7-24-2023 12:00 PM

Forecasting the Future Capacities of Superconducting Quantum Computers: Extending Moore's Law Through Machine Learning

Christopher Tam, *Western University*

Supervisor: Capretz, Luiz F., *The University of Western Ontario*
A thesis submitted in partial fulfillment of the requirements for the Master of Engineering Science degree in Electrical and Computer Engineering

Abstract

Quantum computing has emerged as a promising technology that can perform certain tasks exponentially faster than classical computers. Despite the potential for quantum computers to revolutionize the field of computing, the development of fault-tolerant quantum computers remains a critical challenge. Moore's Law has accurately predicted the exponential growth in the capacity of classical computers, with transistor capacity doubling roughly every year. This prediction, established in the 1960s, held true until the early 2010s. However, the emergence of quantum computers raises questions about how to predict the rate of development these technologies.

This work presents a novel approach using machine learning to extend classical Moore's Law into a quantum Moore's Law. Unlike previous attempts, which relied on limited quantum computer data, this model incorporates historical classical transistor data to predict qubit capacities. This thesis proposes a novel approach to forecasting the future capacities of superconducting qubits and gate speeds using machine learning.

The proposed model builds upon Moore's Law and its predictions for the transistor capacity of classical computers. First, it establishes a polynomial relationship between the number of qubits and the number of classical transistors. Then, it trains a machine learning model to predict the number of classical transistors for future years. This prediction is used in conjunction with the established relationship to estimate the number of qubits for a given year. The same methodology is applied on data on the best achieved classical computations per second values to predict the speed of execution of quantum gates in the future.

The findings indicate that the proposed model outperforms previously proposed methods in predicting qubit capacities, suggesting an improved method for predicting the future capacities of superconducting qubits and gate speeds based on the relationship between qubit and classical transistor capacities. Using a data-driven approach, the model can incorporate new data as quantum milestones are achieved.

In this study, we present a novel approach to predicting the growth of quantum computing by extending and evolving classical Moore's Law using machine learning. Our proposed model makes use of historical information on classical transistors to estimate recent qubit capacities more accurately than earlier studies, showing improved prediction accuracy in comparison to previous work. The proposed model provides valuable insight into the potential trajectory of quantum computing technology if Moore's Law continues to hold in this domain.

Summary for Lay Audience

Quantum computing has the potential to revolutionize computing by performing tasks much faster than classical computers. However, developing fault-tolerant quantum computers is a major challenge. While Moore's Law accurately predicted the growth of classical computers, the emergence of quantum computers raises questions about how to predict their development. This study presents a novel approach that combines machine learning with historical classical transistor data to forecast the capacities of quantum computers. The proposed model establishes a relationship between the number of qubits and classical transistors and uses machine learning to predict future transistor capacities. This prediction is then used to estimate the number of qubits for a given year. The same approach is applied to predict the speed of quantum gate execution. The results show that this model outperforms previous methods, providing more accurate predictions of qubit capacities. By incorporating new data as quantum milestones are achieved, the model can continuously improve its predictions. This study offers valuable insights into the potential growth of quantum computing if Moore's Law continues to apply in this domain. The proposed model extends and evolves classical Moore's Law using machine learning, leveraging historical information on classical transistors to estimate recent qubit capacities more accurately. Overall, this approach provides a promising method for predicting the future capacities of superconducting qubits and gate speeds in quantum computing.

Keywords: Moore's Law, quantum computing, quantum computer, quantum algorithms, qubits, quantum gate speed

Acknowledgements

I would like to express my deepest appreciation and gratitude to my supervisor, Dr. Luiz Fernando Capretz, for his invaluable guidance and support throughout my Master's program. His dedication, expertise, and unwavering commitment to my academic and professional growth has been instrumental in shaping my research journey. Under Dr. Capretz's mentorship, I have had the privilege of receiving funding from his NSERC research grant which greatly facilitated this study. I am grateful for the countless hours Dr. Capretz has spent reviewing and providing feedback on my work, helping me refine my ideas and improve the quality of my research. His insightful comments and constructive criticism have been invaluable in shaping my thesis and enhancing the overall quality of my research output. Luiz, your mentorship has played a pivotal role in shaping my academic journey, and I am truly honored to have had the opportunity to work under your supervision.

I would also like to extend my gratitude to Afrad Basheer and Eric Howard, whose contributions were integral to this thesis. Their consistent commitment and collaboration throughout the research process have been pivotal to the successful completion of this work. I am particularly thankful to Afrad for his valuable input in co-developing the research design, conducting the experiments, and crafting the conference paper. His expertise and insight were instrumental in shaping our understanding of this complex subject matter. Similarly, Eric has been an exceptional collaborator. His involvement in the research design, and expertise in quantum computing has been crucial in bringing this work to fruition. His sharp intellect and diligence added much value to this research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Quantum computers have garnered significant attention for their ability to perform specific tasks in cryptography, quantum chemistry, and combinatorial optimization that are believed to be infeasible for classical computers. Quantum computers differentiate themselves from classical computers by operating over quantum bits (qubits) in place of classical bits. Qubits are capable of processing information through quantum mechanics, allowing for the solution of certain problems at an exponentially faster rate than classical computers.

Despite the potential for quantum computers to revolutionize the field of computing, the development of fault-tolerant quantum computers remains a critical challenge. Such systems are essential for practical applications but are not yet available. However, the current state-of-the-art quantum computer, IBM Osprey, is a significant achievement, with a remarkable 433 qubits, in contrast to its predecessor, IBM Eagle, which had only 127 qubits.

With the advancement of more powerful quantum computers, there is an expectation that efficient algorithms and solutions that were once thought to be impossible will become attainable, leading to transformative developments in various fields of science and technology. However, there are still significant obstacles to overcome in terms of scalability, stability, and error correction. Consequently, further research is necessary to develop practical applications that can exploit the potential of quantum computing fully.

The progress of the capacity of classical computers over the years have been well predicted by Moore's Law. According to this law, the transistor capacity of classical computers will

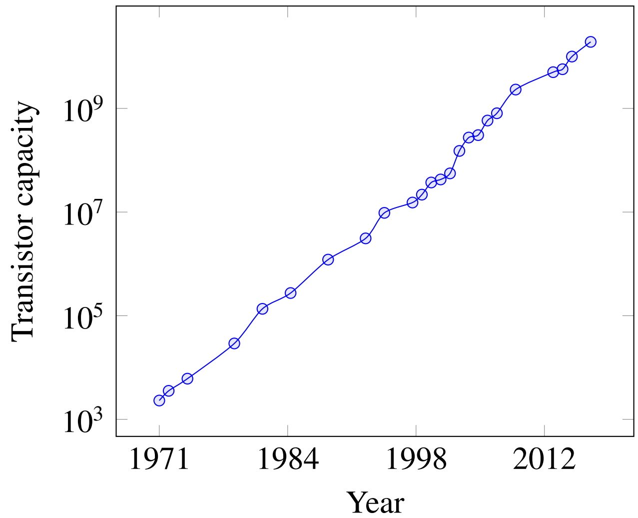

Figure 1.1: Historical transistor capacities.

roughly double every year, giving rise to an exponential growth in the power of the best classical computer over the years. Quite surprisingly, this prediction, made in the mid 60s, have been consistent even until early 2010s. This can be seen in Figure 1.1.

However, with the advent of quantum computers, there is a debate to be had on how we incorporate these machines into the standard Moore's Law framework; are quantum computers devices that are too different from classical computers hence requiring a completely different version of Moores Law to predict the qubit capacities moving forward? Or can we view quantum computers as natural evolution of classical computers and thus somehow predict qubit capacities using extensions of Moores Law for classical computers?

In this work, we try to answer this question in a novel model. We use machine learning to extend and evolve classical Moore's Law into a quantum Moores Law with surprisingly good prediction capabilities. Unlike previous attempts at qubit capacity prediction and quantum Moores law, all of which have used only a handful of data pertaining to quantum computers that we have had until today, we have managed to incorporate the classical transistor data of yesteryears to built a model which is more accurate in predicting the qubit capacites of the last few years. Moreover, we observe that the qubit capacities that IBM predicts in the coming years are much higher than our predictions, which could mean that the growth of quantum computers could be fundamentally shifting from the standard Moore's Law in the coming years.

The general structure of our predictive model is as follows: we first learn a polynomial relationship between number of qubits and number of classical transistors. Then, we train a machine learning model that can predict the transistor capacity of any future year measured as the number of classical transistors capable of fitting on an integrated circuit. This prediction is then fed to the learnt functional relationships to get our prediction of the number of qubits for the year in question, To further demonstrate the power of this model, we also predict the speed of execution of quantum gates in the coming years, by making use of data pertaining to best classical computations per second values that we have achieved over the years.

This thesis is organized as follows: In Chapter 2, we begin with a comprehensive background review that establishes the foundational knowledge for the rest of the paper. We start by introducing the basic concepts of quantum computing, including a technical explanation of how quantum computing operates. We then move on to discuss fault-tolerant quantum computers, explaining the critical role of fault tolerance in realizing practical quantum computing. We continue by reviewing key quantum algorithms, specifically Shor's and Grover's algorithms, which hold a pivotal role in the field of quantum computing. We then examine a variety of quantum computing technologies, including gate-based ion trap processors, gate-based superconducting processors, photonic processors, neutral atom processors, Rydberg atom processors, and quantum annealers. Next, we explore potential applications of quantum computing, highlighting demonstrations of quantum supremacy and discussing the limitations of near-term practical quantum advantage. We also elucidate the significance of qubits and gate speeds, crucial for understanding our subsequent discussions. Lastly, we touch on machine learning, providing context for some of the methods we employ later in the paper.

In Chapter 3, we conduct a review of the related work in estimating quantum computer capacities and recent advancements in quantum algorithms.

In Chapter 4, we explain our approach to predicting qubit capacities, including a thorough explanation of the Elastic Net Regression model and its pessimistic variant. We also detail the process of training, validation, and testing. This Chapter concludes with a Chapter on applying to methodology to gate speeds with a slight modification.

In Chapter 5, we provide a detailed analysis of our findings concerning qubits and gate speeds. We also include a sub-chapter for discussion, where we address pertinent questions such as whether it's fair to assume that qubits will follow the same growth function as the number of transistors, how current hardware limitations and technological advancements may influence the future of quantum computing, and whether it's appropriate to envision a ”Quantum Moore's Law”.

Finally, Chapter 6 concludes paper by summarizing our results and their implications. We follow this with a Chapter on potential future work, suggesting potential avenues for further research in this area.

1.1 Main Contributions of the Thesis

This research introduces a unique methodology that leverages machine learning to evolve the conventional Moore's Law into a quantum analog. Unlike preceding strategies which depended on scarce quantum computer data, this model employs historical data from classical transistor development to forecast qubit capacities. The aim of this thesis is to pioneer a novel technique for predicting future superconducting qubits' capacities and gate speeds utilizing machine learning.

The suggested model is an extension of Moore's Law and its forecasts concerning the transistor capacity of classical computers. Initially, it sets up a polynomial correlation between the number of qubits and the count of classical transistors. Then, a machine learning model is trained to estimate the count of classical transistors for subsequent years. This estimated number is paired with the already defined correlation to calculate the quantity of qubits for a particular year. The identical method is used to analyse data from the highest recorded classical computations per second values to forecast future quantum gate execution speeds.

The outcomes indicate that the presented model surpasses prior techniques in forecasting qubit capacities, hinting at an enhanced method to predict the future capabilities of superconducting qubits and gate speeds based on the relationship between qubit and classical transistor capacities. The model, adopting a data-driven strategy, can integrate new data as quantum milestones are reached.

In this investigation, we introduce an innovative methodology for forecasting the evolution of quantum computing by evolving and expanding classical Moore's Law using machine learning. Our suggested model leverages historical data from classical transistors to better estimate contemporary qubit capacities, demonstrating superior prediction accuracy compared to earlier studies. The proposed model offers crucial insights into the possible direction of quantum computing technology, assuming Moore's Law persists in this field.

Chapter 2

Background

Chapter 2 serves as a foundational review, providing essential background knowledge for the remainder of the paper. It covers various aspects, starting with an introduction to the fundamental concepts of quantum computing, followed by an exploration of fault-tolerant quantum computers and their role in practical quantum computing. Key quantum algorithms, such as Shor's and Grover's algorithms, are discussed, along with an examination of different quantum computing technologies. The chapter also delves into potential applications of quantum computing, including demonstrations of quantum supremacy, while addressing the limitations of near-term practical quantum advantage. The importance of qubits and gate speeds is emphasized, and a brief overview of machine learning is provided to contextualize subsequent discussions in the paper.

2.1 Quantum Computing

A quantum computer is a machine that takes advantage of quantum mechanical properties, using these characteristics through unique hardware. The fundamental unit of data in quantum computing is the quantum bit or qubit, capable of existing in a superposed state of two ”base” states, thereby possessing the ability to inhabit both states at once. This unique feature facilitates the development of quantum algorithms capable of performing calculations at a significantly faster rate than our current computers, as these devices lack the ability to harness the attributes of quantum mechanics. The construction of high-quality qubits has been difficult due to quantum decoherence, a phenomenon that induces disruptions in computations when a qubit is inadequately shielded from its surrounding environment. Despite these hurdles, quantum computers have the potential to efficiently solve many problems that no classical computer could solve in any feasible amount of time — a feat known as ”quantum supremacy”.

2.1.1 Technical Background to Quantum Computing

A pure quantum state is denoted by a vector $|\psi\rangle \in \mathbb{C}^d$ with the constraint that its norm is equal to 1, i.e., $\| |\psi\rangle \| = 1$. The vector $|\psi\rangle$ represents a pure quantum state in a vector space. Here, $|\psi\rangle \in \mathbb{C}^d$ denotes that $|\psi\rangle$ belongs to a complex vector space of dimension d. The double vertical bars $\| |\psi\rangle \|$ signify the norm (magnitude) of the vector. In classical computing, a bit can take on values in the set $\{0, 1\}$, representing logical values. Similarly, in quantum computing, a qubit is capable of adopting any state $|\psi\rangle \in \mathbb{C}^2$. The standard basis vectors in $\mathbb{C}^2$, represented as

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{2.1}$$

correspond to the classical logical states 0 and 1. The notation $\{0, 1\}$ represents a set of possible values for classical bits. Similarly, the vector space $\mathbb{C}^2$ denotes the space of possible quantum states for a qubit. The standard basis vectors $|0\rangle$ and $|1\rangle$ are defined using column vectors, representing quantum states analogous to classical logical values.

The state of a system composed of n qubits is described using state vectors in the tensor product space spanned by $\{|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle \mid \forall\, |\psi_1\rangle, \ldots, |\psi_n\rangle \in \mathbb{C}^2\}$. The tensor product symbol $\otimes$ is used to indicate the combination of multiple quantum states to describe a system of n qubits, so the set notation $|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle \mid \forall\, |\psi_1\rangle, \ldots, |\psi_n\rangle \in \mathbb{C}^2$ represents the space of composite states for an n-qubit system. This space, denoted as $\mathbb{C}^{2^n}$, characterizes an n-qubit quantum system. The notation $\mathbb{C}^{2^n}$ signifies the complex vector space that characterizes an n-qubit quantum system. This vector space encompasses all possible states the system can assume and is of dimension 2^n. Thus, a complete description of such a system requires 2^n complex numbers.

Quantum gates acting on n qubits are defined as unitary operators on $\mathbb{C}^{2^n}$. Common 1 and 2 qubit quantum gates include

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \; Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \; Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{2.2}$$

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \; S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}, \; \text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{2.3}$$

Just as boolean functions can be constructed using elementary classical gates like AND, OR, and NOT, any n-qubit quantum gate can be constructed (approximated) using these 1 and 2 qubit quantum gates.

2.2 Fault-Tolerant Quantum Computers

The pursuit of fault-tolerant quantum computing represents a significant global endeavor to establish viable, market-ready quantum computing systems. This theoretical framework contends with the immense power of existing classical computing. The fundamental premise of fault-tolerant quantum computing is to protect qubits from quantum errors introduced by imperfect control or environmental interferences through the use of Quantum Error Correction (QEC). A crucial aspect lies in the design of quantum circuits, ensuring that both QEC and encoded logic operations are implemented in a manner that prevents errors from propagating throughout the quantum circuits. Once individual qubits attain a satisfactory accuracy level such that QEC corrects more errors than are generated, the likelihood of errors causing a failure in the computation decreases thereby facilitating the implementation of complex quantum algorithms (Paler, [22]).

Error correction strategies partially depend on the use of extra qubits for error prevention, a concept often referred to as redundancy. Essentially, a single logical qubit's information is distributed across multiple physical qubits. In the event of an error in one physical qubit, this can be detected and rectified by comparing it with the other qubits. In simpler terms, the more qubits you use to protect the information in a single logical qubit, the less likely it is that errors will disrupt the computation. This process takes advantage of redundancy and error detection/correction which work best when there are plenty of qubits involved. This allows even imperfect quantum systems (i.e., real-world, physical systems that are subject to errors and decoherence) to successfully run complex calculations.

One method for error correction in quantum systems is outlined by Paler et al. [22] using redundant encoding. Two classical codes are employed separately to detect X- and Z-errors and calculating the parity (evenness or oddness) of two qubits in the code block without directly measuring the qubits themselves. This is achieved using an ancilla qubit. The goal is to generate encoded codewords that always occupy specific states, regardless of the information being encoded. If there are any physical errors, they will disrupt these specific states, which can be detected without revealing any details about the encoded information. The number of errors that can be successfully corrected depends on the number of qubits in the code block. Paler et al. [22] observe that the number of errors which can successfully corrected grows linearly with the number of qubits in the code block. For n qubits, $(n-1)/2$ errors can be uniquely correct. It is important to note that increasing the number of qubits alone does not automatically improve the effectiveness of quantum error correction. The efficiency of error correction depends on various factors, such as the quality of the qubits (including their coherence time and error rates), the interactions between qubits, and the accuracy of operations performed on them. While adding more qubits can enhance error correction capabilities, it also increases the complexity of the system and introduces potential sources of errors.

2.3 Quantum Algorithms

2.3.1 Shor's Algorithm

Conceived by Peter Shor in 1994, Shor's algorithm is a quantum algorithm for effiently factoring large numbers (Shor, [28]). It is one of the most well-known quantum algorithms that demonstrates how quantum computers may solve certain problems exponentially faster than classical computers. The importance of Shor's algorithm emerges from its potential impact on widely used encryption methods, such as RSA, which fundamentally rely on the computational complexity of large number factorization. Consequently, the ability of a quantum computer to factorize such numbers with higher efficiency could compromise the robustness of these encryption mechanisms. The basic steps of Shor's algorithm can be encapsulated as follows:

1. Application of Quantum Fourier Transform (QFT): The initial step in the algorithm necessitates the utilization of the Quantum Fourier Transform on a superposed amalgamation of all conceivable inputs. The QFT, representing a quantum counterpart to the traditional discrete Fourier transform, assists in extrapolating periodicity details from the input.

2. Determining the Period: The subsequent phase involves implementing a modular exponentiation, which empowers the algorithm to uncover the period of a specific function. By incorporating a sequence of such modular exponentiations and gauging the subsequent quantum state, Shor's algorithm can proficiently establish the period.

3. Usage of Continued Fractions: On successful determination of the period, the algorithm resorts to established classical methods, such as the implementation of continued fractions, to discern the factors of the input number relying on the period data.

Shor's algorithm has a polynomial-time complexity for factoring large numbers, which is exponentially faster than the best-known classical factoring algorithms. This efficiency gain arises from the ability of quantum computers to perform parallel computations and exploit quantum interference.

It is important to note that Shor's algorithm is contingent on the availability of a large-scale, fault-tolerant quantum computer to achieve its speedup. Given the existing challenges of realizing a fault-tolerant quantum computer machine, Shor's algorithm has not been practically implemented for factoring large numbers that pose a significant cryptographic threat. Regardless, Shor's algorithm has instigated an influential shift in cryptographic research, prompting significant global attention towards post-quantum cryptography that aims at formulating encryption mechanisms resilient against potential quantum computer attacks.

2.3.2 Grover's Algorithm

Grover's algorithm, conceived by Lov Grover in 1996, is a quantum algorithm that provides a quadratic speedup for searching an unsorted database compared to classical algorithms (Grover, [13]). This method is considered a cornerstone in quantum computing, showcasing the compelling attributes of quantum parallelism in addressing computational issues. The primary problem that the Grover's algorithm seeks to resolve is the unstructured search issue. With a database of N elements, unsorted in nature, the aim is to locate a specific target element within. The classical approach requires an average of $O(N/2)$ queries to locate the target element. However, Grover's algorithm offers a substantial speedup utilizing quantum parallelism. An overview of Grover's algorithm can be summarized as follows:

1. Initiation: The process commences by setting up the quantum computer to generate an equal superposition of all feasible states. This is achieved by employing a Hadamard transform on a set of qubits.

2. Oracle Mechanism: A quantum gate, signifying the oracle, is devised to mark the target

element within the superposition. It inverts the phase of the target element, effectively setting it apart from the remainder of the database.

3. Amplitude Amplification: The crux of Grover's algorithm lies in the repeated application of two operations: the oracle and the inversion about the average. The oracle amplifies the amplitude of the target element, while the inversion operation amplifies the amplitude of all other elements. Through iterative application of these operations, the likelihood of measuring the target element gradually enhances.

4. Quantification: Ultimately, a measurement is enacted on the qubits, leading to the collapse of the superposition and thereby deriving the target element with a high probability.

An important aspect of Grover's algorithm is that it achieves a quadratic speedup compared to classical algorithms. While classical algorithms require $O(N/2)$ queries, Grover's algorithm requires only approximately $O(\sqrt{N})$ queries to find the target element with high probability. This quadratic speedup can provide significant computational advantages for large-scale search problems.

It is important to note that Grover's algorithm does not provide an exponential speedup like Shor's algorithm for factorization. Instead, it showcases the power of quantum parallelism and amplitude amplification in solving search problems more efficiently than classical algorithms. Grover's algorithm has applications in various domains, including database search, optimization, and cryptography. It demonstrates the potential of quantum computing to revolutionize search algorithms and inspire the development of new quantum algorithms for solving a range of problems more efficiently.

2.4 Gate-Based Superconducting Quantum Computers

There are several different types of quantum computers that have been proposed and developed, each with its own approach to realizing and manipulating quantum bits (qubits) and performing quantum operations. This thesis will focus on gate-based superconducting quantum computers.

Superconducting quantum computing is another approach to gate-based quantum computing that utilizes superconducting electronic circuits. Superconductivity is observed in certain materials at extremely low temperatures, exhibiting zero electrical resistance and expelling magnetic flux fields below a critical temperature. Superconducting qubits are built with superconducting circuits that operate at cryogenic temperatures, enabling the manipulation of quantum information. This work focuses on qubits and gate speeds in superconducting processors.

2.5 Applications of Quantum Computing

There are a wide range of potential applications of quantum computing, due to it's ability to process massive amounts of data and perform computations that have, until now, been intractable.

1. Cryptanalysis and Cryptography: Existing encryption standards risk being deciphered by the formidable processing capabilities of quantum computers. For instance, Shor's algorithm can factorize large numbers with greater efficacy compared to classical computing devices, thereby posing a threat to the integrity of RSA encryption. Conversely, quantum computing also harbors the potential to spur the evolution of novel, robust cryptographic systems, inclusive of quantum key distribution mechanisms.

2. Pharmacological Innovations: Quantum computational systems may hold the key to modeling intricate molecular interactions at an atomic level, thus expediting the processes of drug discovery and formulation. These advanced systems could offer valuable assistance to scientists in devising new pharmaceuticals or understanding the characteristics of complex biological systems by providing accurate simulations and analysis of their molecular constituents.

3. Quantum Physics: Quantum computers could pave the way for breakthroughs in our understanding of the cosmos through the simulation of quantum systems, an area where classical computers grapple with significant difficulties. Quantum simulations facilitated by such computational power could unveil novel insights into the fundamental nature of the universe.

2.5.1 Demonstrations of Quantum Supremacy

The first computation that could only be accomplished on a quantum processor was marked by an experiment conducted by Arute et al. (Arute, [5]) using a programmable superconducting processor, surpassing the capabilities of classical computing and challenging the extended Church-Turing thesis. By implementing random quantum circuit sampling, the experiment demonstrated the ability to perform a task for which no efficient classical method exists. The researchers employed a programmable superconducting qubit processor, specifically utilizing 53 qubits to create quantum states and explore a computational state-space of dimension 253 (approximately 10^{16}). The resulting probability distribution was sampled through repeated experiments, which were verified using classical simulations. Notably, the Sycamore processor took approximately 200 seconds to sample one instance of a quantum circuit a million times, a task that would hypothetically require around 10,000 years for a state-of-the-art classical supercomputer to complete according to current benchmarks.

Another significant advancement toward practical quantum computing, was marked by Madsen et al. [19], who unveiled the remarkable capabilities of a photonic processor called Borealis. Their study demonstrated the execution of Gaussian boson sampling on 216 squeezed modes, which were intricately entangled with three-dimensional connectivity. The outcomes revealed a staggering contrast between Borealis and conventional supercomputers. While the best available algorithms and supercomputers would require more than 9,000 years to generate a single sample from the programmed distribution, Borealis accomplished the task in a mere 36 microseconds. This exceptional runtime advantage surpassed the previous achievements of photonic machines by a factor of over 50 million. Consequently, this research is widely acknowledged as a pivotal milestone, affirming the viability of photonics as a foundational platform for practical quantum computing and validating its crucial technological attributes.

In a study titled ”Classically Simulating Quantum Supremacy IQP Circuits through a Random Graph Approach” by Codsi [9], novel methodologies were introduced to enhance the classical simulation of random IQP circuits. The research emphasized the need for cautious evaluation when claiming quantum supremacy, as advancements in classical algorithms can rapidly achieve substantial improvements. Furthermore, the paper presented an algorithm capable of computing the amplitudes of IQP circuits ranging from 30 to 50 qubits on a single CPU core of a laptop in a matter of minutes. The algorithm could be readily distributed in parallel, suggesting that a cluster comprising 100,000 CPU cores could calculate the amplitude of a dense 60-qubit circuit in approximately one hour. Additionally, the study implied that the computation of 70-qubit circuits could be within the grasp of the world’s leading supercomputers.

2.5.2 Limitations of Near-Term Practical Quantum Advantage

In their publication titled ”Disentangling Quantum Computing: Realistic Applications and Speedup Criteria,” Hoefler et al. [15] examine potential applications of quantum computers and the factors that determine their feasibility. The authors emphasize the significance of considering not only asymptotic speedups but also the constants associated with quantum computations. To illustrate this, they conduct a comparative analysis between an idealized quantum computer and a classical computer chip, taking into account various factors such as I/O bandwidth, crossover scale, and compute performance. Their investigation underscores the challenges arising from the interplay between quantum and classical systems, particularly concerning data input and output. As a result, they propose that quantum computers may prove more practical for ”big compute” problems involving small datasets rather than those involving big data. The concept of crossover time is introduced, referring to the point at which the quantum speedup compensates for the slower operations of a quantum computer compared to a classical counterpart. The authors stress the importance of achieving short crossover times to enable practical applications. Concrete examples and estimations are provided, outlining the function complexity required to attain crossover times within specified durations. Importantly, the authors conclude that quadratic speedups are inadequate for achieving practical quantum advantage, highlighting the necessity of at least cubic or quartic speedups. They identify certain applications, such as quantum system simulation, cryptanalysis utilizing Shor's algorithm, and solving highly structured linear systems of equations, as the most promising candidates for exponential quantum speedups and practical quantum advantage. Conversely, applications with quadratic quantum speedups, including machine learning, drug design, protein folding, and certain scientific simulations, are deemed unlikely to achieve quantum advantage in the near future. The limitations of quantum computing for big data problems, unstructured linear systems, and database search are attributed to I/O constraints and the nature of black-box algorithms. In light of these findings, the authors call for a focus on super-quadratic or exponential speedups and advise the consideration of I/O bottlenecks when developing quantum algorithms.

2.6 Significance of Qubits and Gate Speeds on Quantum Capacity

The capacity of a quantum computer is one factor in determining its computational abilities and its ability to tackle complex problems. It often encompasses the number of qubits, gate speeds, error rates, coherence times, and other performance metrics that collectively attribute towards the processing power and efficiency of a quantum computing system [7]. Beyond the mere quantity of qubits, the practicality of a quantum computer also encompasses quantum error correction as a critical element in the pursuit of fault-tolerant quantum computing systems. Schemes like surface codes enable the conversion of multiple physical qubits into a single low-error 'logical qubit.' Consequently, a larger number of physical qubits expands the potential for creating a greater quantity of logical qubits, thereby facilitating the execution of more intricate quantum computing tasks.

The operational speed of quantum gates serves as a pivotal factor in assessing the practicality of a quantum computer. These gates, acting as fundamental building blocks of quantum circuits, enable the execution of quantum algorithms. A swifter gate speed translates to faster execution of quantum circuits, resulting in reduced overall computation time. This attribute proves invaluable, particularly considering the often extensive sequences of gates required for many quantum algorithms. Additionally, faster gate operations contribute to a reduction in the accumulation of computation errors.

The interplay between gate speed and coherence time, which measures a qubit's ability to maintain its quantum state, holds significant importance. A faster gate operation allows for a greater number of computations to be performed within the coherence time of a qubit. This, in turn, mitigates the detrimental effects of decoherence on the outcome of the quantum computation. The synergistic relationship between gate speed and coherence time thus plays a crucial role in enhancing the reliability and efficiency of quantum computations.

2.7 Introduction to Machine Learning

Lying in the intersection of computer science, mathematics and statistics, machine learning is a field that has transformed sectors ranging from communication technology (Huang, [16]), astrophysics (Vanderplas, [32]), finance (Dixon, [10]) to language (Chowdhary, [8]) and proofs (Sanchez, [26]). It is the science of learning patterns and functional relationships from known information. Machine learning can be broadly categorized into supervised learning, unsupervised learning and reinforcement learning.

In supervised learning, the known information will contain many input samples and their corresponding outputs or labels. Using a dataset, the goal is to learn a model that can predict the outputs of similar input samples which we have not come across yet. Some examples are spam filters (Renuka, [25]), house prices prediction (Phan, [24]), fraud detection (Khatri, [17]). Popular algorithms used to learn supervised learning models include neural networks, support vector machines, linear and logistic regression, etc.

Similarly, consider a problem where one has to cluster a given set of data points (vectors) into different groups. Unlike supervised learning, in this case, we do not have labelled data. So, the model is not shown any ”known outputs” and has to figure things out by learning on unearthing intrinsic properties of the data points. Such problems fall in the unsupervised learning category. Other prominent examples in this class include dimensionality reduction (Tschannen, [30]) and learning low dimensional vector embeddings for words (Pennington, [23]), general entities (Wu, [33]) and elements of knowledge graphs (Nickel, [21]).

Finally, in reinforcement learning, we frame the learning procedure in a reward based framework, similar to how animals are taught to do certain specific actions. The model to be trained is designed as an agent performing actions in specially constructed environments. Such a model is rewarded for good actions and is trained to maximize rewards. Famous instances of reinforcement learning in action includes AI programs such as AlphaGo Zero (Silver, [29]) and AlphaStar (Arulkumaran, [4]), automated neural network architecture designer AutoML (He, [14]), etc.

Chapter 3

Related Work

The goal to unlock the potential of quantum computing has witnessed remarkable progress in recent years. This chapter delves into the exciting advancements that have propelled the field forward, improving our understanding of the capabilities and limitations of quantum computers. This chapter explores two key areas of advancement: estimating quantum computer capacities and breakthroughs in quantum algorithms.

3.1 Estimating Quantum Computer Capacities

Agarwal et al. [3] presented a series of estimations regarding various quantum computing metrics, such as number of qubits, gate frequency, gate infidelity, and overhead reduction. These predictions were founded upon assumptions made by the authors, without the use of statistical techniques. The study suggests an optimistic prediction of a doubling in the number of qubits every 10 months, and a pessimistic prediction of a doubling every 20 months. Furthermore, the authors predicted that the gate infidelity would follow DeVincenzo's law of reducing infidelity by a factor of 2 per year, plateauing at an infidelity of 5×10^{-6} in the optimistic case, and 5×10^{-5} in the pessimistic case. In their study, the authors focus on two main areas of Bitcoin that could potentially be at risk due to quantum computers. Drawing from their projections, they arrive at two primary conclusions.

1. Proof of Work (PoW): The PoW used by Bitcoin is relatively resistant to substantial speedup by quantum computers in the next 10 years. This is mainly because specialized ASIC miners are extremely fast compared to the estimated clock speed of near-term quantum computers. Using Grover's algorithm, a quantum computer can perform the hashcash PoW by performing quadratically fewer hashes than a classical computer. However, the extreme speed of current specialized ASIC hardware, coupled with much slower projected gate speeds for current quantum architectures, essentially negates this quadratic speedup at the current difficulty level, giving quantum computers no advantage. Future improvements to quantum technology allowing gate speeds up to 100GHz could allow quantum computers to solve the PoW about 100 times faster than current technology, but such a development is unlikely in the next decade.

2. Cryptographic Signatures: The elliptic curve signature scheme used by Bitcoin is at significant risk from quantum computers. The authors estimate that a quantum computer capable of breaking the elliptic curve signature scheme could exist as early as 2027. The primary window for this attack is from the time a transaction is broadcast until it's processed into a block on the blockchain with several blocks after it. By their most optimistic estimates, this period could be less than 10 minutes, the block time used in Bitcoin.

In their study, Sevilla et al. [27] have offered a novel statistical model purposed to predict both the number of qubits and the average two-qubit error rate in quantum computing systems. An essential cornerstone of their study revolves around the anticipation of when the first large-scale, fault-tolerant quantum computer will be capable of breaking the modern cryptographic scheme RSA 2048. This particular milestone is operationalized in their work as a function of the number of generalized logical qubits, with the threshold set specifically at 4100 logical qubits. Sevilla and his team employed a sophisticated multivariate log linear regression model as a tool to thoroughly examine the association between the number of physical qubits in a system and the error rate it experiences. Interestingly, their analysis revealed a positive correlation between these two metrics, indicating that an increase in qubits tends to be associated with higher error rates. This finding suggests the existence of a development frontier in the field of quantum computing where trade-offs are made between the number of qubits and the error rate. The researchers also built their predictions on the assumption of exponential progress in quantum computing. Using this assumption as a foundation, they applied a log-linear multivariate model to estimate an upper bound of the likely progress trajectory of QC, specifically focusing on technologies based on superconductors. In addition to these statistical models, Sevilla et al. [27] made a significant contribution to the field by compiling and providing a comprehensive dataset. This dataset, carefully curated, records a multitude of data points related to the progression of quantum computing, covering aspects such as qubit counts and error rates over time.

In our work, we have chosen to utilize this dataset as a key input to our analyses. We have conducted a direct comparison between the methodology proposed by Sevilla et al. [27] and our own approach. This comparison, detailed in our results, serves to further validate our findings while also providing a basis for discussion on the relative merits and potential improvements in both methodologies.

IBM has published a roadmap delineating their accomplishments in the field of quantum computing, as well as their projections for forthcoming years [1]. The roadmap includes annual estimates for the number of qubits that will be available during each respective year. Similarly, Google has proposed a roadmap conceptualizing its own progression in quantum computing. Google's strategy is delineated through a logarithmic scale "journey", wherein the number of qubits in their quantum systems is envisaged to incrementally increase over time[2]. This offers an insight into Google's strategy for scaling their quantum computing capacities.

In our study, we leveraged IBM's executed quantum roadmap to facilitate a comparison between our results and those proposed by Sevilla et al. [27]. This approach allowed for an evaluation of the effectiveness and accuracy of our methodology, which was able to provide

[1]For more information, visit: `https://www.ibm.com/quantum/roadmap`.

[2]See Google's quantum computing journey here: `https://quantumai.google/learn/map`.

reliable estimations regarding the number of qubits and the error rate in quantum computing. By utilizing these industry roadmaps and predictions, we were able to validate our methodology and further enhance the credibility of our findings. This validation process served two purposes. First, it allowed for an examination of our predictions' accuracy. Second, we were able to substantiate the reliability of our research outcomes by comparing our forecasts to the forecasts proposed by industry experts actively engaged in the development of quantum technologies. Furthermore, our research underscores the significance of industry roadmaps in harmonizing academic research with the pragmatic pace of technological development in the real world, thereby ensuring the relevance and applicability of our findings. The methodology and results presented in this thesis reinforce the principle that academic research in quantum computing should be attuned to the industry's progression. This alignment, facilitated by the utilization of industry roadmaps and forecasts, ensures that the academic contributions remain relevant to the evolving needs of industry.

Since our study, Microsoft's Azure Quantum Resource Estimator [20] has been updated to provide users with rQOPS and error rate outputs for their chosen quantum algorithms and hardware architectures. The rQOPS metric quantifies reliable operations in a practical quantum algorithm for scaling up quantum systems to execute valuable applications. It encapsulates three critical factors: scale, ensuring the presence of a sufficient number of reliable qubits; speed, dependent on the clock speed; and reliability, indicated by the error rate on logical qubits. The rQOPS value is obtained by multiplying the number of logical qubits by the hardware's logical clock speed. The metric incorporates the logical error rate, which denotes the maximum acceptable error rate for operations performed on logical qubits. This is relevant to this work since the number of qubits and gate speeds can be used predicted from our resulting model and used as input to the Quantum Resource Estimator, enabling researchers to assess the feasibility of quantum algorithms and hardware architectures for specific applications.

3.2 Advancements in Quantum Algorithms

Gidney et al. [11] presents significant advancements in the reduction of the cost of factoring integers and computing discrete logarithms in finite fields on a quantum computer. The researchers achieved this by integrating techniques from numerous previous works, including those by Shor, Griffiths-Niu, Zalka, Fowler, Ekerå-Håstad, Ekerå, Gidney-Fowler, and Gidney. The study's computation cost estimates were based on assumptions about large-scale superconducting qubit platforms, including a planar grid of qubits with nearest-neighbor connectivity, a physical gate error rate of 10^{-3}, a surface code cycle time of 1 microsecond, and a reaction time of 10 microseconds. The estimates also took into account factors usually overlooked such as noise, the necessity for repeated attempts, and the spacetime layout of the computation. Compared to prior works, the construction of this study demonstrated a hundredfold decrease in spacetime volume when factoring 2048 bit RSA integers. In the abstract circuit model, ignoring overheads from distillation, routing, and error correction, the construction uses an equation involving logical qubits, Toffoli gates, and measurement depth to factor n-bit RSA integers. Finally, the study discusses the cryptographic implications of their findings, for both RSA and schemes based on the discrete logarithm problem (DLP) in finite fields.

Gouzien et al. [12] presents a study on the use of cat qubits as building blocks for quantum computing. Cat qubits exhibit a tunable noise bias, allowing for exponential suppression of bit-flips based on the average photon number. To protect against phase errors, the study uses a simple repetition code. The research evaluates the cost of such a repetition code and provides insights for selecting a large-scale architecture using cat qubits. The researchers perform a performance analysis using Shor's algorithm to compute discrete logarithms on an elliptic curve. They propose a 2D grid of cat qubits with neighboring connectivity to implement two-qubit gates through lattice surgery and Toffoli gates via offline, fault-tolerant preparation of magic states. These methods involve projective measurements and subsequent gate teleportations. All-to-all connectivity between logical qubits is achieved by routing qubits. The study assumes a ratio between single-photon and two-photon losses of 10^{-5}, along with a cycle time of 500 nanoseconds. With these assumptions, the researchers demonstrate that the proposed architecture can compute a 256-bit elliptic curve logarithm in nine hours using 126133 cat qubits.

Chapter 4

Research Methodology

This chapter provides a comprehensive examination of our approach to predicting qubit capacities, presenting an account of our Elastic Net Regression model and its variant with a pessimistic perspective along with our training, validation, and testing processes. We then extend our focus to the prediction of gate speeds, incorporating a slight modification to the methodology.

4.1 Design Rationale

The design of the methodology is grounded in the goal of predicting the future development of superconducting qubits and gate speeds by leveraging the well-known trend established by Moore's Law. This approach is chosen to provide insights into the potential trajectory of quantum computing capabilities and gate speeds if they were to follow a similar pattern of advancement observed in classical computing. By building 'g' to capture the relation between classical transistors and quantum computing capabilities, it serves as a bridge between the two domains allowing us to to draw parallels between the historical growth of classical computing components and the potential growth of quantum computing. After establishing relationships between classical computer metrics and quantum computing metrics, we may build a model 'f' to predict the future number of classical computing metrics observed under Moore's Law to then bridge back to the quantum computing space, thereby ensuring that predictions in the superconducting quantum computing space is driven by the same trends observed under Moore's Law. An overview of the design architecture for predicting superconducting qubits is provided in Figure 4.1.

4.2 Predicting Qubit Capacities

This chapter details the steps and processes involved in our methodology to forecast the number of qubits in superconducting quantum computers. For a bird's eye view of the method, refer to Figure 4.4 which presents a schematic representation of the process.

Our training phase relies on the combination of two distinct datasets that serve to inform our model. The initial dataset, depicted in Figure 1.1 and sourced from Our World in Data [1], illustrates the evolution of transistor capacities over time. Comprising 24 data points, each sample encompasses two attributes: 'Year' and 'Transistors per microprocessor'. The graph is characterized by an exponential curve, a clear demonstration of Moore's Law in action. Moore's Law, an observation made by Gordon Moore in 1965, posits that the number of transistors on a microchip doubles approximately every two years, while the cost of these computers is halved. The empirical evidence presented in our graph substantiates this theory and provides a strong foundation for our model.

We use a supplementary dataset spotlighting the achievements in superconducting qubit capacities in recent times. This dataset, presented graphically in Figure 4.2, is sourced from Sevilla et al. [27]. Comprising 9 instances, each sample is defined by two variables: 'Year' and 'Superconducting qubit capacity'.Despite its relatively smaller size, this dataset provides crucial insights into the advancements in superconducting quantum computing capacities. The dataset outlines the peak superconducting qubit capacities achieved over the years, beginning from 2007 and extending up until 2021. It provides a clear and succinct overview of the growth and development in this space, effectively capturing the trajectory of progress over time.

In reviewing the available datasets, historical records from 2013 mark a significant milestone: the establishment of a 2-qubit capability in quantum computers. Concurrently, approximately 10^{10} transistors existed in classical computers. Similar data is available for 2014, making both years valuable for comparing classical bits and quantum bits. With this overlapping data, we aim to determine the polynomial relationship, g, which links the number of qubits to the corresponding number of classical bits. Once established, this relationship can predict the number of qubits based on the number of classical transistors, and vice versa, offering insights into the comparative capabilities of classical and quantum computing systems.

After determining the polynomial relationship, the next phase involves training a machine learning model, f. We use the classical dataset, which includes the number of transistors in classical computers over various years, to train f to project future transistor numbers. By utilizing data on the actual growth of classical transistors, f can extrapolate observed patterns and make predictions for any given year. Thus, f offers insights into the projected growth and development of classical computing capacities under the observed pattern of Moore's Law.

In the following stage, we employ the polynomial relationship g to convert predicted numbers of classical bits to quantum bits. For any given year y, we can determine the projected qubit capacity generated by f by evaluating the composition $g(f(y))$.

We further refine our predictions using hyperparameter tuning to find the optimal values for the selected machine learning model. Given the limited data available on superconducting qubits, our approach for validating our model is to use only the last three years of data as our validation dataset 2017, 2018, and 2021. During these years, we compare the performance of our model with other models in the literature. To learn the relationship between qubit and classical transistors, we use data from years 2013 and 2014, and then tune the hyperparameters of our model based on its performance in predicting qubit numbers for 2013, 2014, and 2015. To optimize the hyperparameters, we use the Mean Squared Error (MSE) between the model predictions and the true qubit numbers. Ultimately, we select the final models based on the hyperparameters that result in the lowest MSE scores.

One crucial aspect is the nature of the datasets. Both classical and quantum datasets provide measurements for the maximum number of transistors and qubit capacities at the end of each year. This temporal factor is crucial, as it allows our model's predictions to be interpreted as the projected qubit capacity by year-end. This granularity may inform decision-making and planning, as stakeholders can anticipate advancements in quantum computing capabilities over time.

4.2.1 Elastic Net Regression

The Elastic Net regression model was selected for this study due to its strengths in handling multicollinearity and regularization, which are particularly advantageous when dealing with our dataset. Our data exhibits high multicollinearity due to our assumption of the close relationship between the number of classical transistors and quantum computing capabilities over the years. Elastic Net regression effectively manages this issue by combining the Ridge and Lasso regression models, which, respectively, minimize the sum of square residuals and absolute values of coefficients. Furthermore, Elastic Net introduces an element of regularization which helps to prevent overfitting. This is especially important for our project, as we aim to develop a model that not only performs well with our existing data but also can generalize to future scenarios.

The model can be formally defined as follows. Let $\mathcal{D}$ be a set of p dimensional real vectors and let $g : \mathcal{D} \to \mathbb{R}$. Our goal is to find g, given a finite set of training data $\{(x^{(k)}, g(x^{(k)})) \mid k = 1, 2, \dots, m\}$. In linear regression we assume that g can be approximated by a hyperplane. That is, there exist a vector $\beta_0 \in \mathbb{R}^p$ and scalar $\beta_1 \in \mathbb{R}$ such that

$$g(x) \approx \beta_0^T x + \beta_1. \tag{4.1}$$

Once this parametric model has been decided, the aim then is to find the vector and scalar β_0 and β_1. Given any candidates $\alpha_0 \in \mathbb{R}^p$ and $\alpha_1 \in \mathbb{R}$, we can compute a score of how bad the choice is as follows:

$$C(\alpha_0, \alpha_1) = \sum_{i=1}^{m} (g(x^{(i)}) - \alpha_0^T x^{(i)} - \alpha_1)^2. \tag{4.2}$$

The cost function C simply computes how far the values predicted by the hyperplane is to the true values for each known input $x^{(i)}$. This can be more concisely written as

$$C(\alpha) = \|X\alpha - y\|_2^2, \tag{4.3}$$

where $X \in \mathbb{R}^{m \times p+1}$ with its k^{th} row being $x^{(k)}$ appended by an extra 1, $\alpha = [\alpha_0\ \alpha_1]$ and $y \in \mathbb{R}^m$ with its k^{th} entry being $g(x^{(k)})$. Our aim then is to find

$$\beta^* = \operatorname*{argmin}_{\beta} C(\beta). \tag{4.4}$$

The set $\mathcal{D}$ is composed of vectors, essentially ordered lists of numbers, with each vector having p elements. The function g is the relationship we're trying to find, which takes a vector from the set $\mathcal{D}$ and gives us a real number. The training data we're using to find this relationship is a set of pairs. Each pair consists of a vector $x^{(k)}$ and the corresponding output $g(x^{(k)})$ of our desired function. The goal is to determine what the function g is based on these pairs. In the context of linear regression, we're assuming that g can be approximated by a flat plane (a hyperplane in higher dimensions). This is described by the equation $g(x) \approx \beta_0^T x + \beta_1$, where β_0 is a vector and β_1 is a scalar (a single number), and the ^{T} represents the transpose of the vector (flipping the vector from vertical to horizontal or vice versa).

The next step is to find the best choices for β_0 and β_1 that make the approximation as good as possible. We measure the quality of these choices using a cost function C. The cost function measures how far the hyperplane's predicted values are from the actual values for each input vector. It does this by summing the squared differences between the predicted and actual values. The ultimate goal is to find the values of β_0 and β_1 that make C as small as possible (thus giving us the best approximation for g). This is represented by the equation $\beta^* = \underset{\beta}{\operatorname{argmin}}\, C(\beta)$, which basically says "find the β that minimizes the cost function C".

We can find the best approximation for our function using various methods. One of these is iterative optimization algorithms. However, in many instances, we can also find the best approximation directly without iterating. An issue that can arise when we minimize the cost function too much is overfitting. Overfitting is when our model becomes too good at predicting the outcomes for our training data (the data we used to build the model) but struggles to predict outcomes for new, unseen data. In other words, it has learned the training data so well that it performs poorly when introduced to new information. To avoid overfitting, we use a technique called regularization. Regularization limits the range of values that our coefficients (in this case represented by α) can take. We can think of it as a tuning process to keep our model's predictions reliable. We add this regularization as a penalty term to our cost function, leading to a new cost function:

$$C(\alpha) = ||X\alpha - y||_2^2 + \lambda||\alpha||, \tag{4.5}$$

Here, the symbol $||.||$ represents a type of measurement, known as a vector norm, and λ is a hyperparameter we set. If we choose a large value for λ, the regularization becomes more strict, limiting the range of values that our coefficients can take. The choice of vector norm can lead to different types of regression models. If we choose the Euclidean norm, also known as the $L2$ norm, we call the model Ridge regression. If we choose the taxicab norm, or $L1$ norm, we call the model Lasso regression. Elastic Net regression is a type of regression that combines the constraints of both Ridge and Lasso regression models. It uses both $L1$ and $L2$ norms, and we adjust their influence using two hyperparameters, λ_1 and λ_2. The cost function for Elastic Net regression is:

$$C(\beta) = ||X\beta - y||_2^2 + \lambda_1||\beta||_1 + \lambda_2||\beta||_2^2. \tag{4.6}$$

In this equation, λ_1 and λ_2 are the hyperparameters we need to adjust to get the best model.

Pessimistic model

In order to construct a more conservative or "pessimistic" forecast of quantum capabilities compared to classical transistor capacities, we adjust the nature of the relationship that our model learns between these two variables. Instead of directly learning a polynomial relationship, we introduce a logarithmic transformation, transitioning to a polylogarithmic relationship.

We may clarify this with the following notation. Suppose $n_1, n_2, \ldots, n_T$ denote the transistor capacities of years $y_1, y_2, \ldots, y_T$ respectively, and $q_1, q_2, \ldots, q_T$ represent the corresponding qubit capacities. Under our new approach, the polynomial function g is tailored to fit the logarithm of the transistor capacities to the qubit capacities, i.e., $g(\log n_i) \approx q_i$.

This logarithmic adjustment can be seen as 'slowing down' the increase in qubit capabilities in relation to classical transistor capacities. Essentially, as the transistor capacity grows exponentially over time (as suggested by Moore's Law), the logarithmic transformation moderates this growth, leading to a smaller corresponding increase in the predicted qubit capacity. As a result, the polynomial function g provides a more conservative estimate, dampening the relative value of qubits compared to classical transistors over time.

This change can be valuable in different scenarios. For instance, it might reflect a perspective where advancements in quantum computing face increasing technical challenges and do not keep pace with the exponential growth observed in classical computing capacities. This approach can be useful for stakeholders who prefer to err on the side of caution when predicting future quantum computing capabilities based on the ongoing expansion of classical computing capacities.

4.2.2 Training, Validation and Testing

Figure 4.3 provides a visual representation of how our dataset was partitioned for training, testing, and validation purposes. The division is an important part of the model-building process, allowing us to properly train our model, tune its parameters, and evaluate its performance.

The Elastic Net regression model, which is represented by the black line in figure 4.3, is trained on the transistors data. Meanwhile, the polynomial relationship between transistor capacity and qubit capacities are learned from data corresponding to the years 2013 and 2014. This choice of years is significant, as it represents the overlap in the availability of data for both classical and quantum computer capabilities. Once the initial training phase is complete, we move on to hyperparameter tuning. Hyperparameter tuning based on prediction performance on data from the year of 2015. In future years, more data points may be added to the training and validation sets to improve robustness. Importantly, to ensure a fair evaluation of our model's predictive performance, we deliberately withhold data from the most recent four years, treating these as our test set. This approach ensures that our model is evaluated on unseen data, offering a more realistic estimation of its real-world predictive performance.

Figure 4.4 offers an overview of our model's prediction mechanism. The shaded colour regions are from Figure 4.3, indicating training, validation and test sets. Firstly, using the qubit and classical transistor data from 2013 and 2014, we learn a polynomial function g that approximates the relationship between transistor capacity and qubit capacity. In other words, g(transistor capacity) $\approx$ qubit capacity. Next, we train the Elastic Net regression model, denoted as f, using the classical transistor data. This model is designed to predict future transistor capacities based on a given year. Formally, for any year y, $f(y) \approx$ transistor capacity in year y. Once both functions f and g are learned, predicting the qubit capacity for a given year becomes straightforward. Given a year y, we first use f to predict the transistor capacity for that year, and then pass this predicted transistor capacity into function g. The output, $g(f(y))$, provides our predicted qubit capacity for year y. This two-step process effectively allows us to project future quantum computing capabilities based on trends observed in classical computing.

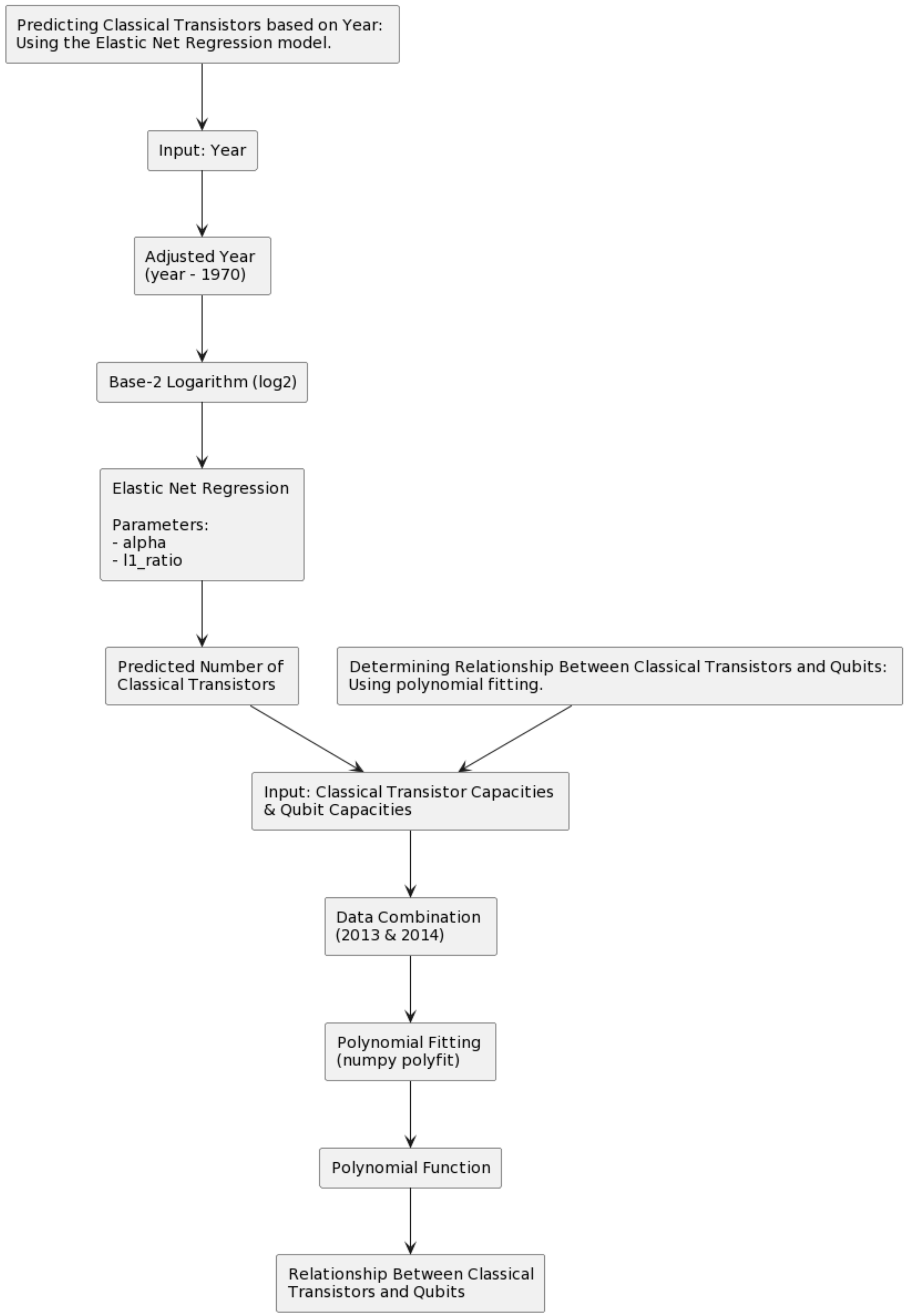

Figure 4.1: Architecture for predicting superconducting qubit capacities.

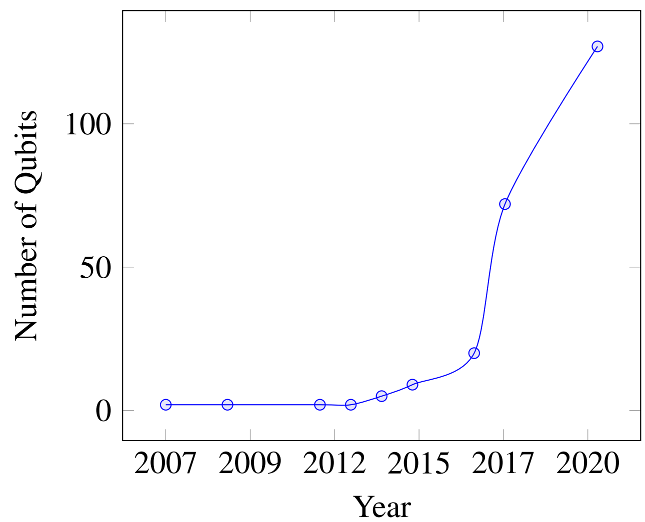

Figure 4.2: Historical superconducting qubit capacities.

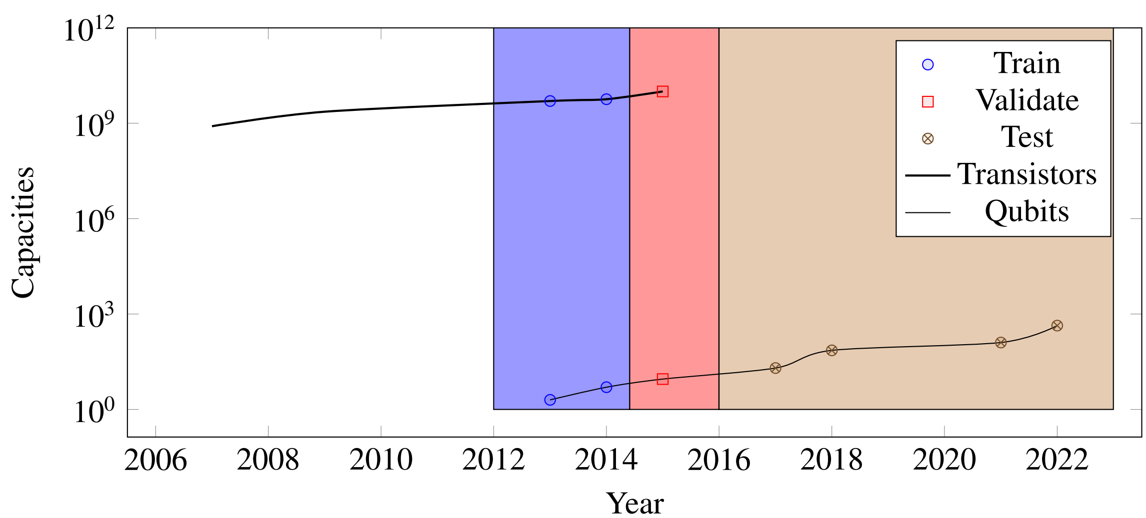

Figure 4.3: Partitioning of data into train, validate and test sets.

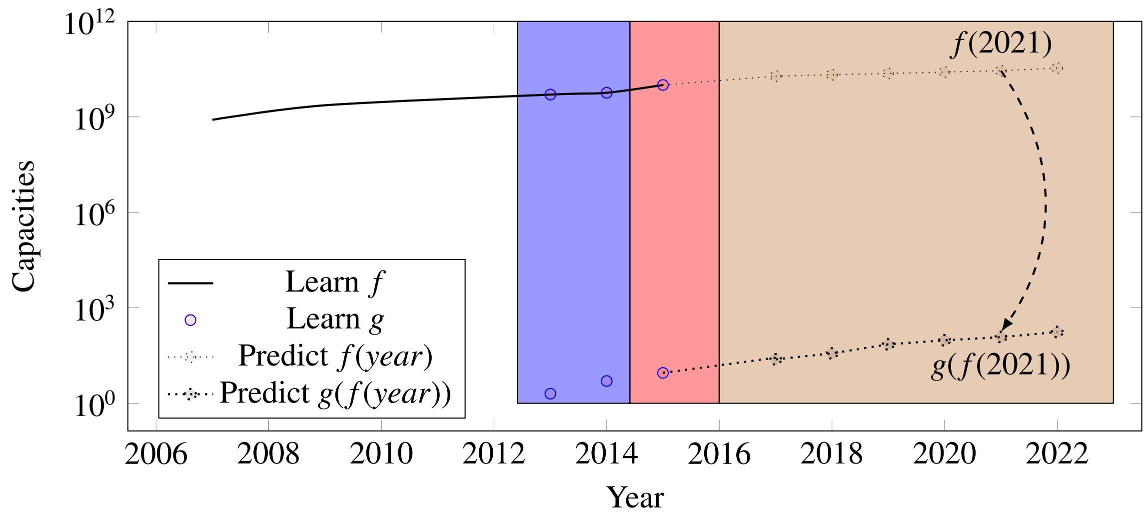

Figure 4.4: Overview of the methodology process.

4.3 Gate Speeds

Quantum gates typically operate at a slower speed than their classical counterparts. However, our goal in this chapter is to predict future advancements in quantum gate implementation speed, leveraging a model similar to the one used for predicting qubit capacities.

To this end, we incorporate two key datasets. The first, derived from the data collected by Koomey et al. [18], offers a temporal overview of operation speeds for classical computers, illustrated in Figure 4.5. This dataset comprises of two variables: 'Year' and 'Computations per second'. The second dataset, studied in Aggarwal et al.'s work [3], provides corresponding information for quantum gate speeds, displayed in Table 4.1. This dataset also comprises of two variables: 'Year' and 'Quantum operations per second'.

A critical point to note is the absence of overlapping years in these two datasets, meaning there are no years for which we possess data for both quantum and classical gates. To overcome this challenge, we adopt a unique approach. First, we use the classical gate speed data to train our model and predict the classical gate speeds for specific years for which we have quantum gate speed data. These predicted values form an artificial overlap between the two datasets.

Specifically, we predicted values for the years of 2013, 2015 and 2016 using a regression model similar to the one used in Chapter 4.2.1 and then use these predicted scores for learning the relationship between qubit and classical bit. This is shown in Figure 4.5, where the classical computations per second data is extended to the years of 2013, 2015 and 2016 in order to create an overlap with the quantum operations per second data provided in Table 4.1. Having established this artificial overlap, we then proceed to learn the relationship between quantum gate speeds and classical gate speeds. By leveraging the model trained on classical data and applying it to the quantum context, we aim to derive meaningful insights into future developments in quantum gate speed. This approach allows us to extrapolate the pace of progress in quantum computing, despite the lack of directly comparable historical data. To refine the performance of our regression model, we used the final three years of the quantum dataset for hyperparameter tuning. We chose to compute the relative squared error (RSE) as the function

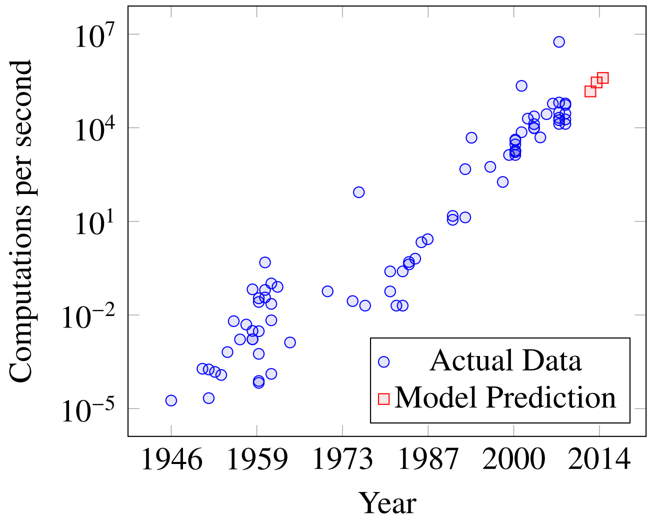

Figure 4.5: Historical trend of computation speeds achieved by classical computers.

Year	Quantum operations per second
2013	2.64×10^6
2015	2.31×10^6
2016	6.25×10^6
2017	2.5×10^7
2018	4×10^7
2019	7.2×10^7

Table 4.1: Peak gate speeds achieved in superconducting devices.

to minimize. The RSE is a measure of the difference between the actual and predicted gate speeds, squared to emphasize larger discrepancies. The hyperparameters tuned in our regression model primarily included the regularization parameter and the mixing parameter in the elastic net regression (referenced in Chapter 4.2.1).

Chapter 5

Results and Discussions

In this chapter, we present our findings pertaining to qubits and gate speeds predictions. We compare our qubit predictions to the related works and discuss the relevant assumptions imposed onto our gate speeds model. A sub-chapter is dedicated to a discussion exploring important considerations, such as the validity of assuming qubits will follow a growth function akin to the number of transistors, the impact of existing hardware limitations and technological advancements on the future trajectory of quantum computing, and the appropriateness of envisioning a ”Quantum Moore’s Law”.

5.1 Qubits

In Figure 5.1, we present the predictive capacities of two distinct models. These models attempt to predict the progression of qubit capacity, which is a key metric of the overall performance of quantum computers. The blue line represents a model based on a polynomial relationship between qubit capacity and transistor capacity. On the other hand, the red line signifies a model which posits a polylogarithmic relationship. Comparing the two, it is clear that the polynomial model tends to forecast higher qubit capacities, making it more optimistic than its polylogarithmic counterpart. Interestingly, both models follow an upward trend, indicating a steady increase in qubit capacities over time. However, the extent of this growth varies significantly depending on the model.

Figure 5.1: Qubit model predictions.

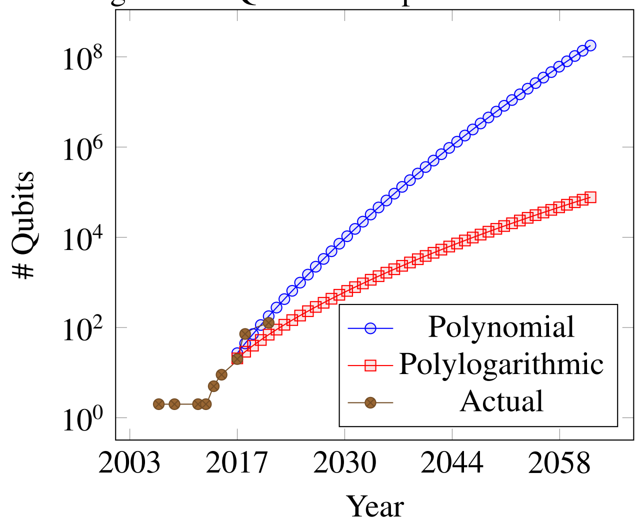

Table 5.1 presents a comparison of our model's predictions with those produced by Sevilla et al. [27] for the years 2017, 2018, and 2021. In the table, we also provide the actual recorded qubit capacities for those years. The Mean Squared Error (MSE) between the predicted and actual capacities for our model stands at approximately 441.33, which is notably lower than the MSE of the model by Sevilla et al. [27] at around 2209.82. Although our model overestimated the qubit capacity in 2017, it was notably more accurate for the years 2018 and 2021. This suggests that our model, while not perfect, tends to provide a more precise prediction of future qubit capacities.

In Table 5.2, we compare our model's predictions, those of Sevilla et al. [27], and the published IBM roadmap [2] for the years 2022 through 2026. Both our model and the model provided by Sevilla et al. [27] fall short of matching the IBM roadmap predictions. However, our model tends to provide predictions closer to those of IBM. For instance, for 2022, our model predicts a capacity of 173 qubits, while Sevilla et al. [27] predict 92 qubits, both significantly below IBM's forecast of 433 qubits.

Looking further into the future, the trend continues. Our model consistently predicts higher qubit capacities than the model in Sevilla et al. [27], with the gap widening as we move forward in time. For example, by 2026, our model forecasts a capacity of 692 qubits compared to the predication made by Sevilla et al. [27] prediction of 374 qubits. Still, both models significantly underestimate the IBM roadmap's predictions, which anticipate a capacity as high as 10,000 qubits by 2026. Looking further into the future, the trend continues. Our model consistently predicts higher qubit capacities than the model in Sevilla et al. [27], with the gap widening as we move forward in time. For example, by 2026, our model forecasts a capacity of 692 qubits compared to the predication made by Sevilla et al. [27] of 374 qubits. Still, both models significantly underestimate the IBM roadmap's predictions, which anticipate a capacity as high as 10,000 qubits by 2026.

In summary, while both our model and the model by Sevilla et al. [27] struggle to match the highly optimistic forecast put forth by IBM's roadmap, our model generally presents more accurate predictions. However, given the rapid and uncertain nature of advancements in quantum computing, these predictions should be interpreted with caution. Nevertheless, they provide valuable insights into possible trends in quantum computing technology and can guide future research and development efforts.

	2017	2018	2021
Sevilla et al. [27]	16	23	65
Our model	24	37	120
Actual	17	72	127

Table 5.1: Historical comparison of our predictions to Sevilla et al. [27] and the true qubit capacities.

	2022	2023	2024	2025	2026
Sevilla et al. [27]	92	131	187	262	374
Our model	173	247	351	494	692
IBM Roadmap	433	1121	1386	4158	10,000

Table 5.2: IBM roadmap comparison.

5.2 Gate Speeds

Our models predict an exponential surge in the quantum gate operation frequency over the next few years, assuming the parallel evolution of classical control circuits to keep pace with these accelerated quantum gate operations. Despite an initial phase of rapid acceleration, we anticipate a considerable slowdown in growth, attributed primarily to the requisite for increasingly faster classical control circuits. However, it is important to note that due to current hardware limitations, the technology is not capable of executing over 1 billion gates per second. Butko et al. [6] describes the limitations of classical control circuits in managing quantum gate operations at different frequencies. The authors state that these circuits struggle to accurately execute quantum gate operations beyond a certain frequency. For instance, a processor running on an FPGA board can guarantee timely gate delivery for up to a density of eight gates with the immediate format and up to a 32-gate density with the mask format if there are only two types of gates. ASIC implementations running at around 2GHz can execute any 32-qubit circuit using the immediate format or mask for circuits with a gate diversity of eight types and lower. Beyond these gate densities and diversities, the processor fails to deliver control gates on time, resulting in circuit execution inaccuracy and erroneous result. Therefore, our prediction function should embody this inherent property. As the operation frequency approaches 1 billion gates per second, the increase becomes progressively more challenging. Unfortunately, our existing models do not adequately reflect this property, necessitating the design of a new model. This new model can be derived by applying the function $f(x) = -\frac{1}{x} + 1e^9$ to the second column of the classical dataset. The key intuition here is the introduction of a function which exhibits a bend similar to the logarithmic function. If the functional relationship between quantum gates per second and classical computations per second mirrors this, the function's increase will diminish as we approach $1e9$. We capped the frequency cap at 50 GHz. This frequency caps reflects our expectation that the classical control circuits will struggle to manage quantum gate operations at frequencies beyond these limits. For this new model, we choose the data points from 2016 and 2017 to learn the polynomial relationship and validate it against the gate speed

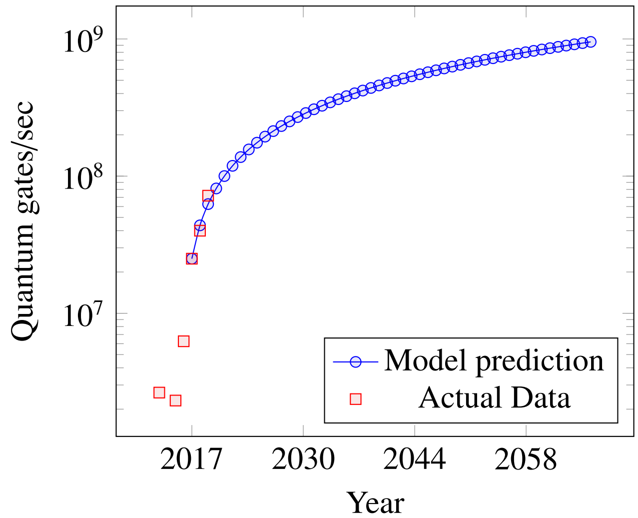

Figure 5.2: Gate speed model predictions.

of years 2018 and 2019.

5.3 Discussions

The subsequent subsections examine three primary results derived from this work. These discussions explore the growth trajectory of qubits in comparison to classical computing transistors, reconciling current hardware limitations and technological advancements and the appropriateness of envisioning a ”Quantum Moore’s Law”.

5.3.1 Is it fair to assume that qubits will follow the same growth function as the number of transistors?

While the rapid progression of transistor development has greatly impacted the landscape of classical computing, it’s not necessarily fair to directly map this trajectory onto quantum computing’s growth. While parallels can be drawn between the two technologies, there exist significant differences in their respective physical properties and implementation challenges. Qubits, in their superconducting guise or otherwise, are faced with unique obstacles such as maintaining coherence, or the ability of a qubit to sustain a superposition of states long enough for meaningful computation.

Environmental noise, along with inter-qubit interactions, often compromise this coherence time. Moreover, the fabrication and control of qubits present complex, resource-intensive problems, rendering current manufacturing techniques somewhat unscalable. However, these limitations have stimulated a flurry of research aimed at discovering innovative qubit designs and materials, as well as enhanced error correction and fault tolerance methods. Despite the growth function of qubits potentially deviating from that of transistors, continued research and development in this area holds promise for the advancement of quantum computing. It is also worth noting the degree of human ingenuity when developing solutions to novel problems. Even in the present day, there are ongoing efforts to sustain the Moore's Law trend observed over the past few decades, despite an expected slowdown in its trajectory due to physical and material design limitations, as discussed by Van Schoot et al. [31]. It is entirely conceivable that analogous breakthroughs in materials and quantum computer development could emerge, sustaining a comparable growth trajectory.

5.3.2 How do current hardware limitations and technological advancements shape the future of quantum computing?

Hardware constraints currently impose significant restrictions on the growth and scalability of quantum computing. These limitations, such as maintaining qubit coherence, designing efficient quantum gates, and executing operations at high frequencies, present both challenges and opportunities for researchers and engineers. As quantum computing technology evolves, new strategies to overcome these hurdles are continually being developed. These include advancements in qubit design, error correction algorithms, and manufacturing processes. On the other hand, the limitations also set the pace for the evolution of quantum computing, providing a realistic lens through which future growth can be projected. Therefore, understanding these constraints, and keeping pace with technological advancements, is pivotal for the accurate modeling and prediction of the progression of quantum computing.

In our study, for instance, we incorporated these aspects into our predictive models, reflecting the inherent complexities associated with the progression of quantum computing technology. This approach not only allows for a more realistic prediction but also provides valuable insights that can guide future research and development efforts. As the field continues to evolve, refining and updating these models will remain an ongoing task, crucial to shaping the future of quantum computing.

5.3.3 Is it fair to envision a ”Quantum Moore’s Law”?

When it comes to quantum computing, it is tempting to envision an analogous ”Quantum Moore’s Law”, where quantum computational power, measured in terms of qubit counts or quantum gate speeds, might follow a similar exponential growth trajectory. However, the direct application of Moore’s Law to quantum computing can be overly simplistic and potentially misleading. The physical properties, manufacturing challenges, and technological limitations of qubits are fundamentally different from those of transistors. Notably, maintaining qubit coherence and creating reliable quantum gates present significant challenges that can impede the doubling rate of qubits.

Moreover, the metric for computational power in quantum computing is not as straightforward as in classical computing. It is not solely about the number of qubits or gate speeds, but also about the quality of qubits, error rates, connectivity, and the effectiveness of quantum error correction, among other factors. Therefore, a ”Quantum Moore’s Law” would have to account for these multi-faceted and interdependent aspects of quantum computing.

Nevertheless, certain trends hint at the rapid progression of quantum computing technology. Qubit counts and coherence times have been increasing, and error rates decreasing, albeit not at a consistent exponential pace. Researchers are continuously developing novel architectures, materials, and techniques to push these limits. Therefore, while a ”Quantum Moore’s Law” might not manifest in the same form as its classical counterpart, it is clear that the field of quantum computing is on a trajectory of rapid growth and development, albeit with a different set of guiding principles and constraints.

There is also the observation made by Hartmut Neven, Director of the Quantum Artificial Intelligence Lab at Google, regarding the progress of quantum computing. Neven's Law, also known as Neven's Rule, Neven's Law states that the power of quantum computers is roughly doubled every year, following a similar trajectory to Moore's Law, which describes the exponential growth of classical computer processing power. Neven's Law suggests that quantum computers can achieve significant advancements in terms of computational capabilities and performance over time, potentially leading to breakthroughs in solving complex problems that are intractable for classical computers. However, it is important to note that Neven's Law is an empirical observation and does not guarantee a fixed rate of progress in quantum computing.

Chapter 6

Conclusion

In conclusion, this work provides an improved method to forecast the future capacities of superconducting qubits and gate speeds based on the relationship between qubit and classical transistor capacities. We make use of a data-driven approach capable of accommodating new data points as quantum milestones are reached. The results of the analysis suggest that the authors' model is a more accurate predictor of qubit capacities compared to the model developed by Sevilla et al. [27]. However, both models struggle to match the trend of the IBM roadmap, which predicts much higher qubit capacities in the coming years. These predictions should be viewed with caution, given the inherent uncertainty in predicting future developments in quantum computing technology. Nevertheless, this work may provide valuable insight into the potential trajectory of quantum computing technology if Moore's law continues to hold in this domain. Further research in this area could help to refine these predictions and improve our understanding of the future potential of quantum computing.

6.1 Future Work

As a logical extension to our work, several directions for future research can be explored:

- **Refining and expanding the predictive models:** The models presented here primarily revolve around the relationship between qubit and classical transistor capacities. Further research can explore the inclusion of additional parameters that influence quantum computer performance. These might include the quality of qubits, quantum gate error rates, qubit connectivity, and the efficiency of quantum error correction techniques. By expanding the model to incorporate these factors, it may be possible to enhance its predictive accuracy and make it more representative of the multi-faceted nature of quantum computing advancements.

- **Incorporating advances in quantum architectures and technologies:** Our models primarily consider superconducting qubits. There are, however, several other types of qubits being researched and developed, such as topological qubits, trapped ions, and photonic qubits, each with their own advantages and challenges. Future work could consider these alternative technologies and their impact on the development trajectory of quantum computers.

- **Exploring quantum-classical hybrid systems:** As quantum computing technology progresses, so does classical computing technology. This suggests potential for quantum-classical hybrid systems, where the strengths of each can be leveraged to compensate for the weaknesses of the other. Further exploration of these hybrid systems might present a novel avenue for advancing the computational landscape.

- **Performing periodic reassessments of the model:** Given the rapid and dynamic nature of advancements in quantum computing, it would be prudent to perform periodic reassessments of the model's predictive accuracy. This would allow for necessary recalibrations and modifications to keep pace with the evolving field of quantum computing.

References

[1] Moore's law: The number of transistors per microprocessor. `https://ourworldindata.org/grapher/transistors-per-microprocessor`.

[2] Ibm unveils breakthrough 127-qubit quantum processor, 2022. `https://newsroom.ibm.com/2021-11-16-IBM-Unveils-Breakthrough-127-Qubit-Quantum-Processor`.

[3] Divesh Aggarwal, Gavin K Brennen, Troy Lee, Miklos Santha, and Marco Tomamichel. Quantum attacks on bitcoin, and how to protect against them. *arXiv preprint arXiv:1710.10377*, 2017.

[4] Kai Arulkumaran, Antoine Cully, and Julian Togelius. Alphastar: An evolutionary computation perspective. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '19, page 314–315, New York, NY, USA, 2019. Association for Computing Machinery.

[5] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[6] Anastasiia Butko, George Michelogiannakis, Samuel Williams, Costin Iancu, David Donofrio, John Shalf, Jonathan Carter, and Irfan Siddiqi. Understanding quantum control processor capabilities and limitations through circuit characterization. In *2020 International Conference on Rebooting Computing (ICRC)*, pages 66–75. IEEE, 2020.

[7] Yu Chen, C Neill, Pedram Roushan, Nelson Leung, Michael Fang, Rami Barends, Julian Kelly, Brooks Campbell, Z Chen, Benjamin Chiaro, et al. Qubit architecture with high coherence and fast tunable coupling. *Physical review letters*, 113(22):220502, 2014.

[8] KR1442 Chowdhary. Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649, 2020.

[9] Julien Codsi and John van de Wetering. Classically simulating quantum supremacy iqp circuits trough a random graph approach. *arXiv preprint arXiv:2212.08609*, 2022.

[10] Matthew F Dixon, Igor Halperin, and Paul Bilokon. *Machine learning in Finance*, volume 1406. Springer, 2020.

[11] Craig Gidney and Martin Ekerå. How to factor 2048 bit rsa integers in 8 hours using 20 million noisy qubits. *Quantum*, 5:433, 2021.

[12] Élie Gouzien, Diego Ruiz, Francois-Marie Le Régent, Jérémie Guillaud, and Nicolas Sangouard. Computing 256-bit elliptic curve logarithm in 9 hours with 126133 cat qubits. *arXiv preprint arXiv:2302.06639*, 2023.

[13] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996.

[14] Xin He, Kaiyong Zhao, and Xiaowen Chu. Automl: A survey of the state-of-the-art. *Knowledge-Based Systems*, 212:106622, 2021.

[15] Torsten Hoefler, Thomas Häner, and Matthias Troyer. Disentangling hype from practicality: on realistically achieving quantum advantage. *Communications of the ACM*, 66(5):82–87, 2023.

[16] Xin-Lin Huang, Xiaomin Ma, and Fei Hu. Machine learning and intelligent communications. *Mobile Networks and Applications*, 23(1):68–70, 2018.

[17] Samidha Khatri, Aishwarya Arora, and Arun Prakash Agrawal. Supervised machine learning algorithms for credit card fraud detection: a comparison. In *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 680–683. IEEE, 2020.

[18] Jonathan Koomey, Stephen Berard, Marla Sanchez, and Henry Wong. Implications of historical trends in the electrical efficiency of computing. *IEEE Annals of the History of Computing*, 33(3):46–54, 2010.

[19] Lars S Madsen, Fabian Laudenbach, Mohsen Falamarzi Askarani, Fabien Rortais, Trevor Vincent, Jacob FF Bulmer, Filippo M Miatto, Leonhard Neuhaus, Lukas G Helt, Matthew J Collins, et al. Quantum computational advantage with a programmable photonic processor. *Nature*, 606(7912):75–81, 2022.

[20] Chetan Nayak. Microsoft achieves first milestone towards a quantum supercomputer. Blog, 06 2023.

[21] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 809–816, Madison, WI, USA, 2011. Omnipress.

[22] Alexandru Paler and Simon J Devitt. An introduction to fault-tolerant quantum computing. *arXiv preprint arXiv:1508.03695*, 2015.

[23] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[24] The Danh Phan. Housing price prediction using machine learning algorithms: The case of melbourne city, australia. In *2018 International conference on machine learning and data engineering (iCMLDE)*, pages 35–42. IEEE, 2018.

[25] D Karthika Renuka, T Hamsapriya, M Raja Chakkaravarthi, and P Lakshmi Surya. Spam classification based on supervised learning using machine learning techniques. In *2011 International Conference on Process Automation, Control and Computing*, pages 1–7. IEEE, 2011.

[26] Alex Sanchez-Stern, Yousef Alhessi, Lawrence Saul, and Sorin Lerner. Generating correctness proofs with neural networks. In *Proceedings of the 4th ACM SIGPLAN International Workshop on Machine Learning and Programming Languages*, pages 1–10, 2020.

[27] Jaime Sevilla and C Jess Riedel. Forecasting timelines of quantum computing. *arXiv preprint arXiv:2009.05045*, 2020.

[28] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999.

[29] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy Lillicrap, Fan Hui, Laurent Sifre, George Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of go without human knowledge. *Nature*, 550:354–359, 10 2017.

[30] Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069*, 2018.

[31] Jan Van Schoot. The moore's law machine: The next trick to tinier transistors is high-numerical-aperture euv lithography. *IEEE Spectrum*, 60(9):44–48, 2023.

[32] Jacob VanderPlas, Andrew J Connolly, Željko Ivezić, and Alex Gray. Introduction to astroml: Machine learning for astrophysics. In *2012 Conference on Intelligent Data Understanding*, pages 47–54. IEEE, 2012.

[33] Ledell Wu, Adam Fisch, Sumit Chopra, Keith Adams, Antoine Bordes, and Jason Weston. Starspace: Embed all the things! In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Curriculum Vitae

Name:	Christopher Tam
Post-Secondary Education and Degrees:	University of Western Ontario London, ON 2014 - 2020 B.Sc. in Computer Science 2020 - 2023 M.E.Sc. in Software Engineering - Candidate
Awards:	2020-2022
Related Work Experience:	Teaching Assistant University of Western Ontario 2020 - 2021 Data Engineering Internship EQ Works 2021-2022 Product Manager, Head of Partnerships BTQ Technologies Corp. 2021-Present

Publications:

- Christopher Tam, Afrad Basheer, Eric Howard, and Luiz Fernando Capretz, *Quantum Moore's Law Using Classical Machine Learning*, Journal of Systems and Software (Special Issue on Software Engineering for Hybrid Quantum Computing Systems), Elsevier, submitted, 20 pages, 2023.
- Christohper Tam and Luiz Fernando Capretz, *Investigating Continual Learning Strate-*

gies in Neural Networks, The 49th Annual Conference of the IEEE Industrial Electronics Society (IECON), Singapore, accepted, 6 pages, 2023.