The Quantum Bottleneck: An Analysis of Data Balancing in QML for Security

Mansur Ziiatdinov[1,*,†], Salvatore Distefano[1,†]

[1]University of Messina

Abstract

Quantum computing promises advantage in data analysis by mapping the input data to a higher-dimensional Hilbert space and performing computations based on physics laws. However, its potential is limited by noise and number of qubits, restricting suitability to full fledged, end-to-end quantum machine learning solutions. Therefore, proper classical data preprocessing is required. As a preprocessing step, data sampling, i.e. the selection or generation of data items in imbalanced datasets, assumes strategic importance. Some undersampling techniques rely on pure random choice, while others use geometric considerations to select data items that better represent the original classes. As quantum machine learning approaches differ significantly from classical ones, it can be of interest to investigate sampling techniques in quantum and classical ML. In this study, an approach to combine classical and quantum data-driven ML workflows is proposed. Different undersampling techniques are then explored through classical and quantum ML models for cybersecurity threat classification on the well-known UNSW-NB15 network dataset taken from the literature to assess the proposed approach.

Keywords

quantum machine learning, undersampling, network intrusion detection

1. Introduction and Motivations

The rising sophistication, variety, scope and range of cyber threats represent a big challenge for modern digital infrastructures. Traditional security mechanisms, increasingly augmented by classical machine learning (ML), strive to detect anomalies and intrusions in real-time. However, classical ML models face mounting pressure from the sheer scale of network traffic and the ever-evolving, asymmetric nature of malicious activities. Two significant hurdles stand out: the curse of dimensionality, where the complexity of data can overwhelm classical algorithms, and the persistent problem of class imbalance, where anomalous or attack data is vastly outnumbered by normal traffic. This imbalance often biases classical models towards the majority class, causing them to miss rare but critical security events.

Quantum Computing offers a revolutionary computational paradigm with the potential to overcome classical limitations. By mapping input data into an exponentially large Hilbert space, Quantum Machine Learning (QML) promises to uncover intricate patterns that are intractable for classical systems. This intrinsic ability to operate in high-dimensional feature spaces suggests a powerful new tool for tasks like network intrusion detection. However, the current era of Noisy Intermediate-Scale Quantum (NISQ) devices imposes significant practical constraints. The performance of today quantum processors is fundamentally limited by qubit decoherence, gate errors, and restricted qubit counts, making end-to-end, fully-fledged quantum solutions presently unfeasible.

This reality requires the development of hybrid quantum-classical models, which strategically combine the strengths of both computational worlds. In such a workflow, classical computers handle data preprocessing and postprocessing tasks, while the quantum processor is tasked with the core computational kernel where its unique capabilities can provide an advantage. Within this hybrid framework, a

BigHPC2025: Special Track on Big Data and High-Performance Computing, co-located with the 4th Italian Conference on Big Data and Data Science, ITADATA2025, September 9 – 11, 2025, Torino, Italy.

[*] Corresponding author.

[†] These authors contributed equally.

mansur.ziiatdinov@unime.it (M. Ziiatdinov); salvatore.distefano@unime.it (S. Distefano)

https://gltronred.info/ (M. Ziiatdinov)

0000-0001-7415-2726 (M. Ziiatdinov); 0000-0003-3448-961X (S. Distefano)

critical but underexplored question arises: How do classical data preparation techniques translate to the quantum domain?

Specifically, techniques like data sampling, which are fundamental for managing class imbalance in classical ML, have an uncertain efficacy when applied to QML. Geometric assumptions that underpin some classical sampling methods may not hold true within the abstract quantum feature space. Such a gap motivates this research, aiming to bridge it by investigating the impact of a foundational sampling technique — random undersampling — on the performance of a hybrid QML classifier for a real-world cybersecurity task. Starting from the well-known UNSW-NB15 network intrusion detection dataset, this paper investigates whether dataset size, balancing, and undersampling techniques are effective strategies for cybersecurity analysis through classical and quantum ML.

The contribution of this work is therefore threefold. First, to provide a methodology and algorithm for hybrid classical-quantum data-driven machine learning workflows. Second, to explore, by a practical, empirical analysis in the cybersecurity domain, the interplay between classical data preprocessing and near-term quantum algorithms. Third, by assessing the performance of a QML model under different conditions of data balance and size, to provide realistic insights into the current capabilities and bottlenecks of classical and quantum computing, as well as their suitability to cybersecurity problems.

To such a purpose, the reminder of the paper is organized as follows: Section 2 discusses about classical and quantum ML overviewing state of the art solutions on data management and balancing, Section 3 proposes a hybrid classical-quantum methodology and algorithm for data-driven ML workflows focusing on data balancing, then applied in Section 4 on a cybersecurity case study from a real dataset to demonstrate its effectiveness. Final discussion and remarks are reported in Section 5 closing the paper.

2. Background and Related Work

Quantum Machine Learning (QML) is an emerging interdisciplinary field that integrates principles from quantum mechanics and machine learning, aiming to develop algorithms that outperform classical counterparts in data processing and analysis [1]. This convergence of quantum physics and artificial intelligence holds the promise of revolutionizing several fields, including healthcare, finance, and materials science, by leveraging quantum properties such as superposition and entanglement to enhance machine learning techniques [2]. QML is attracting rising interest, as demonstrated by the volume of published research, in particular after 2015, witnessing a dramatic increase in scholarly articles and citations [3]. This highlights not only the growing recognition of QML capabilities but also the collaborative efforts among researchers across disciplines, including physics, computer science, and artificial intelligence [4]. Furthermore, bibliometric analyses reveal a dynamic interplay between QML research and its practical applications, especially in areas contributing to technological advancement and intellectual property, such as privacy and cybersecurity [3]. Despite its promising outlook, the field of QML faces significant challenges, including limitations in current quantum hardware and the complexity of integrating quantum algorithms with classical systems [2]. Issues such as qubit decoherence, noise in quantum environments, and the need for effective data encoding techniques remain critical barriers that researchers must address to realize the full potential of QML applications [5]. Additionally, the theoretical foundations of QML, although robust, require further exploration to validate the claimed advantages of quantum algorithms over classical methods, as many existing approaches lack formal evidence of their efficacy [1].

The intersection of classical and quantum machine learning presents a compelling comparative analysis. Recent studies have explored the performance of QML algorithms against classical ML ones across datasets and metrics, investigating their strengths and weaknesses [2]. For example, in [6], a comprehensive analysis involving classical and quantum ML algorithms across different subject systems demonstrated variances in their ability to predict buggy and clean commits, as well as in other software defect predictions. Current research on classical and quantum ML mostly aims at refining quantum-specific solutions while evaluating their generalizability [7], based on the consideration that foundational aspects of QML, such as quantum data encoding and the unique properties of quantum systems, can potentially lead to more efficient processing capabilities and improved outcomes compared to classical ML methods [8].

This stresses the strategic importance of data management and preprocessing in ML, especially for quantum ML [4, 8]. In particular, data balancing is an essential step in ML workflows, addressing the challenges posed by class imbalance in datasets where one class significantly outnumbers another [9]. Such imbalances can lead to biased predictive models that predominantly favor the majority class, thereby misrepresenting the performance of minority classes and potentially compromising the effectiveness of machine learning applications across fields, including healthcare, finance, and cybersecurity [10]. Techniques to balance datasets, such as random undersampling, random oversampling, and the Synthetic Minority Over-sampling Technique (SMOTE), have been developed to enhance model robustness and accuracy by ensuring fair representation of all classes [11]. Even in the realm of quantum machine learning, innovative strategies have emerged that leverage quantum computing principles to improve data balancing techniques. A notable one is QuantumSMOTE, exploiting quantum circuits to generate synthetic data points for minority classes more efficiently than traditional oversampling methods [12, 13].

Despite the promise of these advancements, both classical and quantum data balancing methods face significant challenges. For classical techniques, issues such as overfitting from oversampling and information loss from undersampling persist, necessitating careful evaluation of performance metrics beyond accuracy, such as precision, recall, and F1-score [14, 15]. In the quantum domain, obstacles include the complexity of encoding large datasets into quantum states and the current limitations of quantum hardware, which can impact the feasibility and reliability of implementing large-scale quantum data balancing strategies [16, 17].

Looking ahead, there is significant potential for advancements in data balancing techniques through the integration of classical and quantum computing, paving the way for improved machine learning applications in diverse domains and ultimately contributing to more effective data-driven decision-making. Ultimately, a combined effort to enhance both classical and quantum ML strategies may pave the way for more effective data-driven decision-making in various domains. This paper follow such direction, aiming to investigate data balancing effects on classical and quantum ML algorithm to provide insights on hybrid/combined classical-quantum techniques for data-driven ML workflows.

3. Methodology

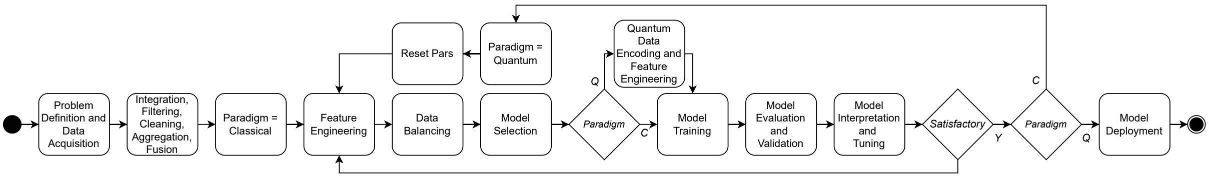
Problem Definition and Data Acquisition
Integration, Filtering, Cleaning, Aggregation, Fusion
Paradigm = Classical
Feature Engineering
Data Balancing
Model Selection
Reset Pars
Paradigm = Quantum
Paradigm
Q
C
Quantum Data Encoding and Feature Engineering
Model Training
Model Evaluation and Validation
Model Interpretation and Tuning
Satisfactory
Y
Paradigm
C
Q
Model Deployment

Figure 1: Proposed hybrid approach for classical-quantum data-driven ML workflows.

The data-driven ML methodology here proposed is described by the algorithm shown in Figure 1 aiming to systematically address the aforementioned data management challenges, particularly in the context of unbalanced datasets, by the integration of classical and quantum computational paradigms. This process starts with *Problem Definition and Data Acquisition*, where the specific challenge is rigorously defined, and relevant datasets are identified and queried to select and retrieve useful data. If such data are not enough for processing, raw data have to be collected by real experiments to complement the dataset, eventually also resorting to synthetic data, as discussed below. These data then undergo a series of critical preprocessing steps, including data *Integration, Filtering, Cleaning, Aggregation, and Fusion.* During this phase, heterogeneous data streams are unified, irrelevant or noisy data points are removed, missing values are addressed, and heterogeneous data are combined to form a coherent, comprehensive dataset suitable for subsequent analysis.

Following the initial data preparation, the workflow diverges based on the computational paradigm selected for model development. At first, a loop is triggered by the preprocessed data *Feature Engineering* for the classical one (*Paradigm=Classical*), where domain expertise and analytical techniques are applied to transform raw data into a set of informative features that enhance the discriminative power for machine learning models. Then, *Data Balancing* techniques are applied to mitigate the inherent class imbalance. This step often involves methods such as undersampling of the majority class, oversampling of the minority class, or synthetic data generation to achieve a more equitable class distribution, which is crucial for preventing model bias and improving the detection of rare, critical events, as detailed below.

Regardless of the chosen paradigm, the prepared data then feeds into *Model Selection*, where appropriate machine learning algorithms — ranging from classical classifiers like Support Vector Machines and Random Forests to quantum-inspired algorithms or hybrid quantum-classical models — are chosen based on the problem characteristics and computational resources. This is followed by *Model Training*, where the selected model learns the underlying patterns from the balanced and engineered datasets. The trained model then undergoes rigorous *Model Evaluation and Validation* using appropriate performance metrics (e.g., precision, recall, F1-score, accuracy, ROC curves) to assess its effectiveness and generalization capabilities on unseen data.

An iterative feedback loop is embedded through *Model Interpretation and Tuning*. Here, the model decisions are analyzed for insights, and hyperparameters are optimized to enhance performance. A *Satisfactory* decision point then determines whether the model performance meets predefined criteria. If unsatisfactory, the process loops back on earlier stages, involving to re-evaluate and re-apply feature engineering and balancing strategies. Once the classical model performance analysis on feature and balancing parameters is satisfactory, based on these experiments and results the quantum one is triggered by switching the paradigm (*Paradigm=Quantum*) and resetting such parameters (*Reset Pars*).

The quantum workflow loop is substantially the same as the classical one except for the data encoding stage required to encode classical data into qubits. The *Quantum Data Encoding and Feature Engineering* step, indeed, involves mapping classical data features onto quantum states, which are then prepared on qubits to be processed by quantum circuits. This encoding is a critical interface between classical data and quantum computation, influencing how information is represented and manipulated in the Hilbert space. While nascent, quantum feature engineering aims to leverage quantum effects to extract novel, potentially more expressive features than classical methods. The quantum loop thus proceeds with quantum model training, evaluation, validation, interpretation and tuning, then iterating on different dataset features and balances for a parametric analysis.

Finally, once both classical and quantum loops are done, the workflow proceeds to *Model Deployment*, where the validated model is integrated into operational systems for live inferences, marking the culmination of the data-driven ML workflow.

3.1. Feature Engineering and Data Balancing

Focusing on classification, a classification problem can be formally defined as follows. Consider a set $\mathbf{X}$ of m tuples in a n-dimensional space: $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \in \mathbb{R}^{m \times n}$, where $\mathbf{x}_i = \{x_{i,1}, \ldots, x_{i,n}\} \in \mathbb{R}^n$ and thus $x_{ij} \in \mathbb{R}$ with $i = 1, \ldots, m$, $j = 1, \ldots, n$, and each data point $\mathbf{x}_i \in \mathbf{X}$ is labeled by the corresponding class $y_i \in \{1, \ldots, c\} \subset \mathbb{N}$. The classification problem asks to predict the $\hat{y} \in \{1, \ldots, c\} \subset \mathbb{N}$ class label for a new, previously unseen $\hat{\mathbf{x}} \in \mathbb{R}^n$ data point. If there are only two classes (i.e. $c = 2$), the problem is called *binary* classification problem.

The set of data points belonging to a particular class $k \in \{1, \ldots, c\} \subset \mathbb{N}$ is denoted by $\mathcal{C}(k) = \{\mathbf{x}_i \in \mathbf{X} \mid y_i = k\}$. Therefore, $\mathcal{C}(k) \cap \mathcal{C}(j) = \emptyset \; \forall k, j \in \{1, \ldots, c\} \subset \mathbb{N}$ with $k \neq j$ and $\bigcup_{k=1}^{c} \mathcal{C}(k) = \mathbf{X}$. The dataset is called *balanced* if different classes have a similar number of data points: $\forall j \neq k : |\mathcal{C}(j)| \approx |\mathcal{C}(k)|$. If the dataset is imbalanced, the smallest class is called the *minority class* and the largest class is called the *majority class.*

There are different approaches to balancing datasets. First, the model can extrapolate additional data points of the minority class (so called *oversampling*), thus increasing its size; second, the model can choose a subset $\mathcal{C}'(k) \subset \mathcal{C}(k)$ for each non-minority class (so called *undersampling*).

This study focuses on the latter approach since current quantum models have limitations in managing large amount of data. The random undersampling method simply selects a random subset of each class. This method allows setting the required number of samples in each class independently, thus controlling the balance of the resulting dataset. In this work, the NearMiss [18] algorithm is adopted as a heuristic algorithm since it includes geometrical criteria into the selection process. Let positive data items be the items belonging to the class to be undersampled and negative data items be the items from the minority class. NearMiss-3 (the version 3 of the NearMiss algorithms family) works in 2 steps: first, for each negative data item, their nearest-neighbors are kept. Then, the positive data items with the largest average distance to the nearest-neighbors are selected.

3.2. Quantum Data Encoding and Model Training

As outlined before, the key difference between classical and quantum paradigms is the data encoding step of the workflow. This study adopts a heuristic `real_amplitudes` encoding $\left|\psi_{\text{qra}}(\mathbf{x}_i)\right\rangle$ from the Qiskit circuit library [19], denoted here as QRA. This technique involves several layers of angle encoding interspersed with entanglement layers. Since it is difficult to express the encoded state in closed form, the corresponding circuit is described. Let the number q of qubits and the number of layers L be such that $n = qL$, where n is the number of features. Then

$$\left|\psi_{\text{qra}}(\mathbf{x}_i)\right\rangle = U_1(\mathbf{x}_i) V U_2(\mathbf{x}_i) V \times \cdots \times V U_L(\mathbf{x}_i) \left|0\right\rangle,$$

where $U_j(\mathbf{x}_i) = \bigotimes_{k=1}^{q} R_{\text{Y}}(x_{i,jq+k})$ rotates the qubits to encode some of the features (i.e. $x_{i,(j-1)q+1}, x_{i,(j-1)q+2}, \ldots, x_{i,(j-1)q+q}$). The entanglement layer V performs CNOT gates between i-th and $i+1$-th qubits for $i \in \{n-2, n-3, \ldots, 1, 0\}$. See Figure 2 for an example.

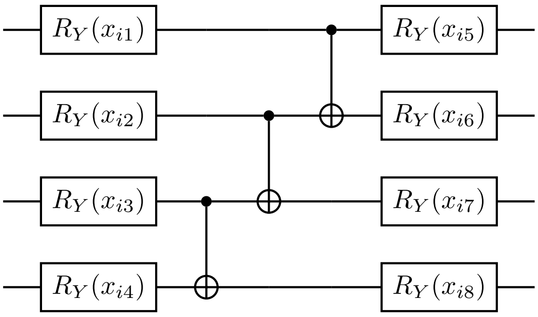

Figure 2: RA circuit for $n = 8, q = 4, L = 2$

After the encoding step, a QML model can be trained. Schuld and Petruccione [20] show that all supervised QML models can be formulated as quantum kernel methods, so let us focus on them. The quantum kernel method uses a quantum computer to compute entries $\kappa(\mathbf{x}, \mathbf{x}') = \left|\langle\psi(\mathbf{x})|\psi(\mathbf{x}')\rangle\right|^2$ of the kernel matrix. Once the entries are calculated, a support vector classifier (SVC) is trained by solving a convex optimization problem:

$$\begin{array}{cc} \underset{(\alpha_1,\ldots,\alpha_m)}{\text{maximize}} & \sum_{i=1}^{m} \alpha_j - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \kappa(\mathbf{x}_i, \mathbf{x}_j), \\ \text{s.t.} & 0 \leq \alpha_j \leq C \text{ and } \sum_{i=1}^{m} \alpha_i y_i = 0. \end{array}$$

The assessment of classification is performed with the standard metrics like accuracy and F_1 score (see, e.g., [21]). Let TP, TN, FP, FN denote, respectively, the number of true positive, true negative, false positive, and false negative predictions. The *accuracy* a is defined as

$$a = \frac{TP + TN}{TP + TN + FP + FN}.$$

The F_1 *score* is the harmonic mean of *precision* prec and *recall* recall:

$$F_1 = 2 \cdot \frac{\text{prec} \cdot \text{recall}}{\text{prec} + \text{recall}}$$

where $\text{prec} = TP/(TP + FP)$ and $\text{recall} = TP/(TP + FN)$.

4. Case Study

4.1. Dataset and Testbed

To demonstrate the effectiveness of the proposed approach, the well-known UNSW-NB15 dataset [22, 23] is taken as benchmark. It focuses on network intrusion detection systems associating with a dataset item the description of a network packet. The UNSW-NB15 dataset has been cleaned from contaminant features in [24]. Once cleaning and encoding the categorical feature as an ordinal, the training dataset contains 175 341 data items with 32 numeric features, while the test dataset contains 82 332 data items with the same features. Each data item is associated with two labels: first, it is labeled as a "normal" packet or as an "attack" packet; second, it is further labeled by different attack types, i.e. "Backdoor", "DoS", "Exploits", etc. (10 classes overall including "normal"). The dataset is highly imbalanced: for example, the smallest class "Worms" contains only 130 data items, while the "Exploits" class contains 33 393 data items.

The experiments focus on the binary classification of a packet between "normal" and "attack" classes. They have been organized in 3 stages:

Experiment 1 establishes a baseline for imbalanced data and classical ML model performance, obtained exploiting the full dataset and the classical SVC with RBF kernel (i.e. $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2)$).

Experiment 2 compares different balancing methods for classical and quantum ML models. Both samplers, implementing the random undersampling method and the NearMiss-3 algorithm, respectively, sample $\ell = 300$ data points per class, 600 overall, which are then used to train classical SVC RBF kernel models and quantum SVC QRA kernel models (with $q = 8, L = 4$ for QRA-8-4 and $q = 4, L = 8$ for QRA-4-8). The QRA parameters have been selected to ensure that all $32 = 8 \times 4$ features could be embedded.

Experiment 3 explores how balancing affects the performance of different classical and quantum models. In each run of the experiment, the random sampler, demonstrated to be more effective than NearMiss-3 by Experiment 2, selects $\ell = 50, 100, 200, 300, 400, 500$ data points *per class*, and these subsets are exploited to train classical RBF and quantum QRA-4-8 and QRA-8-4 SVC kernel models. The choice of the parameter $\ell \leq 500$ is due to hardware limitations on simulations.

All of the experiments follow the same design principles. Each experiment performs 10 runs, and the reported results are statistics obtained by these runs. At each run, the features are scaled using `MinMaxScaler`, i.e. each data point $\mathbf{x}_i = (x_{i,1}, \ldots, x_{i,m})$ is mapped to the vector $\mathbf{z}_i = (z_{i,1}, \ldots, z_{i,m})$, where $z_{ij} = (x_{ij} - \min_{i=1,\ldots,n} x_{ij}) / \max_{i=1,\ldots,n} x_{ij}$, $i = 1, \ldots, n$ and $j = 1, \ldots, m$. Then, if necessary, the sampler chooses ℓ data points for each class (if the class contains less than ℓ data points, all these points are selected, but no new points are generated). Finally, the SVC is trained on this subset of the training data and is evaluated on the balanced subset of the test data containing 100 points per class.

All tests have been performed on a machine with the following characteristics. CPU: AMD Ryzen 9 5950X, RAM: 64 GiB, OS: Linux, kernel: 6.6.74-gentoo, Python: 3.12.9, jupyter-core: 5.7.2, numpy: 2.2.2, qiskit: 1.3.2, qiskit-machine-learning: 0.8.2, scikit-learn: 1.6.1. The source code is available in the repository [25].

4.2. Results

Experiment 1. The results of the Experiment 1, the baseline classical case, are reported in Table 1. Although more sophisticated classical methods can achieve better results, this study aims at investigating the effects of balancing comparing similar classical and quantum methods, i.e. kernel-based SVC.

Table 1
Baseline for classical case without undersampling

Metric	Value
avg. f1 score	0.749577
avg. precision	0.791891
avg. recall	0.757500
avg. specificity	0.757500

Experiment 2. The results of the Experiment 2 are shown in Figure 3. The random sampler shows better f1 score performance than the NearMiss-3 sampler for both classical and quantum models, with $\sim 60\%$ of improvement on the classical model and $\sim 10\%$ for quantum ones. Thus, despite worse than random sampler, the NearMiss-3 one works better in quantum models.

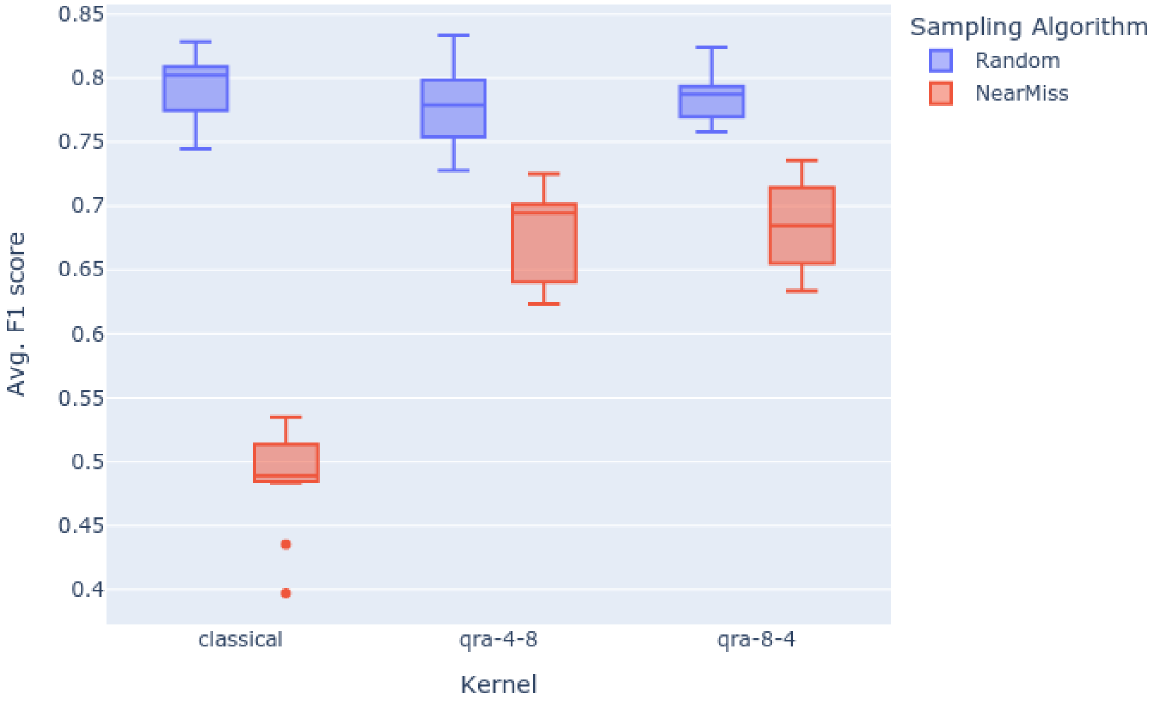

Figure 3: F1 score (higher is better) for different balancing methods.

Experiment 3. The main findings of Experiment 3 are shown in Figure 4. All the model graph shapes are similar: in the beginning, the learning curve is increasing till a maximum value and then decreases, likely due to overfitting. The maximum values for quantum and classical models are close, but do not coincide: the classical model and the quantum model with 4 qubits (QRA-4-8) have their maximum at around $\ell = 300$ and the quantum model with 8 qubits (QRA-8-4) has the maximum at around $\ell = 400$. The quantum models also show evidence of better generalization: they outperform the classical ones for smaller number of data items ($\ell \leq 100$) ($\sim 5\%$ better with majority class size 50).

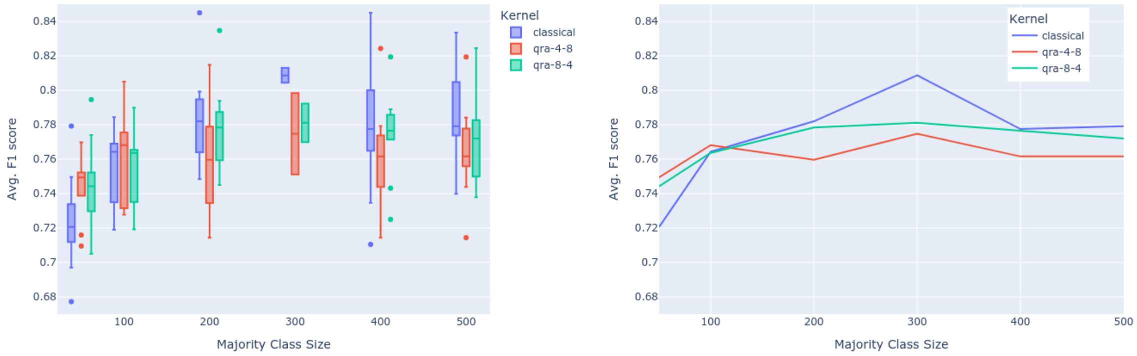

Figure 4: F1 score (higher is better) for quantum and classical training methods after balancing: the left shows the distribution of values with boxplots; the right shows the median values.

5. Discussion and Conclusions

This work focused on investigating how the dataset size and balance can affect on different ML models, both classical and quantum. The investigation on a real and well known cybersecurity dataset (UNSW-NB15) led to three primary findings. First, the use of random undersampling proved to be an effective technique for preprocessing the imbalanced UNSW-NB15 dataset for both classical and quantum classification ML models, suggesting that established classical techniques are valuable in hybrid ML workflows. Second, a strong relationship between the dataset size, despite balanced, and model performance has been observed. Both classical and quantum ML models show concave learning curve, index of overfitting issues. This pushes ML experts and practitioners to find out the optimal majority class size for which the ML model provides best performance. Third, the quantum learning models has demonstrated accelerated learning curves compared to classical ones, especially much better performance with small datasets, while classical model performance maximum outperforms quantum ones. This suggests hybrid solutions exploiting quantum ML models to address data scarcity or as bootstrap models in the beginning of a data acquisition campaign, then switching to classical ones when enough data are available. These findings collectively indicate that while quantum approaches offer potential efficiencies, their practical application is currently dependent on classical preprocessing and advances in quantum hardware fault tolerance.

Acknowledgments

We acknowledge financial support under the National Recovery and Resilience Plan (PNRR), Mission 4, Component 2, Investment 1.4, Call for tender No. 1031 published on 17/06/2022 by the Italian Ministry of University and Research (MUR), funded by the European Union – NextGenerationEU, Project Title "National Centre for HPC, Big Data and Quantum Computing (HPC)" – Code National Center CN00000013 – CUP D43C22001240001.

Declaration on Generative AI

During the preparation of this work, the author(s) used LLM tools in order to: Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication content.

References

[1] Y. Wang, J. Liu, A comprehensive review of quantum machine learning: from nisq to fault tolerance, Reports on Progress in Physics 87 (2024) 116402. URL: http://dx.doi.org/10.1088/1361-6633/ad7f69. doi:10.1088/1361-6633/ad7f69.

[2] S. Tomar, R. Tripathi, S. Kumar, Comprehensive survey of qml: From data analysis to algorithmic advancements, 2025. URL: https://arxiv.org/abs/2501.09528. arXiv:2501.09528.

[3] R. Bansal, N. K. Rajput, Quantum machine learning: Unveiling trends, impacts through bibliometric analysis, 2025. URL: https://arxiv.org/abs/2504.07726. arXiv:2504.07726.

[4] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, J. R. McClean, Power of data in quantum machine learning, Nature communications 12 (2021) 2631.

[5] J. Lopez, Quantum machine learning: Applications, algorithms, and hardware challenges, International Journal of AI, BigData, Computational and Management Studies 5 (2024) 1–13.

[6] M. Nadim, M. Hassan, A. K. Mandal, C. K. Roy, Quantum vs. classical machine learning algorithms for software defect prediction: Challenges and opportunities, arXiv preprint arXiv:2412.07698 (2024).

[7] B. Khanal, P. Rivas, A. Sanjel, K. Sooksatra, E. Quevedo, A. Rodriguez, Generalization error bound for quantum machine learning in nisq era—a survey, Quantum Machine Intelligence 6 (2024) 1–20.

[8] M. Rath, H. Date, Quantum data encoding: A comparative analysis of classical-to-quantum mapping techniques and their impact on machine learning accuracy, EPJ Quantum Technology 11 (2024) 72.

[9] G. Parmar, R. Gupta, T. Bhatt, G. Sahani, B. Y. Panchal, H. Patel, A review on data balancing techniques and machine learning methods, in: 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT), IEEE, 2023, pp. 1004–1008.

[10] B. Yousefimehr, M. Ghatee, M. A. Seifi, J. Fazli, S. Tavakoli, Z. Rafei, S. Ghaffari, A. Nikahd, M. R. Gandomani, A. Orouji, R. M. Kashani, S. Heshmati, N. S. Mousavi, Data balancing strategies: A survey of resampling and augmentation methods, 2025. URL: https://arxiv.org/abs/2505.13518. arXiv:2505.13518.

[11] P. Mooijman, C. Catal, B. Tekinerdogan, A. Lommen, M. Blokland, The effects of data balancing approaches: A case study, Applied Soft Computing 132 (2023) 109853.

[12] N. Mohanty, B. K. Behera, C. Ferrie, P. Dash, A quantum approach to synthetic minority oversampling technique (smote), Quantum Machine Intelligence 7 (2025) 38.

[13] J. Jiang, C. Zhang, L. Ke, N. Hayes, Y. Zhu, H. Qiu, B. Zhang, T. Zhou, G.-W. Wei, A review of machine learning methods for imbalanced data challenges in chemistry, Chemical Science (2025).

[14] J. Mendoza, J. Mycroft, L. Milbury, N. Kahani, J. Jaskolka, On the effectiveness of data balancing techniques in the context of ml-based test case prioritization, in: Proceedings of the 18th International Conference on Predictive Models and Data Analytics in Software Engineering, 2022, pp. 72–81.

[15] H. Kaur, H. S. Pannu, A. K. Malhi, A systematic review on imbalanced data challenges in machine learning: Applications and solutions, ACM computing surveys (CSUR) 52 (2019) 1–36.

[16] N. Mohanty, B. K. Behera, C. Ferrie, Quantum smote with angular outliers: Redefining minority class handling, 2025. URL: https://arxiv.org/abs/2501.19001. arXiv:2501.19001.

[17] S. Kwon, J. Huh, S. J. Kwon, S.-h. Choi, O. Kwon, Leveraging quantum machine learning to address class imbalance: A novel approach for enhanced predictive accuracy, Symmetry 17 (2025). URL: https://www.mdpi.com/2073-8994/17/2/186. doi:10.3390/sym17020186.

[18] I. Mani, I. Zhang, knn approach to unbalanced data distributions: a case study involving information extraction, in: Proceedings of workshop on learning from imbalanced datasets, volume 126, ICML United States, 2003, pp. 1–7.

[19] M. Treinish, et al., Qiskit/qiskit: Qiskit 1.4.3, 2025. doi:10.5281/zenodo.15374661.

[20] M. Schuld, F. Petruccione, Machine learning with quantum computers, volume 676, Springer, 2021.

[21] A. C. Müller, S. Guido, Introduction to machine learning with Python: a guide for data scientists, " O'Reilly Media, Inc.", 2016.

[22] N. Moustafa, J. Slay, Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set), in: 2015 Military Communications and Information Systems Conference (MilCIS), 2015, pp. 1–6. doi:10.1109/MilCIS.2015.7348942.

[23] N. Moustafa, J. Slay, Unsw-nb15, 2024. doi:10.34740/KAGGLE/DSV/9350725.

[24] L. D'Hooge, M. Verkerken, T. Wauters, B. Volckaert, F. De Turck, Discovering non-metadata contaminant features in intrusion detection datasets, in: 2022 19th Annual International Conference on Privacy, Security & Trust (PST), 2022, pp. 1–11. doi:10.1109/PST55820.2022.9851974.

[25] M. Ziiatdinov, S. Distefano, Cyber-risk case study, https://github.com/usce2qc/notebooks?tab=readme-ov-file#cybersecurity, 2025.