Chapter 26

Quantum Neural Networks

Application for credit risk assessment and fraud detection

Sebastian Zajac & Krzysztof Kuba

26.1 Introduction

Quantum computers have introduced a new paradigm in computing, offering transformative potential across various domains such as physics, chemistry, logistics, and finance [1-3]. Within the financial sector, one of the most critical challenges remains the accurate assessment of credit risk [4].

Formally, credit risk is defined as the probability that a debtor will fail to fulfill their obligations to a financial institution. Precise estimation of this risk is fundamental to the stability of the banking sector and the effective allocation of capital within the economy. Since the emergence of the modern banking system—and particularly in the wake of numerous financial crises in the 20th and 21st centuries—credit risk modeling has become a cornerstone of economic, financial, and regulatory research. Furthermore, regulatory frameworks such as Basel II and III have established the importance of quantitative risk measures and standardized the approaches used by financial institutions worldwide [5, 6].

Modern methods of credit scoring are based on advanced statistical techniques and machine learning models, which allow for the processing of huge volumes of data coming from disparate sources: financial, behavioral, transactional, and macroeconomic. The primary objective of these models is not only to classify

This chapter is a featured contribution to the monograph *A Portrait of Quantum Technologies in Finance*, edited by Oswaldo Zapata, PhD, and published by The Quantum Finance Boardroom (2026).

clients according to their credit risk profiles but also to gain deeper insights into the underlying factors influencing default. Traditionally, classical credit risk models have relied on logistic regression, decision trees, random forests, and deep neural networks. In this chapter, we focus our attention on the fundamental architecture of neural networks, which serves as a necessary precursor to understanding their quantum counterparts: quantum neural networks.

The efficacy of these models is heavily contingent upon the availability and quality of data, as well as the underlying computational capacity. In modern financial institutions, datasets frequently scale to hundreds of millions of records and tens of thousands of variables, necessitating advanced parallel computing and cloud infrastructure. Yet, as financial markets grow in complexity, so too does the demand for model sophistication, leading to a significant surge in computational costs. While past technological advancements have kept pace by scaling the power of classical systems, further progress is increasingly hindered by the fundamental physical limitations encountered at the nano-scale [7, 8].

In this context, quantum computers offer a transformative alternative by leveraging phenomena such as superposition and entanglement. These properties enable the pursuit of *Quantum Advantage*—a state where quantum algorithms demonstrate superior speed or efficiency over classical methods. Such improvements in efficiency may be realized through the reduction of model parameters, which in turn significantly lowers the overall computational burden.

This evolution of computational paradigms is not merely a technical shift, but a testament to the profound convergence of fundamental physics and practical data science. For the authors, it represents a deliberate return to the roots of physical intuition within the complex landscape of modern finance.

The first author's journey to Quantum Machine Learning began with a PhD in theoretical physics, specializing in neutrino physics and the abstract beauty of category theory. For many years, the mathematical elegance of morphisms and quantum structures seemed far removed from the daily grind of credit risk modeling and MLOps. However, experience in the banking sector and at the Warsaw School of Economics (SGH) revealed a hidden truth: the vast, noisy streams of financial data are not just tables of numbers, but complex systems that mirror the dynamics of physical fields. By treating data points as excitations of quantum fields—a core focus of his current research—he seeks to bridge the gap between high-level theory and business utility. This approach is born from the conviction that an analyst's greatest strength lies in returning to these physical roots, where the laws of nature provide the ultimate framework for understanding market uncertainty.

Complementing this vision, Krzysztof Kuba brings a perspective that bridges the gap between the microscopic and the macroeconomic. With a background starting in nanoengineering and evolving into quantitative finance, his path reflects the same transition from the physical world to the abstract world of markets. Having passed Level I of the CFA Program, futher grounded his knowledge in finance. Therefore, he ensures that these quantum-inspired theories are not just elegant abstractions, but robust, scalable tools capable of meeting the demands of real-world financial engineering.

At finQbit, we do not view QML as a "black-box" replacement for classical AI. Instead, we see it as a beautiful synthesis: an opportunity to apply the most sophisticated tools of theoretical physics to the very human problems of risk and value. This chapter is an invitation to look at financial modeling not just as an exercise in statistics, but as a realization of quantum mechanics in action.

While quantum computers are still in the stage of intensive research, they are already finding applications in optimization and machine learning—a field known as *Quantum Machine Learning*. In this chapter, we will explore the fundamental concepts of QML, from data encoding to the measurement of quantum states. Following this, we will examine a practical model for credit card fraud detection, providing a foundation for broader applications of quantum machine learning in finance. However, to fully appreciate these quantum concepts, it is essential to first understand their classical counterparts.

26.2 Classical Machine Learning

Technological advancement and digitization have made data a common and valuable resource [3]. It is generated and processed in both structured and unstructured formats. The structuralization of data led to the development of various models, commonly referred to as *Machine Learning*. However, the processing of unstructured data—such as text, images, or videos—contributed to the development of *Deep Learning Models*. Both of these approaches are often collectively called *Artificial Intelligence*. They were created primarily to recognize patterns, but increasingly, they are used to model and generate entirely new data.

26.2.1 Back to the feature: the machine learning process

To appreciate the quantum leap in financial modeling, one must first master the classical framework that precedes it. Understanding the machine learning (ML) process is not merely a procedural requirement; it is about grasping how raw information is transformed into actionable intelligence. This process can be divided into five fundamental stages:

1. Data collection
2. Data cleaning and preparation
3. Model training
4. Model validation
5. Model deployment and monitoring

The journey begins with **data collection**. In the financial realm, data is the lifeblood of the model, sourced from internal bank records, market indices, macroeconomic indicators, and even the sentiment of social media. We distinguish between *structured data*—information that fits neatly into relational tables—and *unstructured data*, such as text or images, which requires more sophisticated handling to reveal its patterns.

Next is **data cleaning**, a stage governed by the fundamental principle: "garbage in, garbage out." If a model is fed low-quality, noisy data, its outputs will inevitably be flawed. Real-world financial data is notoriously messy, often riddled with missing values or extreme outliers that can significantly skew the learning process.

Once the data is refined, we select and **train the model**. The choice of the model depends on what we are trying to predict. If our goal is to assign observations to discrete categories—such as identifying a transaction as either "fraudulent" or "legitimate"—we are dealing with a *classification* problem. If, however, we aim to predict a continuous value, such as a stock price or an option's fair value, we utilize *regression*.

The distinction further depends on the availability of **labels**. In *supervised learning*, the model learns from historical examples where the outcome is known (e.g., predicting customer churn based on past behavior). In contrast, *unsupervised learning* seeks to uncover hidden structures in data without predefined labels, such as clustering clients with similar risk profiles. A third paradigm, *reinforcement learning*, involves an agent that learns to make decisions by receiving feedback (rewards or penalties) from its environment.

After training, the model must undergo rigorous **testing and validation**. This stage ensures the model generalizes well to unseen data and that the underlying business logic remains sound. We must ask: is the level of inaccuracy acceptable? In the high-stakes world of banking, a poorly validated model can lead to catastrophic financial losses or severe reputational damage.

Finally, the model is **deployed** into a production environment. However, deployment is not the end of the journey. Constant monitoring is required to detect "drift" and ensure the model remains aligned with shifting market dynamics. It is crucial to view this entire lifecycle as iterative; a failure in validation often necessitates a return to data cleaning or feature engineering.

26.2.2 Neural networks: optimization as the new paradigm

Neural networks (NN) are a class of machine learning models inspired by the structure and function of the human brain. They consist of interconnected layers of nodes, or "neurons," that process and transmit information by assigning weights to input data and adjusting these weights through learning algorithms.

In the modern landscape of artificial intelligence, machine learning can be fundamentally viewed as the process of defining a specific **network architecture paired with an appropriate loss function**. The challenge is no longer about finding a closed-form analytical solution to a problem—which classical statistics often sought—but rather about navigating a high-dimensional landscape to find the optimal set of parameters. This involves critical decisions: choosing the right architecture, selecting an efficient optimization algorithm, and defining a cost function that accurately captures the nuances of the business problem.

Perhaps the most widely used architecture is the feed-forward neural network, as

shown in Figure 1. To understand how this works in practice, let us consider the credit card fraud detection problem. Here, the goal is to classify transactions as either legitimate (0) or fraudulent (1). The input layer receives variables such as transaction amount, time, and location. These signals are multiplied by weights, passed through non-linear *activation functions*, and propagated through hidden layers until a prediction is reached.

The difficulty of this task is twofold. First, financial data is often heavily imbalanced—fraudulent transactions are rare, making it hard for the model to learn their patterns without biased results. Second, we face the dilemma of misclassification: a *false positive* (blocking a valid payment) causes customer frustration, while a *false negative* (missing a theft) leads to financial loss.

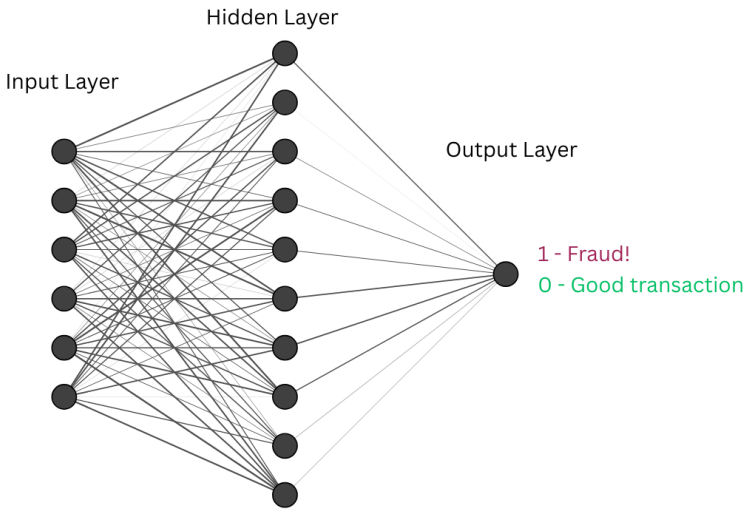

Figure 26.1: A simple neural network design. Each circle acts as a "neuron," and each neuron from a given layer is connected to every other neuron in the next layer. The bigger the weight of connection, the darker the shade.

The true power of a neural network lies in its ability to act as a universal approximator, fitting virtually any arbitrary function. However, this power comes at a high computational price. Training a model involves back-propagation [9], an algorithm that calculates the "gradients" of the loss function with respect to every weight in the network. As we increase the number of layers and parameters to capture more complex patterns, the search for the global minimum of the loss function becomes exponentially difficult.

This is where the weakness of classical computing emerges: the sheer volume of floating-point operations required for such high-dimensional optimization starts to hit the ceiling of current hardware capabilities. Classical systems struggle to find an efficient path through a massive, non-convex loss landscape. This bottleneck is exactly what motivates our transition into the quantum realm. Where a classical computer struggles to find an analytical or numerical path through a massive, non-convex loss landscape, quantum systems offer a new way to explore and represent these complex spaces.

26.3 Quantum Machine Learning

The current phase of quantum technology development is widely referred to as the Noisy Intermediate-Scale Quantum (NISQ) era [10]. The term "Intermediate-Scale" means that contemporary quantum processors have a limited number of qubits to operate, typically a few hundred. For comparison, most daily-use laptops have 16 GB of RAM, which is roughly 128 billion bits. This constraint means that quantum computers currently remain unsuitable for general-purpose computation. Moreover, the "Noisy" term means that the outputs of quantum processors are subject to error. This error can be partially mitigated by Quantum Error Correction (QEC) algorithms. However, these schemes are not yet fully implemented. A further limitation stems from the requirement that quantum gate operations be performed on timescales much shorter than the characteristic decoherence times of the qubits, thereby restricting the achievable circuit depth and hindering the implementation of large-scale, highly complex quantum algorithms.

Despite these limitations, NISQ devices are anticipated to enable the demonstration of Quantum Advantage for computational tasks of practical relevance. In contrast, experimental realizations of Quantum Supremacy to date have been restricted to highly specialized benchmark problems—most notably random circuit sampling—that are not directly applicable to real-world domains such as quantitative finance.

A central class of methods in this regime is based on Parameterized Quantum Circuits (PQCs), which we will explore in this section. In such approaches, quantum circuits containing tunable parameters are trained using classical optimization algorithms to determine the most optimal parameter values. This paradigm, which combines classical optimization routines with quantum information processing, is commonly referred to as hybrid quantum-classical learning.

26.3.1 Parameterized quantum circuits (PQC)

PQCs are the cornerstone of hybrid quantum-classical algorithms. Each gate in the circuit can depend on a set of continuous parameters, which are optimized using classical methods.

Parameterized quantum circuits are realized through a predetermined arrangement of quantum *gates*, augmented by tunable, parameter-dependent gates. These gates are operations performed on a qubit, enabling highly expressive, non-linear transformations of quantum states [11, 12]. Quantum Machine Learning (QML) methodologies are being developed on the basis of such circuits and encompass a broad class of so-called Variational Quantum Algorithms (VQAs). These models can be implemented using Python-based software development kits, such as IBM Qiskit, PennyLane, or Cirq, as well as the specialized finQbit library, and can be executed on both simulators and contemporary quantum processing units (QPUs).

26.3.2 Data as field excitations: a new encoding paradigm

Traditional QML approaches often treat data encoding—the process of mapping classical data x into a quantum state $|\psi(x)\rangle$—as a mere technical necessity, often using simple angle or amplitude encoding. However, as established in our introduction, we propose a more fundamental perspective: treating classical financial data as excitations of quantum fields.

In this framework, the initial state preparation is not just a transformation of coordinates, but the creation of a field configuration. If we consider the financial variables (such as credit history or market volatility) as local field values, the process of encoding them into a quantum circuit becomes equivalent to simulating the interaction of these fields with the qubits. This perspective allows us to utilize the rich mathematical structure of Quantum Field Theory (QFT) to design more robust feature maps. By viewing the input data as a set of excitations, we can better capture the correlations and "entanglement" inherent in financial markets, which are often missed by classical statistical methods. This theoretical bridge ensures that the PQC does not just process numbers, but involves a physical representation of the underlying financial reality.

26.3.3 Quantum neural network model

The implementation of a Quantum Neural Network (QNN) follows a process structured into three main sequential stages:

1. Data preparation (Encoding)
2. Parametrized Quantum Circuit (The Ansatz)
3. Measurement

The **data preparation** step serves a purpose analogous to embeddings in classical neural networks and requires prior steps like data exploration, cleaning, and standardization. In the context of quantum computers, we must translate classical data into quantum states—a process formally known as *data encoding*. This is precisely where our conceptualization of data as *field excitations* comes into play. Rather than merely mapping a numerical value to an angle, we treat data encoding as the preparation of a physical field state, embedding classical financial variables into the exponentially large Hilbert space of the quantum system. The schematic representation of this step is shown in Figure 2. Moreover, if the dataset is highly dimensional, additional dimensionality reduction techniques may be required before encoding.

Next, we introduce the quantum equivalent of a "hidden layer": the parametrized quantum circuit. In quantum jargon, this layer is often called an *ansatz* (a term derived from German, meaning an "educated guess" or "assumption"). Because we rarely know the perfect circuit structure in advance, we postulate a sequence of gates and iteratively optimize their parameters. Choosing this structure is equivalent to defining the architecture (layers and neurons) in classical deep learning.

To get a sense of how the ansatz operates, observe Figure 2. There are two main components: the single-qubit rotation gates (e.g., U3 gates, represented as boxes) and

the two-qubit entangling gates (such as CNOT gates, which connect the horizontal qubit lines). You can think of the U3 gates as the nodes (neurons) in a neural network. These gates contain tunable *parameters* that are updated during the learning process, allowing the model to capture the underlying structure of the data. Furthermore, the CNOT gates are responsible for generating quantum *entanglement*, functioning similarly to the weighted connections that link neurons across different layers.

Designing an effective ansatz involves balancing expressibility (modeling capacity) against trainability. While more parameters and complex entanglement allow the model to capture highly non-linear data patterns, overly deep circuits risk encountering a phenomenon known as *Barren Plateaus*. In this scenario, the gradients vanish exponentially, making the training process entirely ineffective.

Last but not least is the **measurement** stage, which is analogous to the classical output layer. Measuring a qubit causes its quantum state to collapse into a classical bit (either 0 or 1). Returning to the credit card fraud problem discussed earlier: similar to a classical NN, if the result of the measurement is 0, the transaction is classified as normal. Conversely, a measurement of 1 alerts us to a potential fraud.

Formally, there is one additional component required to close the learning loop: the *cost function* (or loss function). The purpose of this function is to evaluate how much our model's prediction differs from the true label. The output of this cost function is then passed to a classical optimizer—often utilizing techniques like the *parameter-shift rule* (the quantum analogue of back-propagation)—which fine-tunes the U3 gate parameters to fit the data.

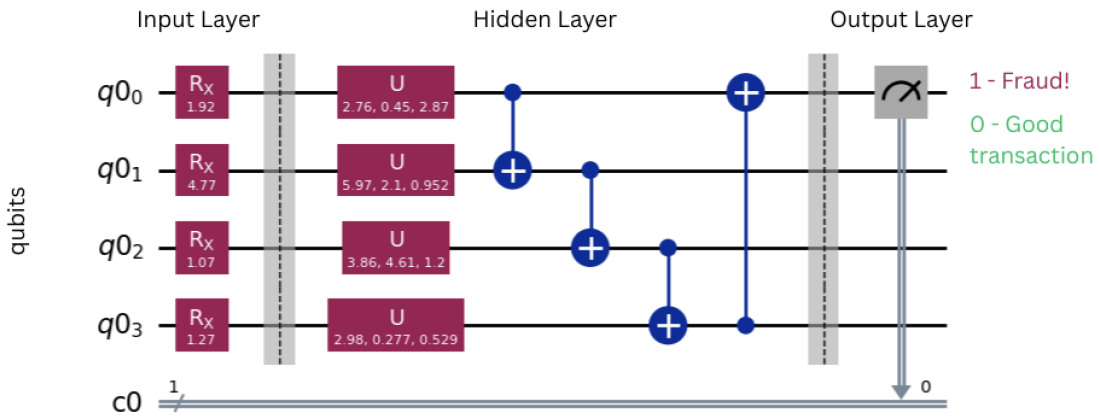

Figure 26.2: Representation of the Quantum Neural Network, from the Input Layer (Data Encoding), via the hidden layer (the ansatz), to the output layer (measurement of the qubit).

Having discussed the architecture of quantum neural networks, let us dive deeper into the intuition behind data encoding and the ansatz. Currently, our classical credit card data lives in a 2-dimensional table on a flat piece of paper. Imagine coloring the points representing normal transactions green, and the fraudulent ones red. Because this "paper" contains hundreds of thousands of densely packed dots, drawing a single straight line that perfectly separates the green dots from the red ones is impossible.

A classical linear classifier can try its best, but it will inevitably fail.

Now, instead of points confined to a 2-dimensional space, imagine these data points as marbles. Throw all these marbles onto a trampoline, bouncing them up into the air. Suddenly, the data is no longer flat! By popping the data into a new, higher dimension (the space above the trampoline), hidden patterns emerge. This additional dimension provides the spatial freedom to simply slide a flat piece of cardboard (a hyperplane) right between the red and green marbles.

To transfer this analogy to quantum computing: *data encoding* is the process of turning flat dots into marbles, the *ansatz* acts as the trampoline launching them into a high-dimensional quantum Hilbert space, and the *measurement* is the flat cardboard plane that cleanly distinguishes between normal and fraudulent behavior. This provides a vivid intuition of what happens when we map data into higher dimensions, illustrating the core power of quantum machine learning.

26.3.4 Is it all any useful?

So far, we have seen how quantum neural networks are different from classical ones. Now the question arises: can we do anything useful with it? The answer is yes, of course! There are several applications of quantum machine learning for classification problems, including, predicting customer behavior, grouping clients and fraud detection.

To begin with, the variety of financial products and their customization requires the ability to predict customer behavior. Banks desperately want to anticipate when a client will buy their first home, when they might abandon their account for a competitor, or which specific investment product they actually need right now. Unfortunately, human decisions are influenced by many different factors, and sometimes are even random and unpredictable. The decrease in monthly spending may happen just by accident, or it may signal upcoming financial problems of an individual. The aim of machine learning models is to distinguish between these two scenarios. Bombarding a customer with loan offers of financial products at the wrong time can discourage the client from the bank. On the other hand, missed opportunity can cause the customer to switch to a competitor.

With the wide verity of highly customizable products, banks may be interested in assigning each customer to a specific group, called *cluster*. By doing so, the bank can advertise product to group of clients with similar habits, needs and risk profiles. In this way, we can offer more personalized services rather than irrelevant, generic products that are almost instantly rejected by the customer. To visualize this problem, imagine trying to organize a massive library. A classical computer might simply group the books by broad categories like genre or author name. In finance, this refers to lumping customers together by basic demographics such as age or zip code, which completely misses the nuanced behavior of how people actually manage their money. As banks collect thousands of distinct data points on every individual, from daily coffee purchases to long-term investment strategies, the sheer volume of variables creates a multi-dimensional puzzle, just like with the green and red marbles. Classical algorithms struggle to calculate the mathematical “distance” or similarity

between customers when there are too many competing factors. In practice, this may result in inaccurate grouping, where a risk averse retiree might be mistakenly offered the same high-risk investment product as an aggressive day trader.

And last but not least, the credit card fraud problem. We have already introduced this problem in the context of neural networks. Let us dive deeper into this problem and investigate the results of a classical neural network model and a quantum version. To show that quantum models can possibly have better capabilities than the classical neural network, we trained both models on the credit card fraud dataset [13]. First, we trained a simple neural network with 30 neurons in the input layer, each neuron corresponding to a different variable. There were also 2 hidden layers, with 8 and 4 neurons each. Finally, the output layer consisted of only one neuron, with output between 0 and 1. This gives a total number of parameters equal to 276. The results of the validation step of the model are shown in Table 1.

The quantum model was essentially the same as the one presented in Figure 2, with 4 qubits, with 4 encoded variables. However, there were 4 hidden layers, instead of only 1 layer, in order to improve model capabilities to recognize patterns. With each U_3 gate containing three parameters, and one gate applied to each of the four qubits across four layers, the architecture possesses a total of $3 \times 4 \times 4 = 48$ trainable parameters. As for large language models, we are told that the more parameters, the better. We will see if this is actually the case for quantum machine learning.

One final remark on the training stage of both models - the trick we used to overcome the issue of imbalanced datasets was to train both models on a smaller dataset, which was balanced. By picking the same number of fraudulent and non-fraudulent transactions, the models had better ability to capture patterns and nuances. It may sound counter-intuitive, as we are told that the more data, the better. Well, not always, and it is the role of machine learning engineer to come up with different solutions to not so obvious problem and see what works best. However, this technique is not methodologically correct when we perform model validation. Each model should be evaluated on separate data that the model had not seen before, which ensures model ability to generalize. If the model performs well on training data, but fails on unseen examples, we might stumble on the problem of *overfitting*. Overfitting should be avoided because it means that the model learned data we provided for training in great detail, but the results are not applicable to unseen, real-world data. Imagine a elementary school student who learns how to add. One student memorized all answers from the exercises: $2 + 3 = 5$, $3 + 4 = 7$, and so on. The other student actually learned how to add numbers. The question on exam is $10 + 11$. Who is going to pass and who is going to fail? The same logic can be applied to overfitting. Having described the architectures of the models, let us put some numbers into perspective.

Table 26.1: Confusion Matrix: Classical Neural Network (NN)

Prediction	**Normal (0)**	**Fraud (1)**
True: Normal (0)	54,820	2,044
True: Fraud (1)	9	89

The tables above show the number of transactions that the model correctly and

Table 26.2: Confusion Matrix: Quantum Machine Learning (QML)

Prediction	**Normal (0)**	**Fraud (1)**
True: Normal (0)	56,762	102
True: Fraud (1)	12	86

incorrectly classified as normal or fraudulent. The truly correct examples are shown in the rows, while predictions of the models are shown in columns. These tables also directly shows the imbalance of the data set. There were almost 55,000 transactions that the model classified as normal, which were in fact, correct. This is called true negative. Here, the quantum model correctly predicts more true negatives than the neural network, which is the first sign of improved performance. Moreover, the number of false negatives (transaction that were classified as normal, where in reality they were frauds) is similar - 12 in case of quantum model and 9 in case of neural network. However, the false positives is the place where the quantum model reveals its potential. Look at the numbers: 102 versus 2,044! These numbers correspond to the case, where the transaction would have been blocked, even though it was completely normal. In reality, these would be almost 2,000 clients complaining about their card being blocked, and potentially leaving their current bank. Remember that these results were achieved with fewer variables (4 versus 30) and fewer parameters (48 versus 276).

26.3.5 Reality check

The empirical success of the quantum model in our credit card fraud example clearly demonstrates its potential to outperform classical architectures. However, does this mean quantum models will *always* reign supreme? To critically assess this claim, we must weigh the inherent advantages and limitations of both paradigms.

One of the primary advantages of classical neural networks is their profound ability to learn and adapt. Their architecture allows them to generalize complex, non-linear dependencies across massive datasets. Coupled with the highly efficient back-propagation algorithm, they have become the workhorses of modern artificial intelligence, powering everything from computer vision to Large Language Models (LLMs).

Furthermore, classical networks are highly scalable. While a relatively simple task like recognizing handwritten digits might require only a few dozen neurons, LLMs seamlessly scale to billions of parameters and hundreds of layers. Yet, this scalability is a double-edged sword. As financial data grows in complexity, the computational power required to train these massive models approaches absolute physical and thermal processing limits, creating a severe bottleneck for future development.

Additionally, classical networks notoriously struggle with imbalanced datasets. They often fail to correctly identify the underrepresented class, leading to high rates of false positives and false negatives. In our baseline test, the classical neural network produced an unacceptable rate of false positives (2,044 normal transactions incorrectly blocked), risking severe customer dissatisfaction. While a naive solution

might be to simply add more neurons, this does not guarantee better performance and significantly increases the risk of *overfitting*—a scenario where the model memorizes the training data but fails to generalize to the real world.

A critical limitation shared by both classical and quantum paradigms is the "black-box" problem of interpretability. In credit scoring, for instance, financial regulations often require banks to explicitly explain why a loan was declined in order to prevent discrimination and ensure equal access to capital. Consequently, a model's parameters must provide interpretable insights into which variables drove the decision. Because neither deep classical NNs nor QNNs natively offer this transparency, their deployment is often restricted to internal tasks (like fraud detection) rather than customer-facing credit decisions. To address this, researchers are actively developing frameworks broadly referred to as XAI, where the "X" stands for *Explainable* Artificial Intelligence.

Quantum neural networks share several of these broad advantages and limitations, but they introduce a fundamental shift in how information is processed. Their true power lies in their dimensionality. By mapping financial data into a high-dimensional quantum feature space—much like bouncing our scattered marbles off a flat surface into the air—QNNs can identify subtle, intricate relationships that classical models miss. Crucially, they achieve this with a fraction of the parameters. In our study, the quantum model achieved superior predictive results using a highly efficient architecture of just 4 qubits, 4 variables, and 48 parameters, slashing the number of false positives to a mere 102.

Despite these compelling theoretical and empirical advantages, practical QML is currently constrained by the Noisy Intermediate-Scale Quantum (NISQ) era. Modern quantum processors possess a limited number of qubits and suffer from high error rates due to environmental noise and decoherence. While researchers routinely use classical computers to flawlessly simulate quantum circuits up to around 30 qubits—which conveniently allows for the use of classical back-propagation to drastically speed up training—this *hybrid* simulation approach hits a hard physical memory limit beyond that scale.

When transitioning to actual quantum hardware (QPUs), we gain access to more qubits, but we lose the ability to use back-propagation. Because inspecting intermediate quantum states to calculate gradients would cause the wavefunction to collapse, hardware training must rely on gradient estimation techniques (like the parameter-shift rule) which require multiple circuit evaluations per parameter. Therefore, while QPUs offer an exponentially large state space, training time remains a significant bottleneck for large-scale quantum models today.

26.4 Summary

In this chapter, we have traversed the evolving landscape of machine learning, from foundational classical approaches to the frontier of quantum models. We began by identifying credit risk, credit scoring, and fraud detection as paramount challenges for modern banking. While classical models—such as feed-forward Neural Networks (NN)—have historically managed these complex datasets through rigorous machine

learning pipelines, they are increasingly constrained by the physical, thermal, and computational limits of classical hardware.

To overcome these limitations, we explored the mechanics of hybrid quantum-classical learning. Just as classical networks rely on input nodes, hidden layers with tunable weights, and non-linear activation functions, Quantum Neural Networks (QNNs) operate through analogous stages: Data Encoding (translating classical data into high-dimensional quantum states, akin to physical field excitations), The Ansatz (a parametrized quantum circuit utilizing tunable rotation gates and entanglement), and Measurement.

Our comparative analysis of a credit card fraud detection model provided a tangible demonstration of this potential. By mapping financial data into a quantum feature space, the QNN achieved superior precision, drastically reducing false positives (from 2,044 down to 102) while utilizing a mere fraction of the parameters required by its classical counterpart.

In conclusion, this chapter highlights the very real promise of *Quantum Advantage* in finance. It is not merely a theoretical concept, but a practical evolution: the ability to harness the fundamental laws of physics and high-dimensional geometry to solve complex, real-world classification problems more efficiently, accurately, and elegantly.

References

[1] R. Orús, S. Mugel, and E. Lizaso. "Quantum computing for finance: Overview and prospects." *Reviews in Physics* 4 (2019): 100028.

[2] D. Herman et al. "Quantum computing for finance." *Nature Reviews Physics* (2023): 1–25.

[3] Frontiers Editorial. "Quantum computing: foundations, algorithms, and emerging applications." *Frontiers in Quantum Science and Technology* (2025). Available at: `https://www.frontiersin.org/journals/quantum-science-and-technology/articles/10.3389/frqst.2025.1723319/full`.

[4] D. Kaszynski, B. Kaminski, and T. Szapiro. "Credit Scoring in Context of Interpretable Machine Learning." Warsaw: SGH, 2020.

[5] Basel Committee on Banking Supervision. *International Convergence of Capital Measurement and Capital Standards: A Revised Framework—Comprehensive Version.* Bank for International Settlements, June 2006. Available at: `https://www.bis.org/publ/bcbs128.htm`.

[6] Basel Committee on Banking Supervision. *Basel III: A Global Regulatory Framework for More Resilient Banks and Banking Systems.* Revised version (originally published December 2010). Bank for International Settlements, June 2011. Available at: `https://www.bis.org/publ/bcbs189.htm`.

[7] M. M. Waldrop. "The chips are down for Moore's law." *Nature* 530.7589 (2016): 144.

[8] G. E. Moore. "Cramming more components onto integrated circuits." *Electronics* 38.8 (1965): 114–117.

[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning representations by back-propagating errors." *Nature* 323.6088 (1986): 533–536. DOI: https://doi.org/10.1038/323533a0.

[10] J. Preskill. "Quantum Computing in the NISQ era and beyond." *Quantum* 2 (2018): 79. DOI: https://doi.org/10.22331/q-2018-08-06-79.

[11] A. P. Lund, M. J. Bremner, and T. C. Ralph. "Quantum sampling problems, BosonSampling and quantum supremacy." *npj Quantum Information* 3.1 (2017): 15. DOI: https://doi.org/10.1038/s41534-017-0018-2.

[12] A. Harrow and A. Montanaro. "Quantum Computational Supremacy." *Nature* 549 (2017): 203–209. DOI: https://doi.org/10.1038/nature23458.

[13] "Credit Card Fraud Detection Dataset." Kaggle. Available at: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud.