Quantum Support Vector Regression for Robust Anomaly Detection

Kilian Tscharke, Maximilian Wendlinger, Sebastian Issel, Pascal Debus
Fraunhofer Institute for Applied and Integrated Security (AISEC), Garching near Munich, Germany
{firstname.lastname}@aisec.fraunhofer.de

***Abstract*—Anomaly Detection (AD) is critical in data analysis, particularly within the domain of IT security. In recent years, Machine Learning (ML) algorithms have emerged as a powerful tool for AD in large-scale data. In this study, we explore the potential of quantum ML approaches, specifically quantum kernel methods, for the application to robust AD. We build upon previous work on Quantum Support Vector Regression (QSVR) for semisupervised AD by conducting a comprehensive benchmark on IBM quantum hardware using eleven datasets. Our results demonstrate that QSVR achieves strong classification performance and even outperforms the noiseless simulation on two of these datasets. Moreover, we investigate the influence of – in the NISQ-era inevitable – quantum noise on the performance of the QSVR. Our findings reveal that the model exhibits robustness to depolarizing, phase damping, phase flip, and bit flip noise, while amplitude damping and miscalibration noise prove to be more disruptive. Finally, we explore the domain of Quantum Adversarial Machine Learning and demonstrate that QSVR is highly vulnerable to adversarial attacks and that noise does not improve the adversarial robustness of the model.**

***Index Terms*—benchmark, semisupervised learning, noise, hardware, adversarial attacks, robustness, quantum kernel methods, quantum machine learning**

I. Introduction

Quantum Machine Learning (QML) merges Quantum Computing (QC) and Machine Learning (ML) to exploit potential advantages of QC for ML. Recently, Quantum Kernel Methods (QKMs) have gained attention for their potential to replace many supervised QML models and their ability to surpass variational circuits, as evidenced by Schuld [1]. Theoretical findings further prove that QKMs can potentially address classification problems intractable for classical ML, such as the discrete logarithm problem [2].

Anomaly Detection (AD) is crucial in the realm of IT security, as it identifies deviations from normal patterns in areas like network intrusion and fraud detection [3]. However, it is important to note that ML models employed in security-sensitive contexts are susceptible to adversarial attacks, where small, carefully crafted perturbations in inputs can result in misclassification. (Quantum) Adversarial Machine Learning (QAML/AML) explores techniques for both generating these adversarial attacks and defending against them.

Given the potential of QML to address problems challenging for classical methods, the application of QML to AD is a tempting progression. In 2023, Tscharke et al. [4] proposed a semisupervised AD approach based on the reconstruction loss of a Quantum Support Vector Regression (QSVR) equipped with a quantum kernel. The authors compared the performance of the QSVR against a Quantum Autoencoder (QAE), a classical Support Vector Regression (CSVR) with a Radial Basis Function (RBF) kernel, and a classical autoencoder (CAE), using ten real-world and one synthetic dataset. Their simulated QSVR demonstrated comparable performance to CSVR, with marginal superiority over the other models. However, the implementation of their QSVR on hardware was left open, a gap tackled by this paper.

In today's Noisy Intermediate-Scale Quantum (NISQ) era, noise limits the application of QC for industrial tasks, underscoring the need for a comprehensive understanding of how noise affects quantum algorithms. Noise impacts QML models by affecting predictive performance and, in QAML, altering robustness against adversarial attacks [5], [6].

The remainder of this work is structured as follows: The next subsection I-A offers a comprehensive review of research related to QKMs, the impact of noise on QML, and QAML. Our contributions are detailed in subsection I-B. The following Background (section II) provides a foundation for understanding QKMs, noisy QC, and adversarial attacks. Next, the Methods (section III) describe the implementation of the QSVR on hardware, the noise simulation, and the adversarial attacks generation. In Results and Discussion (section IV), we analyze the results of the hardware experiments and investigate the influence of noise and adversarial attacks on the model's performance. Finally, Conclusion and Outlook (section V) highlights the key results of this work and provides future research directions.

A. Related Work

In 2019, Havlicek et al. [7] introduced a quantum Support Vector Machine (QSVM) for binary classification on a two-qubit NISQ device. Since then, QSVMs have been applied to many areas, including remote sensing image classification [8], mental health treatment prediction [9], and breast cancer prediction [10]. Kyriienko and Magnusson [11] extended this to unsupervised fraud detection with a simulated one-class QSVM, and Tscharke et al. [4] developed a QSVR for semisupervised AD in 2023. However, to the best of our knowledge, a QSVR for semisupervised AD has not yet been set up on hardware.

Research has also explored the influence of noise in QML models [12]–[14], focusing on noise robustness [15], [16] or beneficial use of noise [17], [18]. However, as far as we know, the influence of noise on a QSVR for semisupervised AD has not yet been evaluated.

Finally, the link between quantum noise and QAML was established by Du et al. [5] in 2021, who found that adding depolarization noise can increase adversarial robustness. Building on this, Huang and Zhang [6] improved the adversarial robustness of Quantum Neural Networks by adding noise layers in 2023. To date, there have been no published results involving adversarial attacks on semisupervised QML models for AD.

B. Contributions

The goal of our work is to gain further insight into the potential of QSVR for semisupervised AD in the NISQ era, which we accomplish through these contributions:

1) We investigate how the model performs on hardware and report that the QSVR achieves good classification performance on a 27-qubit IBM device, even outperforming a noiseless simulation on two out of ten real-world datasets.
2) We show that the QSVR is largely robust to noise by training and evaluating over 500 noisy models. We further observe that *amplitude damping* and *miscalibration* have the most damaging effect on the model's performance and that the artificial *Toy* dataset, constructed to be linearly separable, suffers the most from noise.
3) Finally, we investigate the robustness of the QSVR against adversarial attacks and find it highly vulnerable, with performance on real-world datasets dropping by up to an order of magnitude for a weak attack strength of $\varepsilon = 0.01$. Introducing noise into the QSVR does not clearly improve the model's adversarial robustness.

II. Background

A. Quantum Kernel Methods

A kernel is a positive, semidefinite function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{C}$ on the input set $\mathcal{X}$. It uses a distance measure $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ between two input vectors $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$ to create a model that captures the properties of a data distribution. A *feature map* $\phi : \mathcal{X} \to \mathcal{F}$ maps input vectors $\boldsymbol{x}$ to a Hilbert or *feature space* $\mathcal{F}$. They are of great importance in ML, as they map input data in a higher-dimensional space with a well-defined metric. The feature map can be a nonlinear function that changes the relative position of the data points. As a result, the dataset can become easier to classify in feature space, and even linearly separable. We associate feature maps with kernels by defining a kernel via

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) := \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle_{\mathcal{F}} \tag{1}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{F}}$ is the inner product defined on $\mathcal{F}$.

With the exponentially large Hilbert space of QCs, the use of QKMs for ML is close at hand. A quantum feature map $\psi : \boldsymbol{x} \to |\phi(\boldsymbol{x})\rangle$ is implemented via a feature-embedding circuit $U(\boldsymbol{x})$, which acts on a ground state $|0 \ldots 0\rangle$ of a Hilbert space $\mathcal{F}$ as $|\phi(\boldsymbol{x})\rangle = U(\boldsymbol{x}) |0 \ldots 0\rangle$. The distance measure in the quantum kernel is the absolute square of the inner product of the quantum states. On hardware, this can be realized by the *inversion test*, where a sample $\boldsymbol{x}_i$ is encoded in the unitary U, followed by the adjoint $U^\dagger$ encoding the second sample $\boldsymbol{x}_j$ and measuring the probability of the all-zero state. Thus, the quantum kernel is defined as

$$\begin{aligned} \kappa(x_i, x_j) &= |\langle \phi(\boldsymbol{x}_i) | \phi(\boldsymbol{x}_j) \rangle|^2 \\ &= \left| \langle 0^{\otimes n} | \, U^\dagger(\boldsymbol{x}_i) U(\boldsymbol{x}_j) \, |0^{\otimes n} \rangle \right|^2 \end{aligned} \tag{2}$$

and returns the *overlap* or *fidelity* of the two states. A more in-depth description of quantum kernels can be found in [19], [20].

There exist many different encoding techniques for the circuit realizing the quantum feature map, but for this work, we will focus on *angle encoding* because of its advantageous complexity with respect to the number of gates $\mathcal{O}(k)$ for an input vector $\mathbf{x}$ of dimension k. *Angle encoding* is a special form of time-evolution encoding, where a scalar value $\boldsymbol{x}$ is encoded in the unitary evolution of a quantum system governed by a Hamiltonian H. The unitary of time-evolution encoding is given by

$$U(x) = e^{-i\boldsymbol{x}H}. \tag{3}$$

In the case of *angle encoding*, the Pauli matrices σ_a with $a \in \{x, y, z\}$ are used in the Hamiltonian $H = \frac{1}{2}\sigma_a$.

B. Noisy Quantum Computing

Real quantum systems are never completely isolated from the environment; for example, an electron realizing a qubit will interact with other charged particles. Moreover, quantum computers are programmed by an external system and thus can never be a closed system [21], [22].

In the current NISQ-era, noise significantly limits the performance of quantum algorithms, primarily through *coherent* and *incoherent* noise. *Coherent* noise arises from systematic, reversible errors that lead to predictable but undesired evolution of the states, often caused by imperfect calibrations or imprecise control signals [23]. *Coherent* noise is an *unitary* evolution of the system, characterized by having only one operation element in the *operator-sum representation* introduced below [24], [25].

Incoherent noise, on the other hand, is characterized by random, stochastic processes caused by insufficient isolation of the system from its environment. These uncontrolled interactions between system and environment lead to deviations between the desired and the actual evolution and to a loss of coherence in the system [22], [23].

Quantum noise can be modeled by a quantum channel, where the term ”channel” is drawn from classical information theory [21]. In the *operator-sum representation*, a channel is described by the map $\mathcal{E}$ with operation elements (or *Kraus operators*) $\{E_i\}$ mapping the density operator $\rho = |\psi\rangle \langle\psi|$ to another density operator $\mathcal{E}(\rho)$.

$$\mathcal{E}(\rho) = \sum_i E_i \rho E_i^\dagger. \tag{4}$$

In the following, selected types of single-qubit noisy channels are described, and their operation elements E_i are listed. For a more detailed explanation of the noisy quantum channels, we refer to [21], [24], [26].

1) *Amplitude Damping* channel: describes the effect of energy loss, such as when an atom emits a photon. The channel acts on the quantum system A and the environment E as follows: if both A and E are in their ground state $|0\rangle$, nothing happens. If A is in the excited state $|1\rangle_A$, a photon will be emitted with probability p, leading to the excitation of the environment and causing the transition $|0\rangle_E \to |1\rangle_E$, while A drops to the ground state, i.e. $|1\rangle_A \to |0\rangle_A$. The evolution caused by the channel can be summarized as:

$$\begin{aligned} &|0\rangle_A \otimes |0\rangle_E \mapsto |0\rangle_A \otimes |0\rangle_E \\ &|1\rangle_A \otimes |0\rangle_E \mapsto \sqrt{1-p}|1\rangle_A \otimes |0\rangle_E + \sqrt{p}|0\rangle_A \otimes |1\rangle_E \end{aligned} \tag{5}$$

This is achieved by the operation elements:

$$\begin{aligned} E_0 &= \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{bmatrix} \\ E_1 &= \begin{bmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{bmatrix} \end{aligned} \tag{6}$$

2) *Bitflip* channel: flips the state of a qubit from $|0\rangle$ to $|1\rangle$ and vice versa with probability p. The operators are:

$$\begin{aligned} E_0 = \sqrt{1-p}I &= \sqrt{1-p}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ E_1 = \sqrt{p}X &= \sqrt{p}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{aligned} \tag{7}$$

3) *Depolarizing* channel: the qubit remains intact with probability $1-p$, while an error occurs with probability p. If an error occurs, the state is replaced by a uniform ensemble of the three states $X|\psi\rangle, Y|\psi\rangle, Z|\psi\rangle$. This is a symmetric decoherence channel defined by operation elements:

$$\begin{aligned} E_0 = \sqrt{1-p}I \quad &= \sqrt{1-p}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ X\text{-Error: } E_1 = \sqrt{p/3}X \quad &= \sqrt{p/3}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ Y\text{-Error: } E_2 = \sqrt{p/3}Y \quad &= \sqrt{p/3}\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \\ Z\text{-Error: } E_3 = \sqrt{p/3}Z \quad &= \sqrt{p/3}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned} \tag{8}$$

4) *Miscalibration* channel: a coherent noise channel applying an "overrotation" p to the R_a rotation gate with $a \in \{x, y, z\}$. This can be caused by an imperfect calibration of the device [23]. Since the channel is unitary, it has only one operation element:

$$E_0 = R_a(p). \tag{9}$$

5) *Phase Damping* channel: the *phase damping* or *dephasing* channel models the effect of random environmental scattering on a qubit, such as photon interactions in a waveguide or atomic states perturbed by distant charges. This channel causes a partial loss of phase information without energy loss. It produces the same effect as the *phase flip* channel, with the phase damping λ related to the phase flip probability p by

$$p = \frac{1-\sqrt{1-\lambda}}{2}. \tag{10}$$

6) *Phaseflip* channel: applies a phase of -1 to the $|1\rangle$-state with probability p and leaves the $|0\rangle$-state unchanged. It has the operation elements:

$$\begin{aligned} E_0 = \sqrt{1-p}I &= \sqrt{1-p}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ E_1 = \sqrt{p}Z &= \sqrt{p}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned} \tag{11}$$

The channel has the following effect on the state's density matrix ρ:

$$\mathcal{E}\begin{pmatrix} \rho_{00} & \rho_{01} \\ \rho_{10} & \rho_{11} \end{pmatrix} = \begin{pmatrix} \rho_{00} & (1-2p)\rho_{01} \\ (1-2p)\rho_{10} & \rho_{11} \end{pmatrix} \tag{12}$$

From this, we can see that the phase flip channel destroys superposition by decaying the off-diagonal terms of the density matrix ρ while the on-diagonal terms remain invariant.

C. Adversarial Attacks

QML models are typically trained using a hybrid quantum-classical approach. In this framework, the model parameters are optimized using a classical optimization algorithm, while the quantum part is limited to the evaluation of the loss, which is (partially) done by the quantum computer or simulator. This hybrid approach allows us to easily extend the concept of adversarial attacks from classical ML to QML, which has already been successfully demonstrated in [27].

An adversarial attack is a small, carefully crafted perturbation of the k-dimensional input $\boldsymbol{x}$ that causes the model to misclassify the input [28], [29]. An untargeted adversarial sample is created by maximizing the model's loss $\mathcal{L}$ while keeping the perturbation δ small enough to be imperceptible to humans, e.g., by ensuring $\delta \in \Delta = \{\delta \in \mathbb{R}^k : \|\delta\|_\infty \leq \varepsilon\}$ for some small ε. In general, the ideal perturbation is given by

$$\delta \equiv \underset{\delta' \in \Delta}{\operatorname{argmax}}\, \mathcal{L}\left(f\left(x+\delta'; \theta^*\right), y\right), \tag{13}$$

where f is the model, θ^* are the model's optimized parameters after training, and y is the target.

One of the most widely used methods for generating adversarial samples is Projected Gradient Descent (PGD) [30]. PGD iteratively maximizes the model's prediction error while ensuring that the perturbation remains within a predefined range. This approach has become a standard tool for evaluating the robustness of models to adversarial threats. The perturbed input is determined by

$$\boldsymbol{x}^{t+1} = \Pi_{\boldsymbol{x}+S}\left(\boldsymbol{x}^t + \alpha\,\mathrm{sgn}\left(\nabla_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x}^t, y; \theta^*)\right)\right), \tag{14}$$

where $\boldsymbol{x}^t$ represents the perturbed data at step t, $\Pi_{\boldsymbol{x}+S}$ clips the perturbed data into the range of the normalized input set S, and α is the step size.

A straightforward strategy to increase the adversarial robustness is *adversarial training*, where adversarial samples are included in the training set.

III. Methods

The experiments performed in this work are threefold. First, we benchmarked the QSVR for semi-supervised AD on hardware. Second, we evaluated the influence of noise on the classification performance of the QSVR, and third, we investigated the influence of noise on the adversarial robustness of the QSVR. An overview of the datasets used in the experiments is given in Table VI in Section A in the appendix. The datasets were reduced to five dimensions using Principal Components Analysis. For a more detailed description of the model, the datasets, and the preprocessing techniques, we refer to [4].

A. Quantum Support Vector Regression Model

The QSVR for semi-supervised AD is described in detail in [4] , and the kernel circuit is displayed in Fig. 1. The first data point $\boldsymbol{x}_i$ is encoded by the unitary U, followed by its inverse $U^\dagger$ encoding the second data point $\boldsymbol{x}_j$. The unitary U consists of a layer of RZ gates and a layer of RX gates, followed by a layer of IsingZZ[1] gates to create entanglement. Each of the single-qubit gates encodes one feature, while the parameter of the IsingZZ gate is a product of two features. The kernel entry $K_{ij} = K_{ji}$ is obtained by measuring the probability of the all-zero state after applying both unitaries.

B. Hardware Experiments

In the hardware experiments, we used a training set of size 30 from the normal class and a test set of size 50 with equal class ratio, following [4].

1) Device Specifications: The experiments were performed on the IBM System One in Ehningen, Germany, in January 2024. The system is a 27-qubit superconducting quantum computer with a quantum volume of 64. The QSVR was implemented using qiskit [31] with default error mitigation techniques. Further specifications of the system are listed in table I.

TABLE I
Specification of the IBM System One Ehningen.

Spec	Value
Name	IBM Quantum System One at Ehningen
System type	Superconducting
Number of qubits	27
Quantum Volume	64
Processor type	Falcon
Deployment year	2021
Coherence time	$\approx 150\mu$s
Single qubit error	$\approx 0.025\%$
Two qubit gate error	$\approx 0.7\%$
Operation time of 2 qubit gate	$\approx$ 300ns for CNOT

2) Reference Models: We benchmarked our model against four other models, following the approach in [4]. The models include a simulated QSVR, a simulated quantum autoencoder based on [32], a classical SVR, and a classical autoencoder.

C. Generation of Noise

The influence of noise on the QSVR for semisupervised AD was evaluated by applying six noise channels with different strengths to the quantum circuit that computes the kernel. The seven noise probabilities used in the experiments are $p \in \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$ and the five noise channels are [*bitflip*, *phaseflip*, *depolarizing*, *phase damping*, *amplitude damping*]. This leads to a total of $7 \cdot 5 = 35$ models per dataset. For the *miscalibration* channel, the noise probability p is the overrotation in radians in 20 linear steps between 0 and 2π. For the DoH dataset subject to adversarial attacks of strength $\varepsilon = 0.1$, additional evaluations were performed in the region close to $p = \pi$. The noisy QSVR was simulated using PennyLane [33]. We used a training set of size 100 from the normal class and a test set of size 100 with a balanced class ratio.

D. Generation of Adversarial Attacks

We created 100 adversarial samples of the test set (50 from each class) using PGD with the parameters listed in Table II. The attacks targeted the noiseless models and were then applied to the noisy models. For adversarial training, we create adversarial samples of the training set and train the model using the adversarial training set of size 100.

TABLE II
Overview of the parameters used in the PGD attacks.

Spec	Values
Attack strength ε	[0.01, 0.1, 0.3]
Iterations n	50
α	ε/n

IV. Results and Discussion

In this study, we first benchmarked our QSVR on the 27-qubit IBM Ehningen device (labeled *qc-QSVR*) and compared its performance against the simulated quantum baseline models QSVR (simulated version of our model, see [4]) and QAE (based on [32]), as well as CSVR and CAE as classical baselines. Second, six different noise channels of varying strength were introduced to evaluate the influence of noise on the QSVR algorithm. Third, the adversarial robustness of the model was examined, and the influence of noise on the adversarial robustness was evaluated by exposing the (noisy) models to adversarial attacks. The simulations show no error bars because the SVR is a deterministic model and pennylane's *default.mixed*[2] device used to calculate the kernels computes exact outputs.

[1] https://docs.pennylane.ai/en/stable/code/api/pennylane.IsingZZ.html

[2] https://docs.pennylane.ai/en/stable/code/api/pennylane.devices.default_mixed.DefaultMixed.html

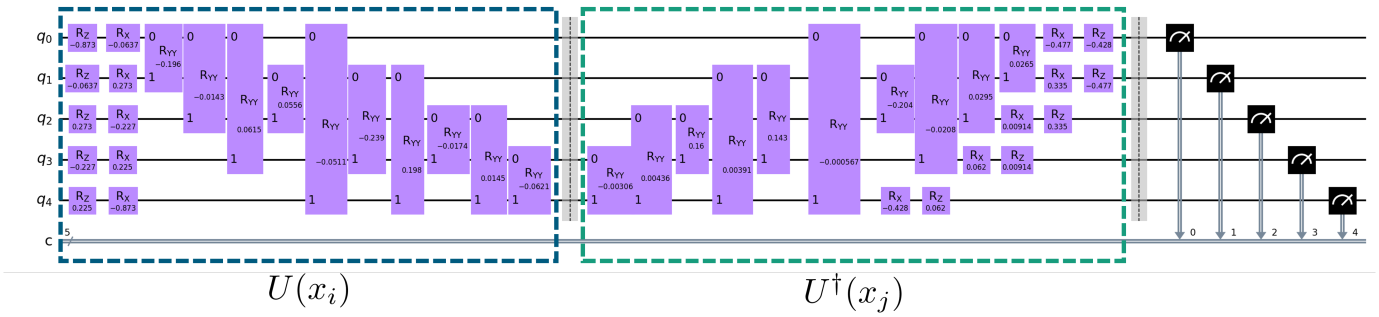

Fig. 1. QSVR circuit. Explanation in text.

A. Hardware Results

Figure 2 compares model performance on eleven datasets using area under the ROC curve (AUC), a commonly used metric in AD that measures the trade-off between the true and false positive rates independent of a detection threshold. An AUC of 1.0 indicates perfect classification of the dataset, and a random classifier achieves an AUC of 0.5 on balanced binary datasets. The datasets include Credit Card Fraud (CC), Census, Forest Cover Type (CoverT), Domain Name System over HTTPS (DoH), EMNIST, Fashion MNIST (FMNIST), Network Intrusion (KDD), MNIST, Mammography (Mammo), URL, and our constructed dataset Toy. The models included in the study are qc-QSVR (ours), QSVR, QAE, CSVR, and CAE. The average performance of each model across all datasets is represented by a dotted line, while the bars indicate the models' performance on individual datasets.

On average, our qc-QSVR exhibits an AUC decrease of 0.04 compared to the simulated QSVR. In 8 out of 11 datasets, the simulated model outperforms the qc-QSVR, with the performance gap explained by hardware noise. On the DoH datasets, both models perform identically, while on the CC and KDD datasets, the qc-QSVR surprisingly outperforms its simulated counterpart. On these two datasets, hardware-induced noise appears to enhance model performance, an effect we attribute to improved generalization. Specifically, the perturbations introduced by the noisy gates mimic the corruptions applied in denoising autoencoders, a technique shown to yield superior generalization compared to standard autoencoders [34]. Our results are consistent with those of other authors who observe that noise can improve the performance of QML models under certain circumstances [35], [36].

B. Influence of Noise on the Model Performance

The hardware results show that noise in NISQ devices affects the performance of QSVR. Therefore, in this section, we examine the influence of six noise channels on the QSVR based on the performance on eleven datasets. Figure 3 shows the AUC of noisy simulations with the noise channels described in Section II-B as well as the influence of adversarial attacks of strength $\varepsilon = 0.1$ on the noisy simulations. The model's robustness against noise is now analyzed, and the adversarial robustness is investigated in Section IV-C. The QSVR is largely

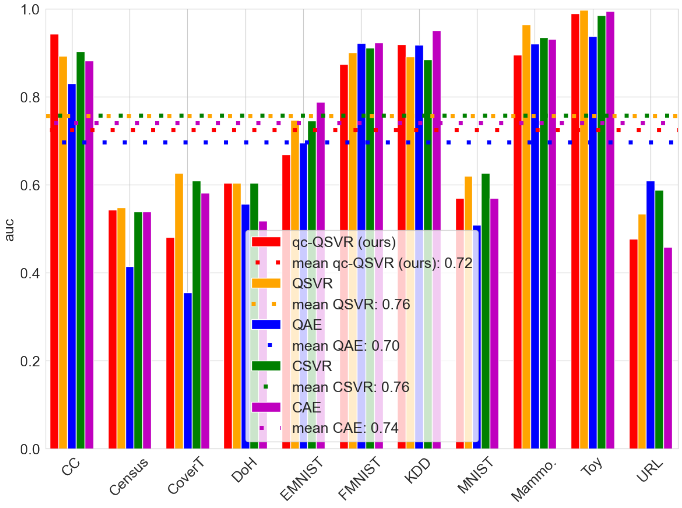

Fig. 2. Plot of area under the curve (AUC) for the different models on the evaluated data sets.

robust against *depolarizing*, *phase damping*, *phaseflip* , and *bitflip* noise, as the AUC for these noise types remains rather stable with increasing noise probability. A theoretical analysis of the robustness of quantum classifiers done by LaRose and Coyle [37] proves that single-qubit classifiers are robust against precisely these noise channels. Taking into account the findings of Schuld [1] that many short-term and fault-tolerant quantum models can be replaced by a general support vector machine whose kernel computes distances between data-encoding quantum states, we expect these results to transfer to our QSVR.

Amplitude damping has a large effect on the model's behavior as the AUC decreases for all datasets except CoverT and URL at $p = 0.1$ and $p = 0.2$. At higher p, the AUC partially recovers, approaching 0.5. This shows that the QSVR is very sensitive to *amplitude damping* , and at high noise levels the model becomes a random classifier. These results are also supported by LaRose and Coyle's [37] analysis, who found that quantum classifiers are generally not robust against *amplitude damping* noise.

The curves of the models subject to *miscalibration* noise show a periodicity of approximately π, with dips at about

$k\pi$ for $k = 0, 1, 2$, and plateaus in between. Small levels of *miscalibration* degrade the performance of the model, but when the noise level is above about 0.25π , the AUC reaches a plateau at a level similar to the one for zero noise until the curve dips again around $p = \pi$. We conclude that small degrees of *miscalibration* reduce model performance, and that this type of noise should be avoided in hardware.

C. Adversarial Robustness

AD is often used in security-critical areas such as credit card fraud detection or network intrusion detection. Therefore, ML models used for AD must be robust to adversarial attacks. For low-dimensional, tabular data sets, it is possible that a sample can be completely and effectively transformed into a sample of a different class at high attack strengths. In these cases, the effect on the AUC may be exaggerated and should be interpreted only as an upper bound on the performance drop.

1) Noise-Free Adversarial Attacks: First, we consider the noise-free model and plot the obtained results of the PGD attacks up to a strength of $\varepsilon = 0.5$ in Figure 4. The noise-free QSVR is highly vulnerable to adversarial attacks, as evidenced by the decrease in AUC for small attack strengths of $\varepsilon = 0.01$ for all datasets except Toy. The largest decrease is in the AUC of the DoH dataset, which drops by an order of magnitude from 0.67 to 0.06 for the $\varepsilon = 0.01$ attack. As the attack strength increases, the AUC continues to decrease for all datasets until it approaches 0.0 for $p = 0.3$. We conclude that techniques to increase the adversarial robustness of the model should be investigated.

The unchanged AUC of Toy for $\varepsilon = 0.01$ can be explained by the creation process of the dataset. The dataset was created to be linearly separable with a separation distance of 0.4 between the classes, so the data points must be shifted a large distance in feature space to be misclassified. However, since the AUC drops to 0.0 for $\varepsilon = 0.3$, which is smaller than the separation distance, we might conclude that the QSVR is susceptible to overfitting.

2) Noisy Adversarial Attacks: Second, the performance of the noisy QSVR when subjected to adversarial perturbations of strength $\varepsilon = 0.1$ is shown in Figure 3. Omitting *miscalibration*, we find that for noise levels below $p = 0.1$, the AUC is low for most noise types and datasets, following the trend of high adversarial vulnerability observed in Figure 4. At higher noise levels, however, the AUC typically recovers to some extent, reaching a plateau at about an AUC of 0.5. This indicates that the model transitions to a random classifier, showing that the adversarial attacks are so powerful that quantum noise cannot improve performance beyond that of a random classifier. Other researchers report similar results, noting that random noise and adversarial noise are fundamentally different, and that models resilient to random noise are often still vulnerable to adversarial noise [38].

Miscalibration noise affects both models under attack and models not under attack, similarly, resulting in spikes around $p = k\pi$ for $k = 0, 1, 2$ in most datasets. Interestingly, for adversarially attacked models, as opposed to models that are not under attack, these spikes can be lower than the adjacent plateaus, depending on the dataset. Between these spikes, the AUC generally remains stable, forming plateau regions.

The DoH dataset is an outlier, with an AUC of 0.0 across almost all noise types and strengths, which is explained by its extreme vulnerability to adversarial attacks seen in Figure 4. This vulnerability can be explained by Figure 5 in Section B in the appendix, showing the p-value from the *Kolmogorov–Smirnov test* and the maximum feature variance within the test set. The DoH dataset has a relatively high p-value combined with a very low variance. The high p-value indicates a high probability that the normal and anomalous samples originate from the same distribution, while the low variance suggests a high degree of similarity between all samples, especially between the normal and anomalous data. As a result, the DoH dataset is difficult to classify, and even tiny adversarial attacks of strength $\varepsilon = 0.01$ lead to manipulations a magnitude greater than the variance within the dataset.

Notably, the AUC for the DoH dataset with *miscalibration* noise is 0.0 over nearly all noise levels and peaks only around $p = \pi$, where it approaches 1.0. An analysis of the adversarial test kernels for the DoH dataset subject to *miscalibration* noise is shown in Table III and highlights major differences between a high-performing run ($p = 2.9 \approx 0.9\pi$, AUC $= 0.98$) and a low-performing run ($p = 1.7 \approx 0.5\pi$, AUC $= 0.00$) for both classes. The mean kernel values for the high-AUC run are four orders of magnitude larger than those for the low-AUC run. In addition, the disparity between kernel values of classes 0 and 1 is greater in the high-performing scenario, allowing for easier distinction by the SVR and thus improved model AUC. Considering that the kernel entries represent the probability of measuring the all-zero state, we observe that *miscalibration* noise with a strength close to π shifts the DoH-embedding states closer to the all-zero state, thus increasing the kernel values.

Since the overrotation introduced by *miscalibration* noise is independent of the data, this type of noise can be thought of as additional fixed parameter rotation gates in the circuit. Because the parameters of these gates have a significant impact on model performance, we highlight the importance of using kernels tailored to the dataset, such as trainable kernels.

TABLE III
ANALYSIS OF THE ADVERSARIAL TEST KERNEL FOR THE DOH DATASET SUBJECT TO MISCALIBRATION NOISE

p	AUC	class	mean kernel value
2.9	0.98	0	1.743e-01 ± 5.157e-02
		1	1.734e-01 ± 5.121e-02
1.7	0.00	0	1.800e-05 ± 7.360e-06
		1	1.796e-05 ± 7.395e-06

The models attacked with $\varepsilon = 0.01$ and $\varepsilon = 0.3$ do not provide new insights, as they exhibit similar behavior to the $\varepsilon = 0.1$ attacks above, and can be obtained from the authors upon reasonable request.

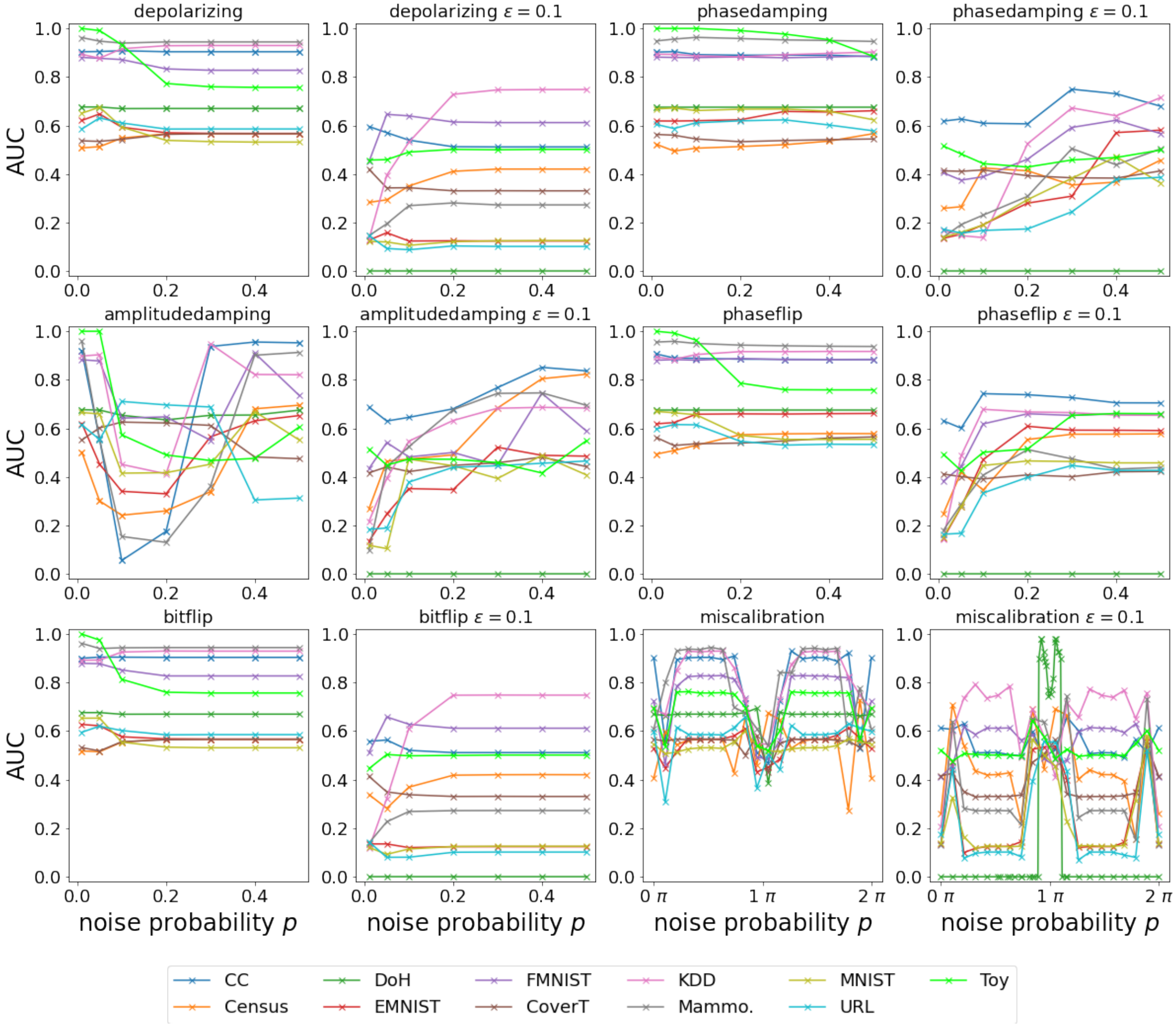

Fig. 3. Left two columns: Influence of six different noise types on the QSVR. Right two columns: Influence of six different noise types on the QSVR for adversarial attacks of strength $\varepsilon = 0.1$

We conclude that quantum noise is not suited for increasing the adversarial robustness of the QSVR. This finding is consistent with prior research [14], where the authors suggest that adding noise to QML models to increase the adversarial robustness is unlikely to be beneficial in practice.

3) Adversarial Training: Adversarial training is a straightforward approach to increasing the adversarial robustness of supervised learning algorithms. Table IV reveals that adversarial training increases the AUC for the adversarial test set on seven out of eleven datasets, and the average AUC over all datasets rises from 0.28 to 0.31. However, the increase in AUC is small, and except for FMNIST and Toy, the AUC remains below 0.5. For the test set without adversarial samples, the AUC decreases through adversarial training on six out of eleven datasets, and the average declines from 0.75 to 0.71. Table V shows the ratio of correctly classified normal samples to the total number of normal samples, as well as the same ratio for the anomalies. For normal data, the ratio is $\frac{\text{tn}}{\text{tn+fp}}$, and for the anomalies it is $\frac{\text{tp}}{\text{tp+fn}}$. We observe that retraining increases the classification ratio for the normal data of the Toy dataset from 0.78 to 0.94, while the ratio for the anomalies remains unchanged at 1.00. For KDD, we report similar results, but the increase in the classification ratio of the normal data through retraining is smaller. This shows that adversarial retraining can lead to more normal samples being classified correctly without influencing the classification of the anomalies, since the latter are not contained in the training set. However, this

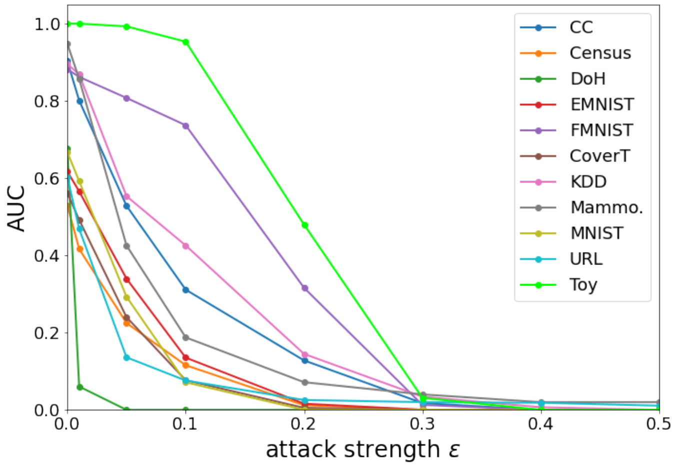

Fig. 4. Influence of adversarial attacks on the noise-free models.

was not observed for other datasets and was most pronounced for the synthetic dataset, suggesting this behavior requires a large separation distance between the two classes.

We conclude that adversarial training cannot be used to reliably harden the QSVR against adversarial attacks. We attribute this to the *semisupervised* setting, meaning that only normal samples are available during training.

TABLE IV
AUCs OF THE MODELS FOR THE TEST SET AND ADVERSARIAL TEST SET WITH AND WITHOUT ADVERSARIAL TRAINING.

Dataset	Test AUC w/ retraining	Test AUC w/o retraining	Adv AUC w/ retraining	Adv AUC w/o retraining
CC	0.85	0.90	0.37	0.31
Census	0.62	0.53	0.05	0.12
DoH	0.68	0.68	0.00	0.00
EMNIST	0.63	0.62	0.17	0.14
FMNIST	0.91	0.88	0.73	0.74
CoverT	0.44	0.56	0.18	0.08
KDD	0.71	0.90	0.46	0.43
Mammo.	0.74	0.95	0.22	0.19
MNIST	0.62	0.67	0.11	0.07
URL	0.59	0.60	0.08	0.08
Toy	1.00	1.00	1.00	0.95
Mean	**0.71**	**0.75**	**0.31**	**0.28**

V. Conclusion and Outlook

We first benchmarked our QSVR for semisupervised AD on 27-qubits IBM hardware and found that the average AUC was slightly lower than that of the noiseless simulation (0.72 compared to 0.76). However, the QSVR outperformed the noiseless simulation on two out of eleven datasets.

Second, the influence of six noise channels on the performance of the QSVR was evaluated, revealing that the QSVR is largely robust against *deporarizing*, *phasedamping*, *phase flip* and *bit flip* noise. *Amplitude damping*, on the other hand, results in the most significant degradation of the model and *miscalibration* noise also has the potential to impact performance.

TABLE V
Ratios of correctly classified normal and anomalous samples.

Dataset	retraining		no retraining	
	norm.	anom.	norm.	anom.
CC	0.96	0.28	0.98	0.24
Census	1.00	0.00	1.00	0.00
DoH	1.00	0.00	1.00	0.00
EMNIST	1.00	0.02	0.98	0.02
FMNIST	0.86	0.66	0.86	0.68
CoverT	1.00	0.00	1.00	0.00
KDD	1.00	0.34	0.96	0.34
Mammo.	0.96	0.18	1.00	0.14
MNIST	1.00	0.00	1.00	0.00
URL	0.96	0.08	0.98	0.04
Toy	0.94	1.00	0.78	1.00
Mean	**0.97**	**0.23**	**0.96**	**0.22**

Finally, the adversarial robustness of the (noisy) model was assessed, and it was observed that the QSVR is highly vulnerable to adversarial attacks. Even weak PGD attacks with a strength of $\varepsilon = 0.01$ can reduce the AUC by up to an order of magnitude. Introducing quantum noise does not yield any beneficial effect, neither on the unattacked model's performance nor on its adversarial robustness. Moreover, adversarial training does not reliably improve the adversarial robustness of the model. Consequently, we conclude that the QSVR demonstrates potential for semisupervised AD in the NISQ era, however, special attention should be paid to the vulnerability to adversarial attacks and *amplitude damping* and *miscalibration* noise.

We emphasize the importance of employing dataset-specific kernels and recommend exploring trainable kernels to further enhance the performance of QML models. Future research directions could also include expanding the model to more qubits, and finally, future work may also focus on techniques to enhance the adversarial robustness of the QSVR and defend against such attacks.

Acknowledgment

The research is part of the Munich Quantum Valley, which is supported by the Bavarian state government with funds from the Hightech Agenda Bayern Plus.

References

[1] M. Schuld, *Supervised quantum machine learning models are kernel methods*, 2021. arXiv: 2101.11020 [quant-ph].

[2] Y. Liu, S. Arunachalam, and K. Temme, "A rigorous and robust quantum speed-up in supervised machine learning," *Nature Physics*, vol. 17, pp. 1013–1017, 9 Sep. 2021, ISSN: 17452481. DOI: 10.1038/s41567-021-01287-z.

[3] L. Ruff *et al.*, "A unifying review of deep and shallow anomaly detection," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 756–795, 2021. DOI: 10.1109/JPROC.2021.3052449.

[4] K. Tscharke, S. Issel, and P. Debus, "Semisupervised anomaly detection using support vector regression with quantum kernel," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Los Alamitos, CA, USA: IEEE Computer Society, Sep. 2023, pp. 611–620. DOI: 10.1109/QCE57702.2023.00075.

[5] Y. Du *et al.*, "Quantum noise protects quantum classifiers against adversaries," *Phys. Rev. Res.*, vol. 3, p. 023 153, 2 May 2021. DOI: 10.1103/PhysRevResearch.3.023153.

[6] C. Huang and S. Zhang, "Enhancing adversarial robustness of quantum neural networks by adding noise layers," *New Journal of Physics*, vol. 25, no. 8, p. 083 019, Aug. 2023. DOI: 10.1088/1367-2630/ace8b4.

[7] V. Havlíček *et al.*, "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, 2019.

[8] A. Delilbasic *et al.*, "Quantum support vector machine algorithms for remote sensing data classification," in *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, 2021, pp. 2608–2611. DOI: 10.1109/IGARSS47720.2021.9554802.

[9] S. Farhan Ahmad, R. Rawat, and M. Moharir, "Quantum machine learning with hqc architectures using non-classically simulable feature maps," in *2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, 2021, pp. 345–349. DOI: 10.1109/ICCIKE51210.2021.9410753.

[10] M. Mafu and M. Senekane, "Design and implementation of efficient quantum support vector machine," in *2021 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, 2021, pp. 1–4. DOI: 10.1109/ICECET52533.2021.9698509.

[11] O. Kyriienko and E. B. Magnusson, *Unsupervised quantum machine learning for fraud detection*, 2022. arXiv: 2208.01203 [quant-ph].

[12] D. García-Martín, M. Larocca, and M. Cerezo, "Effects of noise on the overparametrization of quantum neural networks," *Phys. Rev. Res.*, vol. 6, p. 013 295, 1 2024. DOI: 10.1103/PhysRevResearch.6.013295. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevResearch.6.013295.

[13] Y. Zhou and P. Zhang, "Noise-resilient quantum machine learning for stability assessment of power systems," *IEEE Transactions on Power Systems*, vol. 38, no. 1, pp. 475–487, 2023. DOI: 10.1109/TPWRS.2022.3160384.

[14] D. Winderl, N. Franco, and J. M. Lorenz, *Quantum neural networks under depolarization noise: Exploring white-box attacks and defenses*, 2023. arXiv: 2311.17458 [quant-ph].

[15] N. H. Nguyen, E. C. Behrman, and J. E. Steck, "Quantum learning with noise and decoherence: A robust quantum neural network," *Quantum Machine Intelligence*, vol. 2, no. 1, p. 1, Jan. 2020.

[16] J. Yao *et al.*, "Noise-robust end-to-end quantum control using deep autoregressive policy networks," in *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*, J. Bruna, J. Hesthaven, and L. Zdeborova, Eds., ser. Proceedings of Machine Learning Research, vol. 145, PMLR, Aug. 2022, pp. 1044–1081. [Online]. Available: https://proceedings.mlr.press/v145/yao22a.html.

[17] K. Ju *et al.*, "Harnessing inherent noises for privacy preservation in quantum machine learning," in *ICC 2024 - IEEE International Conference on Communications*, 2024, pp. 1121–1126. DOI: 10.1109/ICC51166.2024.10622663.

[18] D. Winderl, N. Franco, and J. M. Lorenz, *Constructing optimal noise channels for enhanced robustness in quantum machine learning*, 2024. arXiv: 2404.16417 [quant-ph]. [Online]. Available: https://arxiv.org/abs/2404.16417.

[19] M. Schuld and F. Petruccione, *Machine Learning with Quantum Computers* (Quantum Science and Technology), en. Cham: Springer International Publishing, 2021, ISBN: 978-3-030-83097-7. DOI: 10.1007/978-3-030-83098-4.

[20] M. Schuld and N. Killoran, "Quantum machine learning in feature hilbert spaces," *Phys. Rev. Lett.*, vol. 122, p. 040 504, 4 Feb. 2019. DOI: 10.1103/PhysRevLett.122.040504.

[21] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[22] D. Suter and G. A. Álvarez, "Colloquium: Protecting quantum information against environmental noise," *Rev. Mod. Phys.*, vol. 88, p. 041 001, 4 Oct. 2016. DOI: 10.1103/RevModPhys.88.041001.

[23] N. Kaufmann, I. Rojkov, and F. Reiter, *Characterization of coherent errors in noisy quantum devices*, 2023. arXiv: 2307.08741 [quant-ph].

[24] J. Preskill, *Lecture Notes for Ph219/CS219: Quantum Information Chapter 3*, en, Oct. 2018.

[25] J. Wallman *et al.*, "Estimating the coherence of noise," *New Journal of Physics*, vol. 17, no. 11, p. 113 020, Nov. 2015. DOI: 10.1088/1367-2630/17/11/113020.

[26] M. M. Wilde, *Quantum Information Theory*. Cambridge University Press, 2013. DOI: https://doi.org/10.1017/CBO9781139525343.

[27] M. Wendlinger, K. Tscharke, and P. Debus, *A comparative analysis of adversarial robustness for quantum and classical machine learning models*, 2024. arXiv: 2404.16154 [cs.LG].

[28] C. Szegedy *et al.*, *Intriguing properties of neural networks*, 2014. arXiv: 1312.6199 [cs.CV].

[29] I. J. Goodfellow, J. Shlens, and C. Szegedy, *Explaining and harnessing adversarial examples*, 2015. arXiv: 1412.6572 [stat.ML].

[30] A. Madry *et al.*, *Towards deep learning models resistant to adversarial attacks*, 2019. arXiv: 1706.06083 [stat.ML].

[31] A. Javadi-Abhari *et al.*, *Quantum computing with Qiskit*, 2024. DOI: 10.48550/arXiv.2405.08810. arXiv: 2405.08810 [quant-ph].

[32] K. Kottmann *et al.*, "Variational quantum anomaly detection: Unsupervised mapping of phase diagrams on a physical quantum computer," *Phys. Rev. Res.*, vol. 3, p. 043 184, 4 Dec. 2021. DOI: 10.1103/PhysRevResearch.3.043184.

[33] V. Bergholm *et al.*, *Pennylane: Automatic differentiation of hybrid quantum-classical computations*, 2022. arXiv: 1811.04968 [quant-ph].

[34] P. Vincent *et al.*, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08, Helsinki, Finland: Association for Computing Machinery, 2008, pp. 1096–1103, ISBN: 9781605582054. DOI: 10.1145/1390156.1390294. [Online]. Available: https://doi.org/10.1145/1390156.1390294.

[35] E. T. Escudero *et al.*, "Assessing the impact of noise on quantum neural networks: An experimental analysis," in *Hybrid Artificial Intelligent Systems*. Springer Nature Switzerland, 2023, pp. 314–325, ISBN: 9783031407253. DOI: 10.1007/978-3-031-40725-3_27. [Online]. Available: http://dx.doi.org/10.1007/978-3-031-40725-3_27.

[36] L. Domingo, G. Carlo, and F. Borondo, "Taking advantage of noise in quantum reservoir computing," *Scientific Reports*, vol. 13, no. 1, p. 8790, 2023, ISSN: 2045-2322. DOI: 10.1038/s41598-023-35461-5. [Online]. Available: https://doi.org/10.1038/s41598-023-35461-5.

[37] R. LaRose and B. Coyle, "Robust data encodings for quantum classifiers," *Phys. Rev. A*, vol. 102, p. 032 420, 3 Sep. 2020. DOI: 10.1103/PhysRevA.102.032420.

[38] A. Fawzi, S.-M. Moosavi-Dezfooli, and P. Frossard, *Robustness of classifiers: From adversarial to random noise*, 2016. arXiv: 1608.08967 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1608.08967.

[39] A. D. Pozzolo *et al.*, "Calibrating probability with undersampling for unbalanced classification," in *2015 IEEE Symposium Series on Computational Intelligence*, 2015, pp. 159–166. DOI: 10.1109/SSCI.2015.33.

[40] D. Dua and C. Graff, *UCI machine learning repository*, 2017. [Online]. Available: http://archive.ics.uci.edu/ml.

[41] J. A. Blackard and D. J. Dean, "Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables," *Computers and Electronics in Agriculture*, vol. 24, no. 3, pp. 131–151, 1999, ISSN: 0168-1699. DOI: https://doi.org/10.1016/S0168-1699(99)00046-0.

[42] M. MontazeriShatoori *et al.*, "Detection of doh tunnels using time-series classification of encrypted traffic," in *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, 2020, pp. 63–70. DOI: 10.1109/DASC-PICom-CBDCom-CyberSciTech49142.2020.00026.

[43] G. Cohen *et al.*, "Emnist: Extending mnist to handwritten letters," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 2921–2926. DOI: 10.1109/IJCNN.2017.7966217.

[44] H. Xiao, K. Rasul, and R. Vollgraf, *Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms*, 2017. arXiv: 1708.07747 [cs.LG].

[45] M. Tavallaee *et al.*, "A detailed analysis of the kdd cup 99 data set," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, 2009, pp. 1–6. DOI: 10.1109/CISDA.2009.5356528.

[46] Y. Lecun *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. DOI: 10.1109/5.726791.

[47] K. S. WOODS *et al.*, "Comparative evaluation of pattern recognition techniques for detection of microcalcifications in mammography," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 07, no. 06, pp. 1417–1436, 1993. DOI: 10.1142/S0218001493000698. eprint: https://doi.org/10.1142/S0218001493000698.

[48] M. S. I. Mamun *et al.*, "Detecting malicious urls using lexical analysis," in *Network and System Security*, J. Chen *et al.*, Eds., Cham: Springer International Publishing, 2016, pp. 467–482, ISBN: 978-3-319-46298-1.

Appendix A
Overview of the Datasets

TABLE VI
Overview of the datasets used for the experiments.

Dataset	Reference	Normal class	Anomalous class
CC	[39]	Normal	Anomalous
Census	[40]	≤50k	>50k
CoverT	[41]	1-4	5-7
DoH	[42]	Benign	Malicious
EMNIST	[43]	A-M	N-Z
FMNIST	[44]	0-4	5-9
KDD	[45]	Normal	Anomalous
MNIST	[46]	0-4	5-9
Mammo	[47]	Normal	Malignant
Toy	/	Normal	Anomalous
URL	[48]	Benign	Non-benign

Appendix B
Analysis of the DoH Dataset

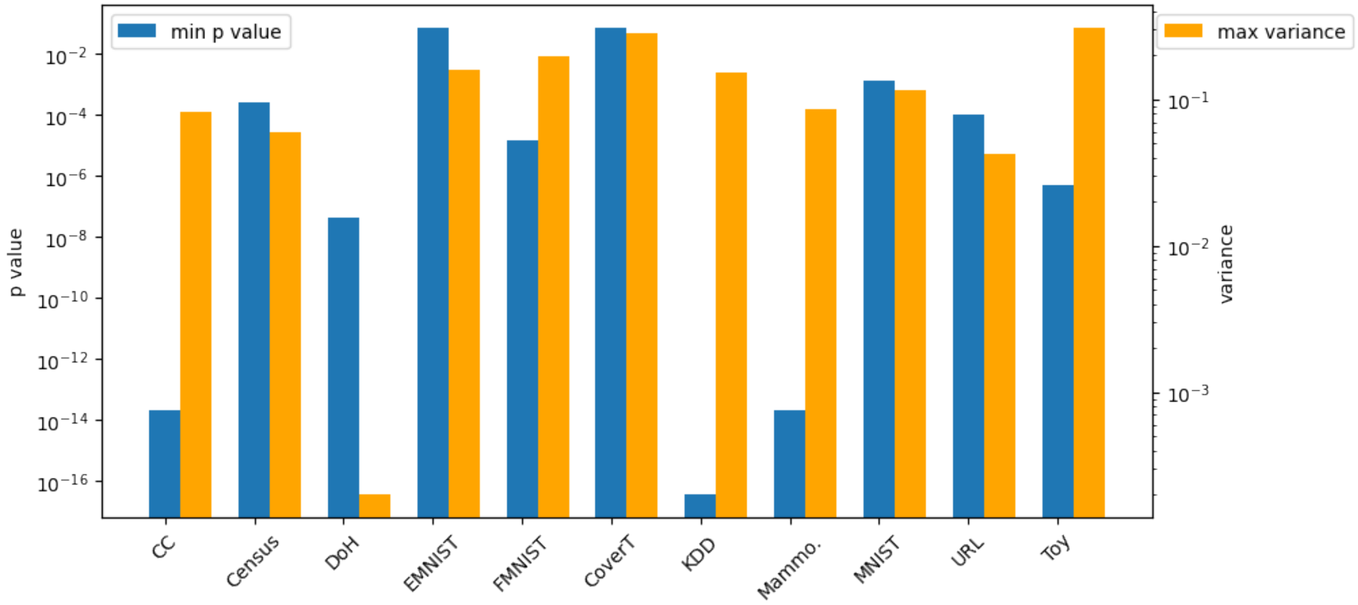

Fig. 5. Left axis: min. p-value obtained from the Kolmogorov-Smirnov test. It gives the probability of the normal and anomalous samples being from the same distribution. Right axis: max. variance within the test set. Both values are build feature-wise and then the min/max value is plotted.

QSVR for Anomaly Detection: Performance, Noise Resilience, and Adversarial Vulnerabilities

Abstract—**Anomaly Detection (AD) is critical in data analysis, especially in IT security. Machine Learning algorithms have emerged as a powerful tool for AD in large scale data. Exploring the potential of Quantum Machine Learning methods, we study the application of Quantum Kernel Methods to AD. Building on existing work, we extend a Quantum Support Vector Regression (QSVR) for semisupervised AD by benchmarking it on IBM hardware, where it achieves good classification performance and even outperforms a noiseless simulator on two datasets. Moreover, we investigate the influence of – in the NISQ-era inevitable – quantum noise on the performance of our QSVR, and report that depolarizing, phase damping, phaseflip, and bitflip noise only minimally affect performance, while amplitude damping and strong miscalibration are more disruptive. Finally, we peek into the field of Quantum Adversarial Machine Learning and show that the QSVR is highly vulnerable to adversarial attacks and that noise does not clearly improve the adversarial robustness of the model.**

Index Terms—**benchmark, semisupervised learning, noise, hardware, adversarial attacks, robustness, quantum kernel methods, quantum machine learning**

I. Introduction

Quantum Machine Learning (QML) merges Quantum Computing (QC) and Machine Learning (ML) to apply QC's potential advantages to ML. Recently, Quantum Kernel Methods (QKMs) have gained attention for replacing many supervised QML models and for their ability to outperform variational circuits **supervised˙qml˙are˙kernel˙schuld2021**. Theoretical findings also suggest QKMs can address classification problems unsolvable by classical ML, like the discrete logarithm problem **quantum˙kernel˙advantage˙discrete˙log˙2021**.

Anomaly Detection (AD) is crucial in IT security, identifying deviations from normal patterns in areas like network intrusion and fraud detection **Ruff2020**. However, ML models in security-critical applications are vulnerable to adversarial attacks, where small, carefully crafted perturbations in inputs lead to misclassification. (Quantum) Adversarial Machine Learning (QAML/AML) explores both generating and defending against these attacks.

Given QML's potential to solve problems challenging for classical methods, applying QML to AD is a tempting progression. In 2023, Tscharke et al. **Tscharke** introduced a semisupervised AD approach based on the reconstruction loss of a Quantum Support Vector Regression (QSVR) with a quantum kernel. They benchmarked their QSVR against a quantum autoencoder (QAE), a classical SVR (CSVR) with RBF kernel, and a classical autoencoder (CAE) on ten real-world datasets. Their simulated QSVR performed comparably to CSVR, slightly outperforming other models. However, the implementation of their QSVR on hardware was left open and is tackled in this paper.

In today's Noisy Intermediate-Scale Quantum (NISQ) era, noise limits QC's applicability for industrial tasks, making it essential to understand how noise affects quantum algorithms. Noise impacts QML models by affecting predictive performance and, in QAML, altering robustness against adversarial attacks **quantum˙noise˙against˙adv˙attacks˙Du**, **noise˙improve˙adv˙robustness˙Huang**.

The remainder of this work is structured as follows. The next subsection I-A reviews related research in QKMs, the influence of noise on QML, and QAML. Our contributions are detailed in subsection I-B. The following Background (section II) covers the fundamentals of QKMs, noisy QC, and adversarial attacks. Next, the Methods (section III) describes the QSVR hardware implementation, noise simulation, and adversarial attack generation. In the Results and Discussion (section IV) we analyze the results of the hardware experiments and investigate the influence of noise and adversarial attacks on the model's performance. Finally, Conclusion and Outlook (section V) highlights the key results of this work and gives future research directions.

A. Related Work

In 2019, Havlicek et al. **Havlicek2019** introduced a quantum Support Vector Machine (QSVM) for binary classification on a two-qubit NISQ device. Since then, QSVMs have been applied to many areas, e.g. remote sensing image classification **Delilbasic2021**, mental health treatment prediction **Ahmad2021** or breast cancer prediction **Mafu2021**. Kyriienko and Magnusson **quantum˙one˙class˙SVM** extended this to unsupervised fraud detection with a simulated one-class QSVM, and Tscharke et al. **Tscharke** developed a QSVR for semisupervised AD in 2023. However, to the best of our knowledge, a QSVR for semisupervised AD has not yet been set up on hardware.

Research has also explored the influence of noise in QML models **noise˙influence˙Diego**, **noise˙influence˙zhou**, **noise˙influence˙winderl**, focusing on noise robustness **noise˙robustness˙Nguyen**, **noise˙robustness˙Yao** or beneficial use of noise **noise˙beneficial˙ju**, **noise˙beneficial˙winderl**. However, as far as we know, the influence of noise on a QSVR for semisupervised AD has not yet been evaluated.

Finally, the link between noise and QAML was established by Du et al. **quantum˙noise˙against˙adv˙attacks˙Du** in 2021, who found that adding depolarization noise can increase adversarial robustness. Building on this, Huang and

Zhang **noise˙improve˙adv˙robustness˙Huang** improved the adversarial robustness of Quantum Neural Networks by adding noise layers in 2023.

B. Contributions

Our work aims to gain further insights into the potential of QSVR for semisupervised AD in the NISQ-era which we achieve through these contributions:

1) We investigate how the model performs on hardware and report that the QSVR achieves good classification performance on a 27-qubit IBM device, outperforming the noise-free simulation on two out of ten real-world datasets.
2) We show that the QSVR is largely robust against noise by training and evaluating a total of 462 noisy models. Furthermore, we observe that *amplitude damping* and strong *miscalibration* have the most harmful effect on the model's performance and that the artificial Toy dataset, constructed to be linearly separable, suffers the most from noise.
3) Lastly, we investigate the QSVR's robustness against adversarial attacks and find it highly vulnerable, with performance on real-world datasets dropping by up to one order of magnitude for a small attack strength of $\epsilon = 0.01$. Introducing noise to the QSVR does not clearly improve the model's adversarial robustness.

II. Background

A. Quantum Kernel Methods

A kernel is a positive, semidefinite function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{C}$ on the input set $\mathcal{X}$. It uses a distance measure $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ between two input vectors $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$ to create a model that captures the properties of a data distribution. A *feature map* $\phi : \mathcal{X} \to \mathcal{F}$ maps input vectors $\boldsymbol{x}$ to a Hilbert or *feature space* $\mathcal{F}$. They are of great importance in ML, as they map input data in a higher-dimensional space with a well-defined metric. The feature map can be a nonlinear function, which changes the relative position of the data points. Hence, the dataset can become easier to classify in feature space, and even linearly separable. We connect feature maps to kernels by defining a kernel via

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) := \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle_{\mathcal{F}} \tag{1}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{F}}$ is the inner product defined on $\mathcal{F}$.

With the exponentially large Hilbert space of QCs, the use of QKMs for ML is close at hand. A quantum feature map $\psi : \boldsymbol{x} \to |\phi(\boldsymbol{x})\rangle$ is implemented via a feature-embedding circuit $U(\boldsymbol{x})$ that acts on a ground state $|0\ldots0\rangle$ of a Hilbert space $\mathcal{F}$ as $|\phi(\boldsymbol{x})\rangle = U(\boldsymbol{x})\,|0\ldots0\rangle$. The distance measure in a quantum kernel is obtained by the absolute square of the inner product of quantum states. On hardware, this can be realized by the *inversion test*, where sample $\boldsymbol{x}_i$ is encoded in the unitary U followed by the adjoint $U^\dagger$ encoding the second sample $\boldsymbol{x}_j$, and measuring the probability of the all-zero state. Hence, the quantum kernel is defined as

$$\begin{aligned} \kappa(x_i, x_j) &= |\langle \phi(\boldsymbol{x}_i) | \phi(\boldsymbol{x}_j) \rangle|^2 \\ &= \left| \langle 0^{\otimes n} | \, U^\dagger(\boldsymbol{x}_i) U(\boldsymbol{x}_j) \, | 0^{\otimes n} \rangle \right|^2 \end{aligned} \tag{2}$$

and returns the *overlap* or *fidelity* of the two states. A more in-depth description of quantum kernels can be found in **Schuld˙ML˙w˙QC**, **Schuld˙QML˙in˙Feature˙Hilbert˙Spaces**.

There exist many different encoding techniques for the circuit realizing the quantum feature map but for this work, we will focus on *angle encoding* because of its beneficial complexity with respect to the number of gates of $\mathcal{O}(k)$ for an input vector **x** of dimension k. *Angle encoding* is a specific form of time-evolution encoding, where a scalar value $\boldsymbol{x}$ is encoded in the unitary evolution of a quantum system, governed by a Hamiltonian H. The unitary of time-evolution encoding is given by

$$U(x) = e^{-i\boldsymbol{x}H}. \tag{3}$$

In the case of *angle encoding*, the Pauli matrices σ_a with $a \in \{x, y, z\}$ are used in the Hamiltonian $H = \frac{1}{2}\sigma_a$.

B. Noisy Quantum Computing

Real quantum systems are never completely isolated from the environment; for example, an electron realizing a qubit will interact with other charged particles. Moreover, quantum computers are programmed by an external system and hence can never be a closed system **Nielsen**, **Suter˙Noise**.

In the current NISQ-era, noise significantly limits the performance of quantum algorithms, primarily through *coherent* and *incoherent* noise. *Coherent* noise arises from systematic, reversible errors that lead to predictable yet unwanted evolution of the states, often caused by imperfect calibrations or imprecise control signals **kaufmann2023˙noise**. *Coherent* noise is an *unitary* evolution of the system, characterized by having only one operation element in the *operator-sum representation* introduced below **preskill˙lectures˙noise**, **Wallman˙Noise**.

Incoherent noise, on the other hand, is characterized by random, stochastic processes caused by insufficient isolation of the system from its environment. These uncontrolled interactions between system and environment lead to deviations between the targeted and the actual evolution and a loss of coherence in the system **kaufmann2023˙noise**, **Suter˙Noise**.

Quantum noise can be modeled by a quantum channel, with the term "channel" drawn from classical information theory **Nielsen**. In the *operator-sum representation*, a channel is described by the map $\mathcal{E}$ with operation elements (or *Kraus operators*) $\{E_i\}$ mapping the density operator $\rho = |\psi\rangle\langle\psi|$ to another density operator $\mathcal{E}(\rho)$.

$$\mathcal{E}(\rho) = \sum_i E_i \rho E_i^\dagger. \tag{4}$$

In the following, selected types of single-qubit noisy channels are described and their operation elements E_i are listed. For a more detailed explanation of the noisy quantum channels, we refer to **Nielsen**, **Wilde˙2013**, **preskill˙lectures˙noise**.

1) *Miscalibration* channel: a coherent noise channel applying an "overrotation" by p to the R_a rotation gate with $a \in \{X, Y, Z\}$. This can be caused by an imperfect calibration of the device **kaufmann2023˙noise**. Since the channel is unitary, it has only one operation element:

$$E_0 = R_a(p). \tag{5}$$

2) *Bitflip* channel: flips the state of a qubit from $|0\rangle$ to $|1\rangle$ and vice versa with probability p. The operation elements are:

$$\begin{aligned} E_0 &= \sqrt{1-p}I = \sqrt{1-p}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ E_1 &= \sqrt{p}X = \sqrt{p}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{aligned} \tag{6}$$

3) *Phaseflip* channel: applies a phase of -1 to the $|1\rangle$ state with probability p and leaves the $|0\rangle$ state unchanged. It has the operation elements:

$$\begin{aligned} E_0 &= \sqrt{1-p}I = \sqrt{1-p}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ E_1 &= \sqrt{p}Z = \sqrt{p}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned} \tag{7}$$

The channel has the following effect on the state's density matrix ρ:

$$\mathcal{E}\begin{pmatrix} \rho_{00} & \rho_{01} \\ \rho_{10} & \rho_{11} \end{pmatrix} = \begin{pmatrix} \rho_{00} & (1-2p)\rho_{01} \\ (1-2p)\rho_{10} & \rho_{11} \end{pmatrix} \tag{8}$$

From this we can see that the phaseflip channel destroys superposition by decaying the off-diagonal terms of the density matrix ρ while the on-diagonal terms remain invariant.

4) *Depolarizing* channel: the qubit remains intact with probability $1-p$, while an error occurs with probability p. If an error occurs, the state is replaced with a uniform ensemble of the three states $X|\psi\rangle, Y|\psi\rangle, Z|\psi\rangle$. This is a symmetric decoherence channel, defined by operation elements:

$$\begin{aligned} E_0 = \sqrt{1-p}I \quad &= \sqrt{1-p}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ X\text{-Error: } E_1 = \sqrt{p/3}X \quad &= \sqrt{p/3}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ Y\text{-Error: } E_2 = \sqrt{p/3}Y \quad &= \sqrt{p/3}\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \\ Z\text{-Error: } E_3 = \sqrt{p/3}Z \quad &= \sqrt{p/3}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned} \tag{9}$$

5) *Phase Damping* channel: the *phase damping* or *dephasing* channel models the effect of random environmental scattering on a qubit, such as photon interactions in a waveguide or atomic states disturbed by distant charges. This channel causes partial loss of phase information without energy loss. It produces the same effect as the *phase flip* channel, with the phase damping λ related to the phase flip probability p by

$$p = \frac{1-\sqrt{1-\lambda}}{2}. \tag{10}$$

6) *Amplitude Damping* channel: describes the effect of energy loss, such as when an atom emits a photon. The channel acts on the quantum system A and the environment E in the following way: if both A and E are in their ground state $|0\rangle$, nothing happens. If A is in the excited state $|1\rangle_A$, a photon will be emitted with probability p, leading to the excitation of the environment causing the transition $|0\rangle_E \rightarrow |1\rangle_E$ while A drops to the ground state, i.e. $|1\rangle_A \rightarrow |0\rangle_A$. The evolution caused by the channel can be summarized as:

$$\begin{aligned} &|0\rangle_A \otimes |0\rangle_E \mapsto |0\rangle_A \otimes |0\rangle_E \\ &|1\rangle_A \otimes |0\rangle_E \mapsto \sqrt{1-p}|1\rangle_A \otimes |0\rangle_E + \sqrt{p}|0\rangle_A \otimes |1\rangle_E \end{aligned} \tag{11}$$

This is achieved by the operation elements:

$$\begin{aligned} E_0 &= \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{bmatrix} \\ E_1 &= \begin{bmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{bmatrix} \end{aligned} \tag{12}$$

C. Adversarial Attacks

QML models are typically trained using a hybrid quantum-classical approach. In this framework, model parameters are optimized using a classical optimization algorithm, while the quantum part is restricted to the evaluation of the loss which is done (partly) by the quantum computer or simulator. This hybrid approach allows us to easily extend the concept of adversarial attacks from classical ML to QML, which has already been successfully demonstrated in **adv˙robustness˙wendlinger**.

An adversarial attack is a small, carefully crafted perturbation of the k-dimensional input $\boldsymbol{x}$ that leads to a misclassification of the input by the model **adv˙attacks˙szegedy**, **adv˙attacks˙goodfellow**. An untargeted adversarial sample is created by maximizing the loss $\mathcal{L}$ of the model, while keeping the perturbation δ small enough to be imperceptible to humans, e.g. by ensuring $\delta \in \Delta = \{\delta \in \mathbb{R}^k : \|\delta\|_\infty \leq \varepsilon\}$. In general, the ideal perturbation is given by

$$\delta \equiv \underset{\delta' \in \Delta}{\operatorname{argmax}}\, \mathcal{L}\left(f\left(x+\delta'; \theta^*\right), y\right), \tag{13}$$

where f is the model, θ^* are the model's optimized parameters after training, and y is the target.

One of the most widely used methods for generating adversarial samples is Projected Gradient Descent (PGD) **pgd˙madry**. PGD iteratively maximizes the model's prediction error while ensuring the perturbation remains within a predefined range. This approach has become a standard tool for evaluating the robustness of models against adversarial threats. The perturbed input is found by

$$\boldsymbol{x}^{t+1} = \Pi_{\boldsymbol{x}+S}\left(\boldsymbol{x}^t + \alpha \operatorname{sgn}\left(\nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}^t, y; \theta^*)\right)\right), \qquad (14)$$

where $\boldsymbol{x}^t$ represents the perturbed data at step t, $\Pi_{\boldsymbol{x}+S}$ clips the perturbed data into the range of the normalized input set S, and α is the step size.

III. Methods

The experiments performed in this work are threefold. First, we benchmarked the QSVR for semisupervised AD on hardware. Second, we evaluated the influence of noise on the classification performance of the QSVR and third, we examined the influence of noise on the adversarial robustness of the QSVR. For a detailed description of the model, the datasets, and the preprocessing techniques, we refer to **Tscharke**. The datasets were reduced to five dimensions using Principal Components Analysis. The source code can be obtained from the authors upon reasonable request.

A. Quantum Support Vector Regression Model

The QSVR for semisupervised AD is described in detail in **Tscharke** and the kernel circuit is displayed in Fig. 1. The first data point $\boldsymbol{x}_i$ is encoded by the unitary U, followed by its inverse $U^\dagger$ encoding the second data point $\boldsymbol{x}_j$. The unitary U consists of a layer of RZ gates and a layer of RX gates followed by a layer of IsingZZ[1] gates for creating entanglement. The single qubit gates encode one feature each, while the IsingZZ gate's parameter is a product of two features. The kernel entry $K_{ij} = K_{ji}$ is obtained by measuring the probability of the all-zero state after applying both unitaries.

B. Hardware Experiments

In the hardware experiments, we used a training set of size 30 from the normal class and a test set of size 50 with an equal class ratio, following **Tscharke**.

1) Device Specifications: The experiments were performed on IBM System One at Ehningen in January 2024. The system is a 27-qubit superconducting quantum computer with a Quantum Volume of 64. The QSVR was implemented using qiskit **qiskit** with the default error mitigation techniques. Further specifications of the system are listed in table I.

2) Reference Models: We benchmarked our model against four other models, following the approach in **Tscharke**. The models include the simulated QSVR and QAE based on **Kottmann2021**, a classical SVR, and a classical Autoencoder.

C. Generation of Noise

The influence of noise on the QSVR for semisupervised AD was evaluated by applying six noise channels with different strengths to the quantum circuit calculating the kernel. The seven noise probabilities used in the experiments are $p \in \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$ and the six noise channels are [*miscalibration*, *bitflip*, *phaseflip*, *depolarizing*, *phase damping*, *amplitude damping*]. For the *miscalibration* channel, the noise probability p is the overrotation in radians. This leads to a total of $7 \cdot 6 = 42$ models per dataset. The noisy QSVR was simulated using PennyLane **pennylane**. We used a training set of size 100 from the normal class and a test set of size 100 with an equal class ratio.

TABLE I
Specification of the IBM System One Ehningen.

Spec	Value
Name	IBM Quantum System One at Ehningen
System type	Superconducting
Number of qubits	27
Quantum Volume	64
Processor type	Falcon
Deployment year	2021
Coherence time	$\approx 150\mu$s
Single qubit error	$\approx 0.025\%$
Two qubit gate error	$\approx 0.7\%$
Operation time of 2 qubit gate	$\approx$ 300ns for CNOT

D. Generation of Adversarial Attacks

We created 100 adversarial samples (50 of each class) using PGD with the parameters listed in table II. The attacks targeted the noise-free models and were then transferred to the noisy models.

TABLE II
Overview of the parameters used in the PGD attacks.

Spec	Values
Attack strength ε	[0.01, 0.1, 0.3]
Iterations n	50
α	ε/n

IV. Results and Discussion

In this study, we first benchmarked our QSVR on the 27-qubit IBM Ehningen device (labelled *qc-QSVR*) and compared its performance against the simulated quantum baseline models QSVR (simulated version of our model, see **Tscharke**) and QAE (based on **Kottmann2021**), as well as CSVR and CAE as classical baselines. Second, we introduced six different noise channels of various strengths to evaluate the influence of noise on the QSVR algorithm. Third, the influence of noise on the adversarial robustness of the model was evaluated by exposing the (noisy) models to adversarial attacks.

A. Hardware Results

Figure 2 compares model performance across eleven datasets using Area Under the ROC Curve (AUC), a commonly used metric in AD that measures the trade-off between the true and false positive rate independent of a detection threshold. An AUC of 1.0 means perfect classification of the dataset. The datasets include Credit Card Fraud (CC), Census, Forest Cover Type (CoverT), Domain Name System over HTTPS (DoH), EMNIST, Fashion MNIST (FMNIST), Network Intrusion (KDD), MNIST, Mammography (Mammo), URL, and the constructed dataset Toy. The models included in the study

[1] https://docs.pennylane.ai/en/stable/code/api/pennylane.IsingZZ.html

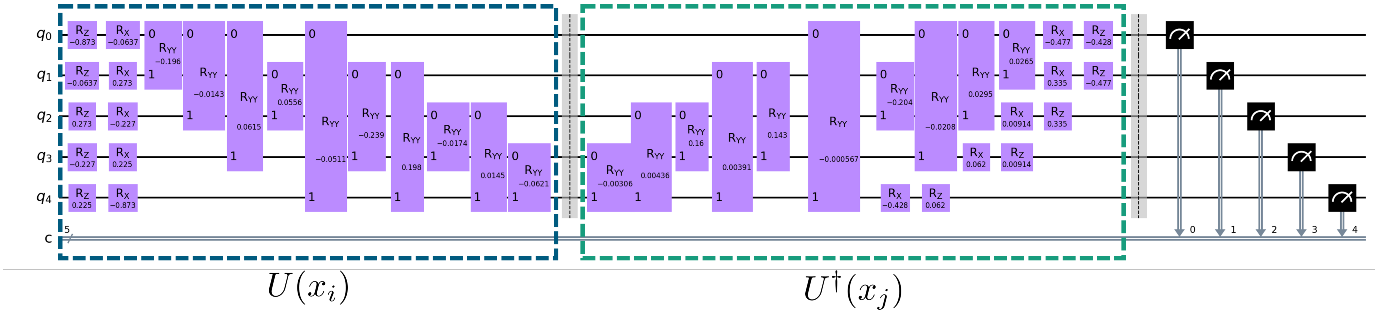

Fig. 1. QSVR circuit. Explanation in text.

are qc-QSVR (ours), QSVR, QAE, CSVR, and CAE. Each model's average performance across all datasets is represented by a dotted line, while the solid bars indicate the model's performance on individual datasets.

The comparison between our qc-QSVR and the simulated QSVR shows that the simulated model achieves a higher mean AUC (0.76 to 0.72) and performs better on 8 datasets, while the AUC is identical on the DoH dataset and the qc-QSVR achieves higher AUC on the CC and KDD datasets. The biggest performance gap between the simulated and the hardware model is 0.15 on the CoverT dataset. The average difference of the AUC between the simulated and the hardware model on the eight datasets, on which the simulated model performs better, is 0.06. From this we conclude that the QSVR model is suitable for operation on hardware, with only a slight decrease in classification performance.

The QAE achieves better results compared to **Tscharke**, which might be explained due to an update in qiskit and some optimizations on the code. The CSVR produces identical results as in **Tscharke**, which was expected because of the deterministic nature of the model. The CAE demonstrates a small improvement in performance over the results in **Tscharke** which can be attributed to the use of the *tanh* activation function in the last layer of the encoder.

shorten description of results. Instead explain why noise increases performance on CC, KDD and leads to same performance on DoH. Maybe bc noise makes model less prone to overfitting like regularization, find references

B. Influence of Noise on the Model Performance

The hardware results show that noise in NISQ devices affects our QSVR's performance. That is why in this section, we examine the influence of six noise channels on the performance of our model. The left two columns of Figure 3 show these results and reveal that, with a few exceptions, the effect of noise on our QSVR is small for real-world datasets. *Amplitude damping* shows notable behavior, with AUC for most datasets decreasing at $p = 0.1$ and $p = 0.2$, then partially recovering at higher p. Only URL, forest type, and DoH do not follow this trend, with URL showing the highest AUC for $0.1 < p < 0.3$.

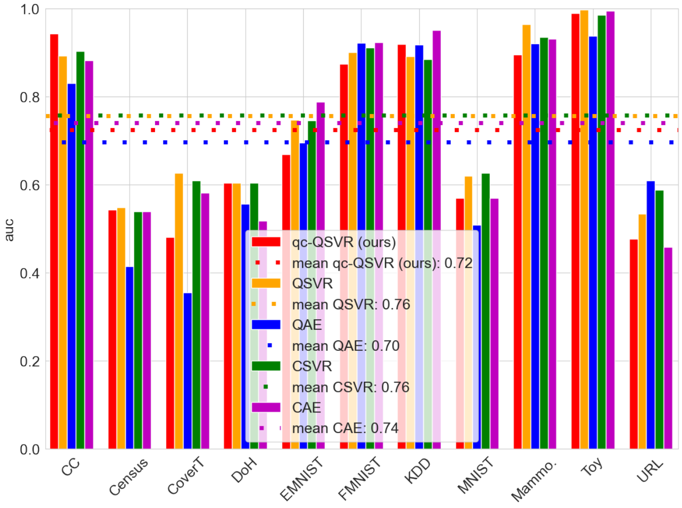

Fig. 2. Plot of area under the curve (AUC) for the different models on the evaluated data sets.

The presence of *miscalibration* noise significantly impacts model performance. For most datasets, excluding Census and Toy, the AUC begins to decline at $p = 0.05$, corresponding to an overrotation of approximately 3°. A large drop in AUC is visible for $p = 0.1$, however this equals an overrotation of about 6°, which signifies a huge miscalibration of the device. Overall, there is a clear trend of decreasing AUC as noise probability increases before reaching a plateau in certain datasets.

Considering the noise types *depolarizing*, *phase damping*, *phaseflip* and *bitflip*, we observe that the influence of noise on the model's performance is largely negligible for most datasets.

Only for the linearly separable dataset Toy (color lime), for which the noise-free model reaches an AUC of 1.0, we see a decrease in AUC with increasing noise probability for all noise types after surpassing a plateau for small noise probabilities. The plateau can be contributed to the separation gap of 0.4 between the classes in the Toy dataset.

All in all we observe that *depolarizing*, *phase damping*, *phaseflip* and *bitflip* noise have a small influence on our QSVR for real-world datasets and only *amplitude damping* and *miscalibration* have a strong effect on the performance of the model. We will leave the theoretical explanation for the QSVR's robustness against certain types of noise open for future work.

shorten description of figures. Instead try to explain model behavior under different types of noise.

C. Adversarial Robustness

AD is often used in security-critical areas such as credit card fraud detection or network intrusion detection. Hence, ML models used for AD must be robust against adversarial attacks. For lower dimensional, tabular data sets, it is possible that for high attack strengths, a sample might be transformed completely and factually to a sample of another class. In these cases the effect on the AUC might be exaggerated and must therefore be interpreted just as an upper bound for the drop performance. First, we consider the noise-free model and plot the obtained results in figure 4, which shows the AUC of the noise-free model under PGD attacks up to a strength of $\epsilon = 0.5$. The noise-free QSVR is highly vulnerable against adversarial attacks, visible by the drop in AUC for small attack strengths of $\epsilon = 0.01$ for all datasets but Toy. The largest decrease is in the AUC of the DoH dataset which decreases for the $\epsilon = 0.01$ attack from 0.67 by one order of magnitude to 0.06. With increasing attack strength, the AUC decreases further for all datasets until it approaches 0.0 for $p = 0.3$. From this we conclude that techniques for increasing the adversarial robustness of the model should be investigated.

The unchanged AUC of Toy for $\epsilon = 0.01$ can be explained due to the creation process of the dataset. The dataset was created to be linearly separable with a separation gap of 0.4 between the classes and hence, the datapoints must be shifted by a large distance in feature space to be misclassified. However, as the AUC drops to 0.0 already for $\epsilon = 0.3$, which is smaller than the separation gap, we conclude that the QSVR is susceptible to overfitting.

Second, the two right columns of Figure 3 show how the noisy QSVR models perform when attacked with adversarial perturbations of strength $\epsilon = 0.1$. The behavior of the attacked models with noise is strongly dependent on the dataset, noise type and noise strength. Generally, for noise levels below $p = 0.1$, the AUC drops, but at higher noise levels, it often recovers partly until it reaches a plateau. This recovery is most notable with *phase damping* and *amplitude damping* noise.

For the Toy dataset, the AUC drops sharply from 0.95 (noise-free) to 0.46 with 0.01 *depolarizing* noise. Other noise types similarly reduce the AUC at this low noise strength, indicating high sensitivity to minimal noise. The KDD dataset also shows an interesting behavior, its AUC improves from 0.43 for the noise-free model to 0.72 with 0.2 *depolarizing* noise and further to 0.75 at $p = 0.5$. This trend holds across most noise types.

The DoH dataset is an outlier, with an AUC of 0.0 across nearly all noise types and strengths. However, with *miscalibration* noise of strength 0.2, the AUC briefly spikes to 1.0 before dropping back to 0.0.

The observed trend of increasing AUC with increasing noise level for some datasets and noise types is likely due to the adversarial perturbations being drowned out in the noise, so that the noise is equalizing the effect of the carefully crafted perturbations. The models attacked with $\epsilon = 0.01$ exhibit similar behavior to the $\epsilon = 0.1$ attacks above. The $\epsilon = 0.3$ attacks also follow a similar pattern but lack the AUC drop at low noise levels, as the noise-free models' AUC is near zero. Results for $\epsilon = 0.01$ and $\epsilon = 0.3$ attacks are not shown here, as they do not provide new insights and can be obtained from the authors.

We conclude that adding noise to our QSVR does not improve the model's robustness against adversarial attacks. This finding aligns with prior research **noise˙influence˙winderl**, where the authors suggest that adding noise to QML models to increase the adversarial robustness is unlikely to be beneficial in practice.

shorten description of figures. Instead try to explain model behavior under different types of adv attacks and noise.

V. Conclusion and Outlook

First, we benchmarked our QSVR for semisupervised AD on 27-qubits IBM hardware, finding its AUC averages only slightly lower than the noise-free simulation's (0.72 compared to 0.76), even outperforming the noise-free simulation in two out of eleven datasets.

Second, the influence of six noise channels on the performance of the QSVR was evaluated, revealing that the effect of noise on our QSVR is small with three exceptions. Specifically, *amplitude damping* and high levels of *miscalibration* impair model performance across most datasets. Additionally, for the linearly separable Toy dataset, model performance declines with increasing noise probability across all noise types.

Finally, the adversarial robustness of the (noisy) model was tested, where we observe that the QSVR is highly vulnerable against adversarial attacks, and even small PGD attacks with a strength of $\epsilon = 0.01$ can reduce the model's AUC by up to one order of magnitude. Introducing noise to the model has no benefits for both the unattacked model's performance and its adversarial robustness. From this we conclude that our QSVR shows potential for semisupervised AD in the NISQ era, however, special attention should be paid to the vulnerability to adversarial attacks and certain types of noise.

Potential next steps for future work include scaling the model up to a larger number of qubits and theoretically evaluating the influence of noise on the QSVR. As we notice the vulnerability of these models to adversarial attacks, future work may also focus on techniques to increase the robustness and defend against such attacks.

Acknowledgment

The research is part of the Munich Quantum Valley, which is supported by the Bavarian state government with funds from the Hightech Agenda Bayern Plus.

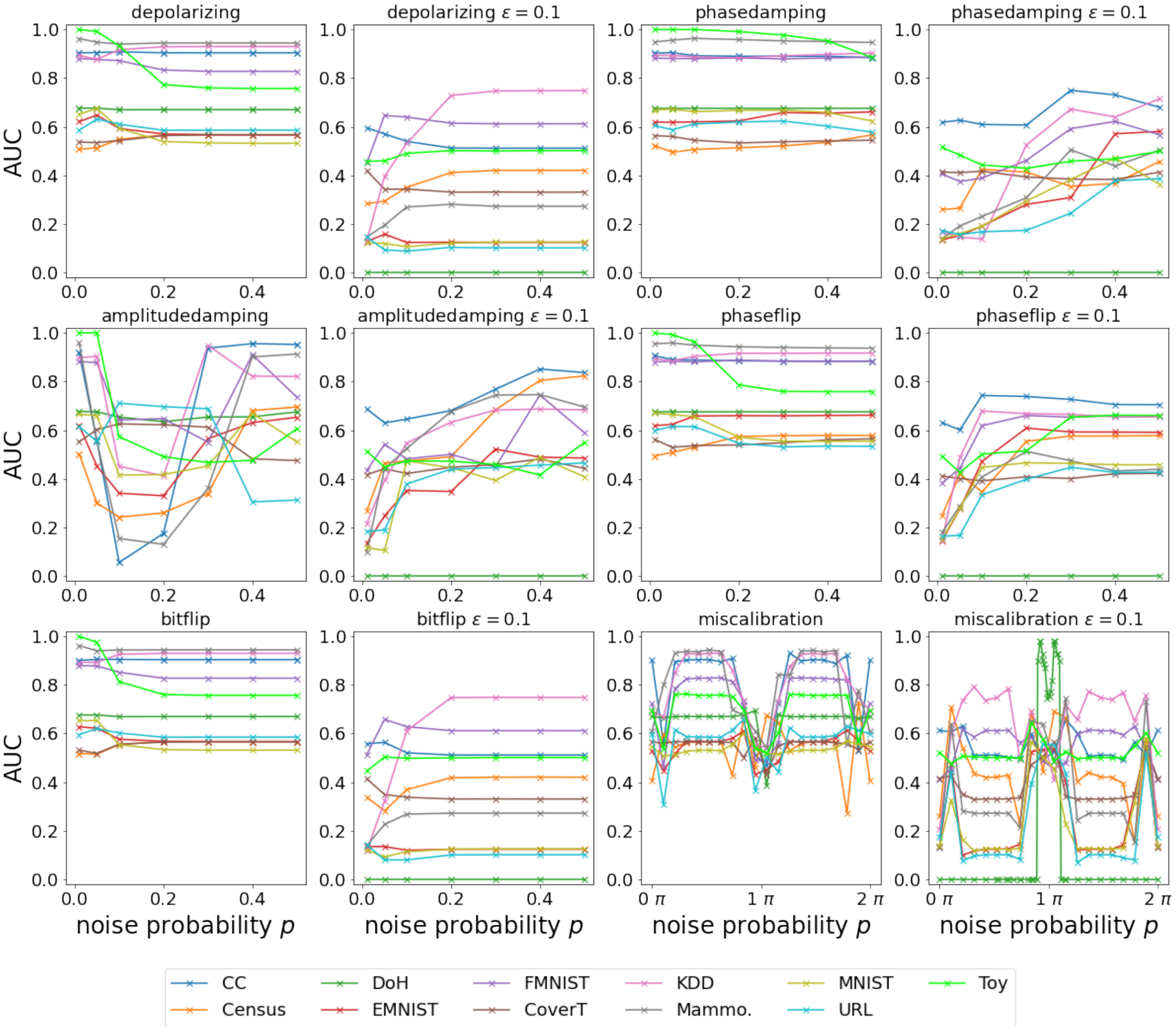

Fig. 3. Left two columns: Influence of six different noise types on the QSVR. Right two columns: Influence of six different noise types on the QSVR for adversarial attacks of strength $\epsilon = 0.1$

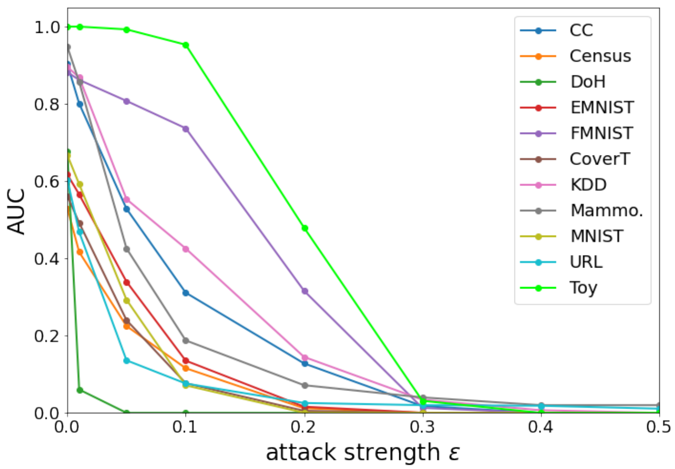

Fig. 4. Influence of adversarial attacks on the noise-free models.