Data Classification using Microsoft Quantum Machine Learning Library

by Jayprakash Asolkar

Supervisor: Prof. Rozenn Dahyot

Declaration

I declare that this thesis has not been submitted as an exercise for a degree at this or any other university and it is entirely my own work.

I agree to deposit this thesis in the University′s open access institutional repository or allow the library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

Jayprakash Asolkar

July 11, 2020

Forewords

This is an example of a possible template designed for TCD students for writing a report in LATEXwritten by Prof R. Dahyot[1].

---

[1]https://www.scss.tcd.ie/Rozenn.Dahyot/

Acknowledgments

To be completed.

Abstract

To be completed.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

1.2 Research Question

Chapter 2

Background Research

The development of Quantum Mechanics over the last century has paved the way for the new paradigm of computation. Paul Benioff with his research on Quantum Information and Quantum mechanical model of Turing Machines pioneered the field of Quantum Computing. Quantum computation is based on the postulates of quantum mechanics which define the way quantum systems behave. In the last couple of decades, many theoretical quantum algorithms have been developed which suggest a possible superiority of quantum computers over their classical counterparts. This section discusses, in brief, the basics of quantum computing required to build more complex algorithms which can help to solve some of the challenging problems faced by classical computers. Furthermore, it gives an overview of state of the art applications of quantum computing across various domains.

2.1 Quantum Bit

The most basic unit of computation for classical computers is a bit (binary digit) which is used to store and process information. Quantum computers make use of a similar unit of information called Quantum Bit or Qubit. Unlike the classical bit which can represent either a zero state or one state at any given time, a qubit can simultaneously be in a superposition state of both $|0\rangle$ and $|1\rangle$ state. More precisely, the state of a single qubit is a unit vector in a 2-dimensional complex vector space called Hilbert Space. The special states $|0\rangle$ and $|1\rangle$, of the qubit vector are called the orthonormal basis of the Hilbert space. Any arbitrary state $|\psi\rangle$ of the qubit can be defined as a linear combination of the basis vectors.

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \qquad where \quad \langle\psi|\psi\rangle = |\psi|^2 = 1 \tag{2.1}$$

In the equation 2.1, α and β are complex numbers and $|\alpha|^2$, $|\beta|^2$ are the probabilities of measuring the qubit in $|0\rangle$ or $|1\rangle$ basis states respectively. The matrix representation of the arbitrary qubit state $|\psi\rangle$ and basis states $|0\rangle$, $|1\rangle$ is given by equation 2.2. On an ideal quantum computer, measurement

of a qubit in $|0\rangle$ basis state yields the binary digit 0, hundred percent of the times. Similarly there is 100% probability of a qubit in $|1\rangle$ state resulting in the measurement of binary digit 1.

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \qquad |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{2.2}$$

The pure state of a qubit can also be thought as a point residing on the surface of the Bloch Sphere. Figure 2.1 shows the Bloch sphere representation of the qubit state vector.
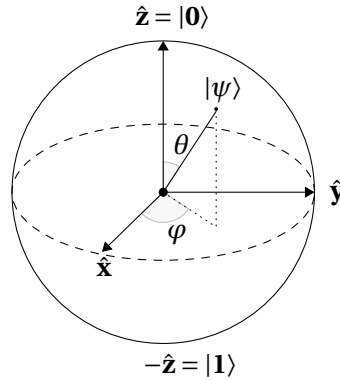

Figure 2.1: Bloch Sphere

North and south poles of the sphere represent $|0\rangle$, $|1\rangle$ states respectively. As shown in equation 2.3, by ignoring the global phase, angles θ and φ can be used to define any arbitrary state of a qubit.

$$|\psi\rangle = cos\,\frac{\theta}{2}\,|0\rangle + e^{i\varphi}\,sin\,\frac{\theta}{2}\,|1\rangle \tag{2.3}$$

Real quantum computers make use of atoms, trapped ions, electrons, photons to realize quantum bits. Physically $|0\rangle$ and $|1\rangle$ states may correspond to polarization of a photon or the alignment of nuclear spins.

2.2 Multiple Qubits and Entanglement

A single qubit alone is not sufficient to perform quantum computation. The power of quantum computers lies in the use of registers of qubits and their co-related states. Using n qubits in superposition state of $|0\rangle$ and $|1\rangle$, for eg. $\dfrac{|0\rangle + |1\rangle}{\sqrt{2}}$, we can represent 2^n states simultaneously in a multi-qubit system. However, it should be noted that measurement of the state of a qubit collapses the state to one of the basis states and destroys the quantum information. Hence, at any given time, only a single state can be retrieved through measurement of a quantum system. This remains a challenge in developing efficient quantum algorithms.

A multiple qubit system with n qubits has 2^n computational basis states. These basis states of a multi-qubit system can be obtained by using tensor product of the basis states of individual qubits (For eg. $|00\rangle = |0\rangle \otimes |0\rangle$). Equation 2.4 represents an arbitrary state $|\psi\rangle$ in 2-qubit system.

$$|\psi\rangle = \alpha |00\rangle + \beta |01\rangle + \gamma |10\rangle + \delta |11\rangle \quad where \quad |\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1 \tag{2.4}$$

An important feature of multi-qubit quantum system is the ability to prepare entangled states of two or more qubits. The states of 2 entangled qubits affect each other even if the qubits are separated by a long distance. The entangled state $\frac{|00\rangle + |11\rangle}{\sqrt{2}}$ is called the Bell state or EPR pair and is the basic ingredient of multiple quantum algorithms. The terminology entanglement of two qubits specifies that the measurement of the state of only a single qubit among the EPR pair is sufficient to predict the outcome of measurement of the second qubit with absolute certainty.

2.3 Quantum Gates

For any meaningful computation, the ability to manipulate states of the qubits is essential. Various quantum gates are the means to linearly transform the state vector of a single qubit or register of multiple qubits. Quantum computers make use of lasers, magnetic fields or other technologies to physically alter the qubit state.

Similar to classical logic gates namely AND, OR, XOR etc., quantum computers make use of quantum gates which linearly change the qubit state vector. The Pauli-X gate is similar to classical NOT gate and acts on a single qubit to change the state from $|0\rangle$ to $|1\rangle$ and vice versa. The Pauli-X gate is represented in matrix format as shown in equation 2.5.

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{2.5}$$

The quantum Pauli-X gate acts linearly on the arbitrary state $|\psi\rangle$ to swap the measurement probabilities of $|0\rangle$ and $|1\rangle$ vectors. The equation 2.6 shows the application of the Pauli-X gate as the inner product of the gate with the state vector.

$$X|\psi\rangle = \alpha X |0\rangle + \beta X |1\rangle = \alpha \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \beta \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \alpha |1\rangle + \beta |0\rangle \tag{2.6}$$

The Pauli-Y and Pauli-Z gates are other two gates in the Pauli set and both of these act on a single qubit. Another important single qubit gate is the Hadamard gate. The Hadamard gate creates an equal superposition of both the computational basis states. Matrix representation of Pauli-Y, Pauli-Z and Hadamard gates is given by equation 2.7.

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \qquad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{2.7}$$

Quantum computation also makes use of gates acting on states of multiple qubits. A 4x4 CNOT gate acts on a register of 2 qubits. One of the qubits acts as the control qubit whereas another qubit in the set is called the target qubit. Based on the state of control qubit, the CNOT gate acts on the target qubit to flip its state. If the control qubit is in the state $|0\rangle$ then the CNOT gate does nothing to the target qubit and its state remains unchanged, whereas if the control qubit is in the state $|1\rangle$, the CNOT gate acts as a X gate on the target qubit. The dimensions of the matrix of a multi-qubit gate acting on n qubits is $2^n \text{x} 2^n$.

$$U_{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.8}$$

All quantum gates are unitary matrices i.e. the hermitian conjugate or adjoint of the gate matrix is also the inverse of the matrix. A hermitian conjugate is obtained by flipping the given matrix over its diagonal and changing the sign of the imaginary parts of the complex numbers. This constraint on the quantum gates makes them reversible. Hence, information about the input to the quantum gate never gets lost unlike the classical logic gates. Equation 2.9 is true for every quantum gate U and depicts that applying any gate twice on input qubit states restores the original quantum state. Also, unitary matrices maintain the normalization condition of the qubit state given by $|\alpha|^2 + |\beta|^2 = 1$.

$$UU^\dagger = I \qquad where \ U^\dagger = Adjoint(U) \tag{2.9}$$

As shown in equations 2.10 and 2.12, the reversibility of quantum gates can be verified for any of the gates discussed so far.

$$HH = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I \tag{2.10}$$

$$HH |\psi\rangle = I |\psi\rangle = |\psi\rangle \tag{2.11}$$

Effect of any arbitrary quantum gate on the state of a qubit can be thought of as a rotation of the state vector around different axes in the Bloch sphere. Additionally, any arbitrary transformation of the qubit state can be achieved by using combination of few basic gates.

2.4   Quantum Circuits

Quantum algorithms are depicted using the circuit model of computation. A quantum circuit comprises of register of qubits, quantum gates and measurements in classical bits. Logically, the simplest possible quantum circuit is a quantum wire which retains the state of a qubit over a period of time or a communication channel. However, this is the most difficult quantum circuit to implement physically, as the state of a qubit gets easily affected by its environment, corrupting the quantum information.
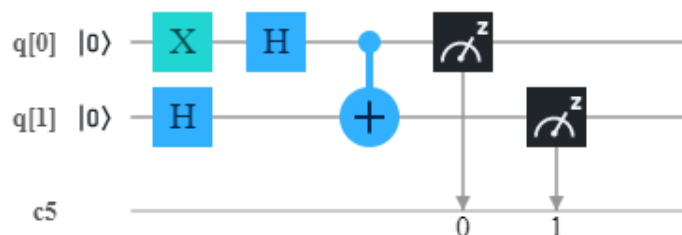

Figure 2.2: An example of quantum circuit visualized using IBMQ circuit composer

Figure 2.2 shows the composition of a quantum circuit. Two qubits q_0 and q_1 of the quantum register are prepared in the $|0\rangle$ basis state. The qubit q_0 is passed through the quantum wire across the X and Hadamard gates, whereas the qubit q_1 is passed through the quantum wire across the Hadamard gate. The 2-qubit CNOT gate uses q_0 as the control qubit and works on the target qubit q_1. Finally both the qubits are measured on the classical channel c5. After the measurement, the state of each of the qubits collapses to the computational basis state. More complex circuits can be built in similar way to realize various quantum algorithms which can utilize quantum properties such as superposition and entanglement.

2.5   Quantum Computing - Hype and Promises

The Deutsch-Jozsa algorithm was one of the first algorithms which showed that the quantum computers can be used to achieve exponential speedups using quantum parallelism over the best known classical algorithms. Shor's prime factoring algorithm was a catalyst for the growing interest in quantum computing and quantum cryptography. It makes use of the efficiency of quantum fourier transform to find prime factors of given number N in polynomial time. No known classical algorithm exists that can acheive the same task polynomially even on best super computers. Grover's search algorithm provides quadratic speed improvement for searching a database with unsorted entries. Currently available quantum computers are still in early stages of development and thus restrict the practical

verification of many theoretical algorithms which claim quantum supremacy. Furthermore, these machines are prone to errors due to environmental interference and decoherence, which is an active area of research in the field of quantum computation. However, steady growth of number of qubits and power of quantum computers, cloud access to physical systems and arrival of various software development kits for programmers all around the world to develop and test new algorithms, are some of the encouraging factors for research in this area. The no-cloning theorem of quantum bits states that quantum information stored in one qubit can not be copied to another qubit without altering the state of the original qubit. This phenomenon has implications in the field of quantum cryptography and future of data security. Quantum chemistry is a major field of research as quantum computers can simulate quantum systems efficiently. Classical computers require exponentially increasing resources for simulation of such quantum information. Multiple companies have recognized the vast potential of quantum computers and are investing heavily in the field. Quantum machine learning, image processing, data analysis are few more fields that are gaining interest from scholars all around the globe.

2.6   Introduction to Quantum Algorithms

As discussed in previous sections a quantum algorithm can be described using a quantum circuit consisting of qubits prepared in the desired state, gates acting on the state vector of the qubits and measurement operations in the computational basis states. This section discusses in brief few of the most important algorithms in quantum computation. Discussion on mathematical derivations of the algorithms is out of the scope of this dissertation. All these algorithms form the basis of many complex quantum algorithms and are essential in understanding how quantum computers can be utilized to solve problems.

2.6.1   Quantum Superposition

Quantum Superposition is a quantum mechanical phenomena wherein a single qubit occupies multiple possible states at any given time. Although this state is not directly observable by any measurement apparatus, its applicability in quantum parallelism gives an edge to quantum computers over their classical counterparts to carry out multiple operations on the qubits at any single instance. If the computational states of measurement are $|0\rangle$ and $|1\rangle$, then an equal superposition of these two states can be obtained by applying the Hadamard transformation to the register of input qubits. Figure 2.3 shows the circuit to achieve the Hadamard transformation which achieves an equal superposition state of all possible states of individual qubits. If n is the number of input qubits and N=2^n are the

computational basis states of the combination of the qubits, then the probability of measuring the qubits in any of the basis states is equal to $\frac{1}{N}$. For the 3 qubit circuit, possible basis states are $|000\rangle$, $|001\rangle$, $|010\rangle$, $|011\rangle$, $|100\rangle$, $|101\rangle$, $|110\rangle$ and $|111\rangle$. After hadamard transform, the measurment probability of each of these states is $\frac{1}{8}$. This forms the basis for the algorithm of random number generation in quantum computers.
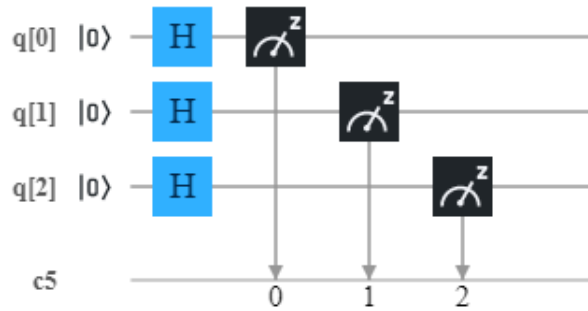

Figure 2.3: Circuit to create equal superposition of the states of 3 qubits

2.6.2 Quantum Teleportation

Quantum Teleportation makes use of quantum entanglement of qubits to communicate the information about an arbitrary state of a qubit between two parties. Consider a scenario where Alice has a qubit in the state $\alpha|0\rangle + \beta|1\rangle$. As the knowledge about the parameters α and β can not be obtained by classical measurement, Alice can not communicate this information directly to Bob using classical channels. However, Alice and Bob both possess a single qubit of the entangled pair $\frac{|00\rangle + |11\rangle}{\sqrt{2}}$. This entangled pair can be used to send arbitrary state $|\psi\rangle$ of a qubit from Alice to Bob.
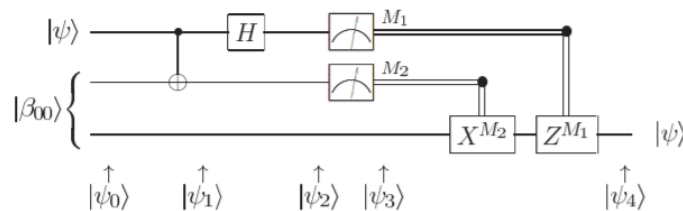

Figure 2.4: Circuit for Quantum Teleportation

In this algorithm, Alice has a qubit in arbitrary state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ and one of the entangled pair of qubits, while the other qubit of the entangled pair is in the possession of Bob. Initial state of the 3 qubits together can be specified as $|\psi_0\rangle = \frac{1}{\sqrt{2}}\left[\alpha|0\rangle(|00\rangle + |11\rangle) + \beta|1\rangle(|00\rangle + |11\rangle)\right]$. Alice applies a CNOT gate with the arbitrary qubit as a control qubit and entangled qubit as the target qubit. This changes the state to $|\psi_1\rangle = \frac{1}{\sqrt{2}}\left[\alpha|0\rangle(|00\rangle + |11\rangle) + \beta|1\rangle(|00\rangle + |01\rangle)\right]$. The state $|\psi_2\rangle$, after applying the

Hadamard gate to the arbitrary qubit is given by equation 2.12.

$$|\psi_2\rangle = \frac{1}{2}\left[|00\rangle\left(\alpha|0\rangle + \beta|1\rangle\right) + |01\rangle\left(\alpha|1\rangle + \beta|0\rangle\right) + |10\rangle\left(\alpha|0\rangle - \beta|1\rangle\right) + |11\rangle\left(\alpha|1\rangle - \beta|0\rangle\right)\right] \qquad (2.12)$$

Alice does the measurement of both the qubits and shares the classical measurement result with Bob. Depending on the result of Alice's measurements, Bob applies unitary transformations to his qubit to obtain the arbitrary state.

Measurement Result	Bob's Qubit State	Transformation
00	$\alpha|0\rangle + \beta|1\rangle$	No Transformation Required
01	$\alpha|1\rangle + \beta|0\rangle$	Apply Pauli-X gate
10	$\alpha|0\rangle - \beta|1\rangle$	Apply Pauli-Z gate
11	$\alpha|1\rangle - \beta|0\rangle$	Apply Pauli-X followed by Pauli-Z gate

Table 2.1: Transformation on Bob's qubit for Teleportation

This algorithm is in accordance with the no-cloning theorem and does not copy the state of Alice's qubit to Bob's qubit. The arbitrary state of Alice's qubit is destroyed due to the measurement before teleportation. This algorithm is the basis of quantum communication.

2.6.3   Grover's Search Algorithm

Consider a database with N entries of bit strings without any well defined structure. This database contains a bit string x_1 such that f(x_1)=1. For all other bit strings in the database f(x)=0. In the worst case, best known classical algorithms require to scan all the entries in the database to find x_1 resulting in the complexity of O(N). Grover's search algorithm solves this problem with a quadratic speedup over the classical algorithms.
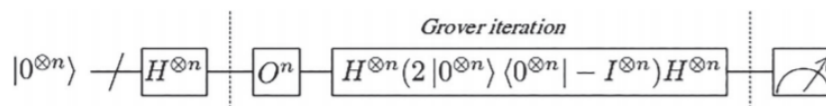

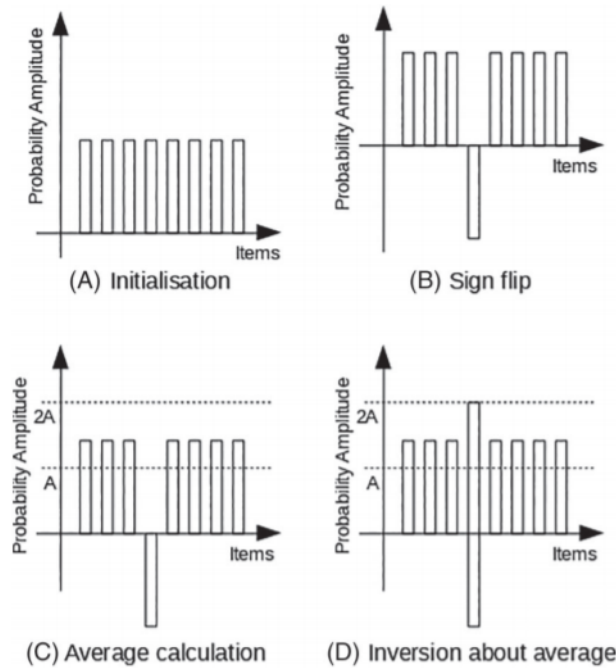Figure 2.5: Grovers Algorithm - Circuit Diagram

Figure 2.6: Grovers Algorithm - Steps

Grovers algorithm works by initializing state vector in equal superposition of n=log_2N qubits using the hadamard transformation. It then applies the Grover's iteration on the register of qubits which involves phase flip of the desired state followed by state inversion about the mean as shown in figure 2.6. Grover's iteration is repeated $\sqrt{N}$ times to achieve the desired measurement with high probability. The circuit diagram for Grover's algorithm is shown in figure 2.5.

2.6.4   Shor's Factoring Algorithm

Shor's algorithm makes use of quantum fourier transform to find prime factors of a number within polynomial time as opposed to classical algorithms which take an exponential time. This algorithm can be understood by taking a simple example of finding the prime factors of number 15. The algorithm selects a random number less than 15 which is co-prime with 15. Using the randomly selected number Shor's period finding algorithm finds the period r such that a^r mod N = 1 and $r \neq 0$. If r is even, then there is a high probability that the greatest common divisor of either $\left(a^{\frac{r}{2}}+1\right)$ and N or $\left(a^{\frac{r}{2}}-1\right)$ and N, is a prime factor of N. If r is odd, then the algorithm repeats the steps till it finds an even r. For our example, the period is 4 as shown in the table 2.2. So, prime factors of 15 can be calculated by finding the GCD of 7^2 + 1 = 50 & 15 = 5 and the GCD of 7^2 - 1 = 48 & 15 = 3. The part of period finding in this algorithm makes use of quantum fourier transform which takes polynomial time.

7^0 mod 15 $\equiv$ 1 mod 15
7^1 mod 15 $\equiv$ 7 mod 15
7^2 mod 15 $\equiv$ 4 mod 15
7^3 mod 15 $\equiv$ 13 mod 15
7^4 mod 15 $\equiv$ 1 mod 15
:
:

Table 2.2: Period finding of Shor's algorithm

2.7   Quantum Data Classification

Classical machine learning technology has been developed over the last few decades and has provided ingenious solutions to various computational problems. Applications utilizing machine learning techniques span across fields like healthcare, industrial automation, image processing, automobiles etc. Recent progress in neural networks has improved the accuracy of these applications even further. However, efficiency and accuracy of most of the machine learning algorithms depends on big data analysis and this is a challenge for classical computers due to the huge growth of required data.

Recent work in the field of Quantum Machine Learning claims to have achieved significant speed improvements over best available classical algorithms. Quantum ML algorithms can be broadly classified into 3 categories. The first category contains classical ML algorithms converted to be executed on quantum computers. Second category of QML algorithms are called quantum inspired ML algorithms and take inspiration from quantum principles to improve performance. Hybrid classical-quantum machine learning algorithms fall into third category and make use of classical computers along with quantum subroutines. Many of the Quantum ML techniques such as Quantum Support Vector Machines and Quantum Principle Component Analysis are based on the HHL quantum algorithm for solving linear equations. While, quantum K-Means clustering algorithm makes use of Grover's search algorithm along with Phase estimation and Swap-Test.

Data Classification is one of the most basic machine learning application. Recently many supervised and unsupervised learning algorithms are implemented in quantum computers for classifying input data vectors. For these methods to work, classical data which is represented in bit strings needs to be encoded in the states of qubits. Basic encoding of classical data does 1:1 mapping of bit information into qubits. In this method, a bit string 1111 will be represented using qubit state vector as $|1111\rangle$. Although this is a simple method of data representation, it does not provide any quantum advantage. Another approach called Amplitude encoding stores the classical information in the am-

plitudes α and β of the state vector $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. This method is used in many QML algorithms and requires the classical data vector to be normalized before encoding.

Due to growing interest in extracting the benefits of possible quantum advantage, multiple researchers and organizations have introduced development kits for programming quantum computers. Various quantum algorithms can be executed on quantum simulators as well as real quantum computers via cloud access. IBM's QISKIT, Google Cirq, Microsoft's Quantum Development Kit (QDK), XANADU's PennyLane are few of the available frameworks for quantum development. Few of these frameworks also have implementations for quantum machine learning algorithms. This dissertation presents a review and evaluation of machine learning library in Microsoft's QDK framework. This library can be used for the supervised QML tasks of data classification.

The machine learning library in QDK uses hybrid QML approach incorporating both classical and quantum computers. It uses a shallow circuit utilizing limited number of qubits and quantum gates to reduce errors on currently available noisy quantum computers and claims to achieve comparable results with current state of the art methods. Amplitude encoding is used to represent classical data into the state of the qubits. Quantum entanglement captures the correlation between different qubits to predict classes of input data. The library is open sourced by Microsoft and is available for review. Next sections discuss the working of various parts of the classification algorithm along with its evaluation on different datasets.

Chapter 3

Work to date

Chapter 4

Future work

Bibliography

Schuld, M., Bocharov, A., Svore, K. M., Wiebe, N. (2020). Circuit-centric quantum classifiers. Physical Review A, 101(3). doi:10.1103/physreva.101.032308

Appendix A

my appendix