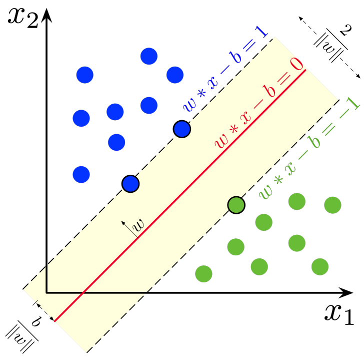
$$\Phi(\mathbf{x}) : \mathbb{R}^n \to \mathcal{H} \tag{6}$$

from a low dimensional space non-linearly in to a high dimensional Hilbert-space $\mathcal{H}$ which is commonly referred to as the *feature space*.

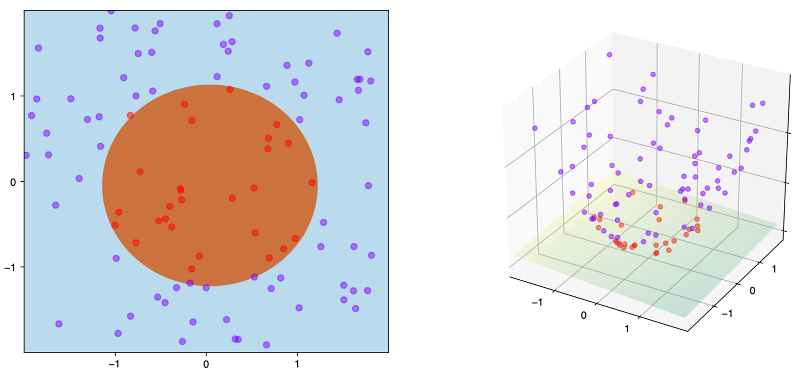
Definition 2 (Kernel function)**.** A function $\mathcal{K}$ is a valid kernel (in machine learning) if and only if? the matrix $\mathcal{K}(x, x')$ is symmetric and positive semi-definite.

Some common kernels:

[1] unsupervised learning



(a) linearly separable SVM



(b) kernel trick idea: SVM with kernel given by $\phi(\mathbf{x} := (a, b)) = (a, b, a^2 + b^2)$ and thus $\mathcal{K}(\mathbf{x}, \mathbf{y}) = \mathbf{x} \cdot \mathbf{y} + \|x\|^2 \|y\|^2$ that only depends on inner product. The training points are mapped to a 3-dimensional space where a separating hyperplane can be easily found. (from Wikipedia: Kernel method)

- polynomial kernel:
- radial:
- Gaussian kernel:

Definition 3 (Reproducing Kernel Hilbert Space)**.**

Theorem 1 (Representer)**.**

Hilbert space

2 Diffusion Kernels and Continuous-Time Quantum Random Walk

2.1 Classical diffusion kernels on graphs

[KL02] with Euclidean space $\Omega = \mathbb{R}^m$

Definition 4 (Adjacency matrix)**.** Given a (undirected, unweighted) graph $G = (V, E)$, its *adjacency matrix* $\hat{A}$ is defined as

$$\hat{A}(v, v') := \begin{cases} 1, & (v, v') \in E \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where the matrix entry is 1 if the two vertices (labels of the column and the row) are connected by an edge, otherwise 0.

Definition 5 (Graph Laplacian)**.** With the adjacency matrix $\hat{A}$, the graph Laplacian is defined as

$$\hat{\mathfrak{L}} := \hat{A} - \hat{D} \tag{8}$$

where $\hat{D}_{vv} := \mathsf{deg}(v)$ is its diagonal degree of (vertex v) matrix.

Remark 1. Graph Laplacian $\hat{\mathfrak{L}}$ is the discrete version of (continuous) Laplacian operator ∇^2.

Lemma 1. *exponential of i.e.,* $e^{\beta \hat{H}}$ *is a valid kernel*

2.1.1 Diffusion, heat equation, random walk,

The continuous-time random walk on G is defined as the **solution of the differential equation**

$$\frac{\mathrm{d}}{\mathrm{d}t}p_j(t) = \sum_{k\in V} \hat{\mathfrak{L}}_{jk}\ p_k(t), \tag{9}$$

where $p_j(t)$ denotes the probability associated with vertex j at time t and $\hat{\mathfrak{L}}$ is Graph Laplacian.

Since the columns of L sum to 0

$$\frac{\mathrm{d}}{\mathrm{d}t}\sum_{j\in V} p_j(t) = \sum_{j,k\in V} \hat{\mathfrak{L}}_{jk}p_k(t) = 0 \tag{10}$$

which shows that an initially normalized distribution remains normalized: the evolution of the continuous-time random walk for any time t is a *stochastic process. random walk*, *heat equation*

2.2 Continuous-time quantum random walk

The continuous-time quantum random walk [CFG02] is the quantum analogue of classical diffusion (continuous-time random walk). By a direct observation, Eq. (9) is very similar to the time-dependent (evolution) schrodinger equation governed by a Hamiltonian operator $\hat{H}$

$$i\hbar\frac{\mathrm{d}}{\mathrm{d}t}|\psi\rangle = \hat{H}|\psi\rangle \tag{11}$$

except that the factor of $i\hbar$.

Definition 6 (Quantum propagator, kernel, transition)**.**

2.3 Relation and examples

2.3.1 Ring (closed line)

classical kernel

$$\mathcal{K}() \tag{12}$$

quantum propagation (kernel)

$$\begin{aligned} \langle z_F|e^{-\mathrm{i}t\hat{H}_0}|z_I\rangle &= \sum_{p=1}^{N} e^{-\mathrm{i}t2\cos\left(\frac{2\pi}{N}p\right)+\mathrm{i}\frac{2\pi}{N}p(z_I-z_F)} && (13)\\ &\approx e^{2\mathrm{i}t}(-\mathrm{i})^d J_d(2t) && (14) \end{aligned}$$

Remark 2. The random walk on this graph starting from the origin (in either continuous or discrete time) typically moves a distance proportional to $\sqrt{t}$ in time t. In contrast, the quantum random walk evolves as a wave packet with speed 2.

2.3.2 Tree

2.3.3 Hyercube

2.3.4 Cayley graphs

Definition 7 (Cayley graph)**.** Cayley graph is a graph that encodes the abstract structure of a group.

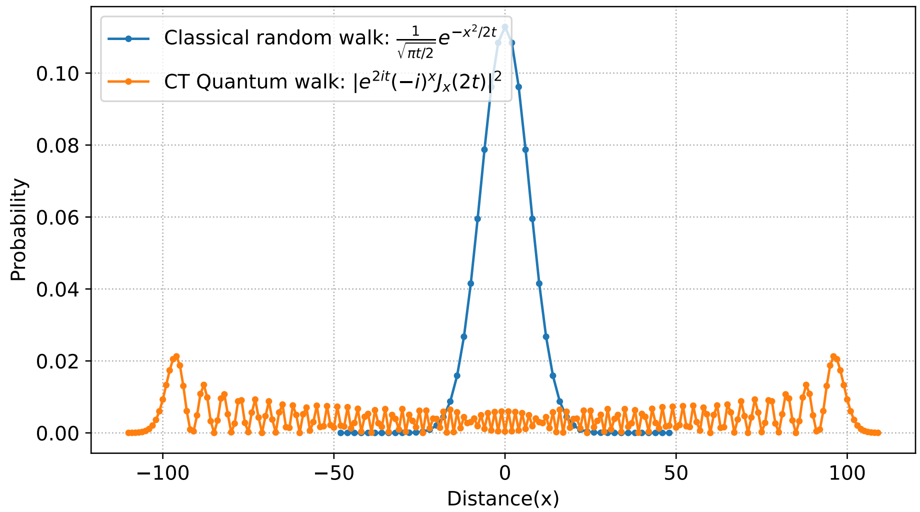


Figure 2

3 Quantum Speedups via QKE

A quantum version of this approach has already been proposed in [RML14], where an exponential improvement can be achieved if data is provided in a coherent superposition.

Remark 3. However, when data is provided in the conventional way, i.e. from a classical computer, then the methods of [15] cannot be applied.

input model, quantum RAM; quantum-inspired [Tan19]

3.1 Previous works: Quantum Kernel Estimation

quantum kernel estimation [SK19] [Hav+19] the quantum state space (Hilbert space) as the feature space to still obtain a quantum advantage mapping the input data non-linearly to a quantum state

$$\Phi(\mathbf{x}) : \Omega \to |\Phi(\mathbf{x})\rangle\langle\Phi(\mathbf{x})| \,, \tag{15}$$

the direct quantum analogy of classical Feature map

3.1.1 Explicit method (Quantum Variational Classification)

variational quantum circuit: generates a separating hyperplane in the quantum feature space

1. $\mathbf{x} \in \Omega$ (feature) mapped to a quantum state by applying a unitary circuit $U_{\Phi(\mathbf{x})}$ to a reference (initial) state $|0\rangle^n$
2. a short depth quantum circuit $W(\theta)$
3. for binary classification, apply a binary measurement $\{M_y\} = 2^{-1}(\mathbb{1} + y\mathbf{f})$?
4. to obtain the empirical distribution $p_y(\mathbf{x})$, perform repeated measurement shots. then assign the label according to p_y?

3.1.2 Implicit method (QKE)

estimate the kernel function quantumly and implement a conventional SVM. Rather than using a variational quantum circuit to generate the separating hyperplane, we use a classical SVM for classification.

1. the kernel $\mathcal{K}(\mathbf{x}, \mathbf{x}')$ is estimated on a quantum computer
2. the quantum computer is used a second time to estimate the kernel for a new datum (test) $\mathbf{s} \in S$ with all the support vectors.

The kernel entries are the fidelities between different feature vectors. The overlap can be estimated directly from the transition amplitude

$$\left|\left\langle \Phi(\mathbf{x}) \middle| \Phi(\mathbf{x}') \right\rangle\right|^2 = \left| \langle 0^n | U^\dagger_{\Phi(\mathbf{x})} U_{\Phi(\mathbf{x}')} | 0^n \rangle \right|^2 \tag{16}$$

measure the final state in the Z-basis R-times and record the number of $|o^n\rangle$. The frequency of this string is the estimate of the transition probability. The kernel entry is obtained to an additive sampling error of $\tilde{\epsilon}$ when $\mathcal{O}(\tilde{\epsilon}^{-2})$ shots are used. *quantum feature map.* quantum propagation, kernel in the path-integral formalism.

Theorem 2 ([Chi+03]). *There exists exponential separation with respect to query complexity in the adjacency matrix model*

[Zhe+22]
[LAT21]

3.2 Provable: Symmetries, graph properties, and quantum speedups

symmetric functions rule out exponential speedup [Ben+20]

3.2.1 Permutation, symmetry, and speedup

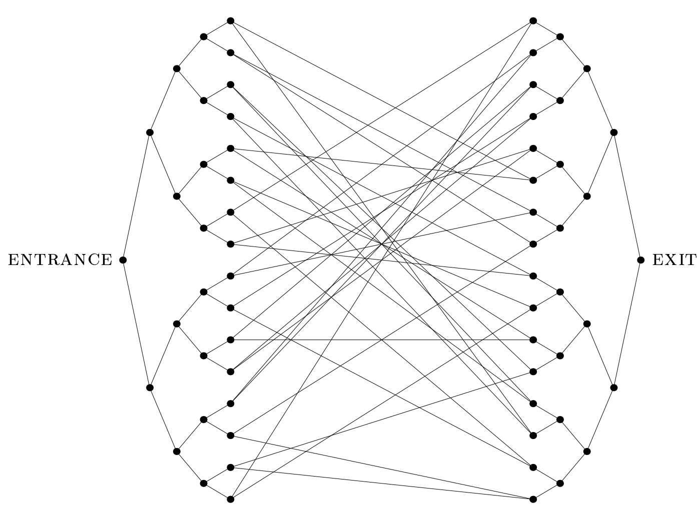


Figure 3: [Chi+03]

3.3 Heuristic: Group, invariance, symmetries, physical systems

3.3.1 Group, symmetries in physics

covariant [Gli+21] group theory, [Kon08]; symmetries in physics [Bog+20]; equivariant CNN [Zhe+22]

3.3.2 Group theory and machine learning

[KL02]

4 Experiment

4.1 Datasets and benchmark

4.1.1 Artificial data

we generate artificial data that can be fully separated by our feature map.

4.1.2 Real dataset

JET?; quantum phase transition? order?

5 Discussion and Conclusion

References

[Ben+20] Shalev Ben-David et al. Symmetries, Graph Properties, and Quantum Speedups. *2020 IEEE 61st Annu. Symp. Found. Comput. Sci. FOCS* (Nov. 2020), pp. 649–660. arXiv: 2006.12760 (cit. on p. 6).

[Bog+20] Alexander Bogatskiy et al. Lorentz Group Equivariant Neural Network for Particle Physics. Proceedings of the 37th International Conference on Machine Learning. International Conference on Machine Learning. PMLR, Nov. 21, 2020, pp. 992–1002 (cit. on p. 7).

[CFG02] Andrew M. Childs, Edward Farhi, and Sam Gutmann. An Example of the Difference between Quantum and Classical Random Walks. *Quantum Inf. Process.* 1.1/2 (2002), pp. 35–43. arXiv: quant-ph/0103020 (cit. on p. 4).

[Chi+03] Andrew M. Childs et al. Exponential Algorithmic Speedup by Quantum Walk. *Proc. Thirty-Fifth ACM Symp. Theory Comput. - STOC 03* (2003), p. 59. arXiv: quant-ph/0209131 (cit. on p. 6).

[Chi04] Andrew MacGregor Childs. "Quantum Information Processing in Continuous Time". Thesis. Massachusetts Institute of Technology, 2004 (cit. on p. 1).

[Far+00] Edward Farhi et al. "Quantum Computation by Adiabatic Evolution". Jan. 28, 2000. arXiv: quant-ph/0001106 (cit. on p. 1).

[Gli+21] Jennifer R. Glick et al. "Covariant Quantum Kernels for Data with Group Structure". May 7, 2021. arXiv: 2105.03406 [quant-ph] (cit. on p. 7).

[Hav+19] Vojtech Havlicek et al. Supervised Learning with Quantum Enhanced Feature Spaces. *Nature* 567.7747 (Mar. 2019), pp. 209–212. arXiv: 1804.11326 (cit. on p. 5).

[KL02] Risi Imre Kondor and John Lafferty. Diffusion Kernels on Graphs and Other Discrete Structures (2002), p. 8 (cit. on pp. 1, 3, 7).

[Kon08] Risi Kondor. “Group Theoretical Methods in Machine Learning”. Thesis. Columbia University, 2008 (cit. on pp. 1, 7).

[LAT21] Yunchao Liu, Srinivasan Arunachalam, and Kristan Temme. A Rigorous and Robust Quantum Speed-up in Supervised Machine Learning. *Nat. Phys.* 17.9 (Sept. 2021), pp. 1013–1017. arXiv: `2010.02174 [quant-ph]` (cit. on p. 6).

[RML14] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum Support Vector Machine for Big Data Classification. *Phys. Rev. Lett.* 113.13 (Sept. 25, 2014), p. 130503. arXiv: `1307.0471 [quant-ph]` (cit. on p. 5).

[SK19] Maria Schuld and Nathan Killoran. Quantum Machine Learning in Feature Hilbert Spaces. *Phys. Rev. Lett.* 122.4 (Feb. 1, 2019), p. 040504. arXiv: `1803.07128 [quant-ph]` (cit. on p. 5).

[Tan19] Ewin Tang. A Quantum-Inspired Classical Algorithm for Recommendation Systems. *Proc. 51st Annu. ACM SIGACT Symp. Theory Comput.* (June 23, 2019), pp. 217–228. arXiv: `1807.04271` (cit. on p. 5).

[Xu21] Jue Xu. “On Lagrangian Formalism of Quantum Computation”. Dec. 7, 2021. arXiv: `2112.04892 [quant-ph]` (cit. on p. 8).

[Zhe+22] Han Zheng et al. “Speeding up Learning Quantum States through Group Equivariant Convolutional Quantum Ans\"atze”. Jan. 20, 2022. arXiv: `2112.07611 [math-ph, physics:quant-ph, stat]` (cit. on pp. 6, 7).

A Machine Learning, Group Theory, and Lagrangian

A.1 Machine learning

A.2 Group theory and symmetries

A.2.1 Representation

A.3 Lagrangian formalism

[Xu21]