Generating Universal Adversarial Perturbations for Quantum Classifiers

Gautham Anil[1*], **Vishnu Vinod**[1*], **Apurva Narayan**[2,3,4]

[1]Indian Institute of Technology Madras
[2]University of British Columbia
[3]University of Western Ontario
[4]University of Waterloo
{gauthamga.gga,vishnuvinod2001}@gmail.com, apurva.narayan@uwo.ca

Abstract

Quantum Machine Learning (QML) has emerged as a promising field of research, aiming to leverage the capabilities of quantum computing to enhance existing machine learning methodologies. Recent studies have revealed that, like their classical counterparts, QML models based on Parametrized Quantum Circuits (PQCs) are also vulnerable to adversarial attacks. Moreover, the existence of Universal Adversarial Perturbations (UAPs) in the quantum domain has been demonstrated theoretically in the context of quantum classifiers. In this work, we introduce QuGAP: a novel framework for generating UAPs for quantum classifiers. We conceptualize the notion of additive UAPs for PQC-based classifiers and theoretically demonstrate their existence. We then utilize generative models (QuGAP-A) to craft additive UAPs and experimentally show that quantum classifiers are susceptible to such attacks. Moreover, we formulate a new method for generating unitary UAPs (QuGAP-U) using quantum generative models and a novel loss function based on fidelity constraints. We evaluate the performance of the proposed framework and show that our method achieves state-of-the-art misclassification rates, while maintaining high fidelity between legitimate and adversarial samples.

Introduction

Rapid advancements in the field of machine learning have led to development of solutions to problems which were previously intractable (Jordan and Mitchell 2015; LeCun, Bengio, and Hinton 2015; He et al. 2015; Zhao et al. 2023; Silver et al. 2016). Concurrently, rapid advances are being achieved in the domain of quantum computing. While development of fault-tolerant large-scale quantum computers holds the potential to unravel solutions to classically hard problems (Shor 1994; Cao et al. 2019), quantum computing has applications even in the current NISQ (Noisy Intermediate Scale Quantum) era (Preskill 2018; Lau et al. 2022; Bharti et al. 2022).

A promising direction of research, termed quantum machine learning (QML) attempts to bridge the domains of machine learning and quantum computing in an attempt to leverage NISQ devices to enhance machine learning methodologies (Biamonte et al. 2017; Lloyd, Mohseni, and Rebentrost 2014; Rebentrost, Mohseni, and Lloyd 2014; Dallaire-Demers and Killoran 2018). Within QML, parametrized quantum circuits (PQCs), recognized as a quantum analogue of classical neural networks, have garnered significant attention in recent times (Cerezo et al. 2021; Benedetti et al. 2019). While the application domains where QML can provide an advantage are being explored, recent advances suggest that classification is a promising candidate. (Huang et al. 2021). The rise in prominence of PQC-based classifiers stems from their capacity to achieve performance comparable to classical neural networks in multiple use cases while utilizing far fewer parameters (Abbas et al. 2021; Schuld et al. 2020).

Classical machine learning classifiers are well-known to be susceptible to adversarial attacks that severely degrade performance (Xu et al. 2020; Akhtar and Mian 2018; Zhang et al. 2020). Recent studies reveal that PQC-based classifiers too, are susceptible to adversarial attacks, mirroring the vulnerabilities observed in classical settings (Liu and Wittek 2020; Lu, Duan, and Deng 2020). Akin to the classical ML , the existence of universal attacks has also been demonstrated in the realm of quantum classifiers (Gong and Deng 2022). In this work, we propose QuGAP (Quantum Generative Adversarial Perturbation), a framework to generate UAPs for quantum classifiers. Our main contributions are:

- We theoretically demonstrate the existence of *additive* UAPs for quantum classifiers used in the classification of amplitude-encoded classical data.
- We propose a strategy for generating *additive* UAPs using classical generative models and conduct experiments to validate the viability of the proposed approach.
- We propose a novel strategy for generating *unitary* UAPs using explicit fidelity constraints. We empirically evaluate the performance of the proposed framework and achieve state-of-the-art results.

Related Work

Adversarial Attacks on Neural Networks

The idea that carefully constructed adversarial samples, which differ only slightly from the true samples, could fool classical deep neural networks was first discussed in (Szegedy et al. 2014). After the conception of this idea, a

*These authors contributed equally.

number of attacks were proposed in the white-box setting, where the adversary has full access to the target classifier. The proposed attacks adopt different strategies, including gradient-based methods (Goodfellow, Shlens, and Szegedy 2015; Madry et al. 2019), optimization-based methods (Carlini and Wagner 2017; Liu et al. 2017) and generative model based methods (Xiao et al. 2019; Bai et al. 2021). However, these attacks are all input specific attacks as the perturbation applied on each input is different.

The notion of a "universal" adversarial perturbation (UAP) was initially introduced in (Moosavi-Dezfooli et al. 2017). The idea was to generate input-agnostic perturbations which can fool the target model in a classification problem across all classes. The idea of generating UAPs using generative networks was proposed in (Poursaeed et al. 2018). This method of generating UAPs showed a marked improvement over the iterative method proposed in (Moosavi-Dezfooli et al. 2017). Our work takes inspiration from the method proposed in (Poursaeed et al. 2018) to generate additive UAPs for quantum classifiers.

Adversarial Attacks on PQC-Based Classifiers

The vulnerability of quantum classifiers to adversarial attacks has been studied in (Liu and Wittek 2020; Lu, Duan, and Deng 2020). Quantum adversarial attacks and adversarial learning were experimentally demonstrated in (Ren et al. 2022). The effect of noise in making quantum classifiers robust to adversarial attacks was studied in (Du et al. 2021; Huang et al. 2023) whereas the effect of encoding schemes in protecting quantum classifiers was studied in (Gong et al. 2022). The transferability of attacks across classical and quantum classifiers was studied in (West et al. 2023). The existence of unitary UAPs for quantum classifiers has recently been theoretically demonstrated in (Gong and Deng 2022); they also propose an iterative scheme for generating UAPs. In this work, we propose a more effective framework for generating unitary UAPs for quantum classifiers.

Background

We use "ket" notation, like $|\psi\rangle$ for instance, to denote a complex column vector. Unless specified otherwise, $|\psi\rangle \in \mathbb{C}^d$ where $\mathbb{C}^d$ is the complex $d-$space and d is a positive integer. In general, a quantum state is represented by its density matrix σ. A pure quantum state is characterized by a density matrix of rank 1, and therefore can be equivalently represented by a complex column vector $|\psi\rangle$. For the rest of the paper, "quantum state" is used to refer to a pure quantum state represented by a column vector $|\psi\rangle$ unless specified otherwise. For a more detailed introduction to quantum states, density matrices and other concepts in quantum computing, we refer the readers to (Nielsen and Chuang 2010).

Quantum Classifiers We denote a PQC-based quantum classifier by $\mathcal{Q}$. For a review of such classifiers, we refer the readers to (Cerezo et al. 2021). The classifier $\mathcal{Q}$ takes in a quantum state $|\psi\rangle$ as input and outputs a label $\mathcal{Q}(|\psi\rangle) \in \{0, 1, ...k-1\}$ for a k-class classification problem. Note that the state $|\psi\rangle$ could be from a quantum dataset or could be from an encoded classical dataset. Analogous to classical classifiers, given a dataset $\mathcal{H}$ of samples $\{(|\psi_i\rangle, c_{\psi_i})\}_{i=1}^N$ where c_{ψ_i} denotes the assigned label of $|\psi_i\rangle$, we loosely say a quantum classifier $\mathcal{Q}$ is trained if it achieves $\mathcal{Q}(|\psi_i\rangle) = c_{\psi_i}$ for "most" of the samples in $\mathcal{H}$. The exact accuracy that can be achieved depends on the classifier as well as the dataset.

Classical UAPs Consider a classical dataset $\mathcal{D}$ of samples $\{(x_i, c_{x_i})\}_{i=1}^N$ where $x_i \in \mathbb{R}^d$ and c_{x_i} denotes the assigned label of x_i. Let $\mathcal{D}' \subset \mathcal{D}$ be the subset of all samples such that $\mathcal{M}(x_j) = c_{x_j}$ for every x_j in $\mathcal{D}'$ and a trained classical classifier $\mathcal{M}$. In this context, a UAP for $\mathcal{M}$ is an *additive perturbation* $\delta \in \mathbb{R}^d$ such that:

1. $\mathcal{M}(x_j + \delta) \neq c_{x_j}$
2. $||\delta||_p \leq \epsilon$

for "most" of x_j in $\mathcal{D}'$. Effectiveness of δ is measured by the misclassification rate, which is the fraction of such x_j in $\mathcal{D}'$. Note that $||.||_p$ denotes L_p norm and ϵ is a user-defined threshold controlling the strength of the perturbation.

Quantum UAPs As in the classical case, let $\mathcal{H}' \subset \mathcal{H}$ be the subset of the dataset such that for every $|\psi_j\rangle$ in $\mathcal{H}'$, $\mathcal{Q}(|\psi_j\rangle) = c_{\psi_j}$ for a trained quantum classifier Q. A quantum UAP for Q is a *unitary transformation* $U \in \mathbb{C}^{d\times d}$ such that:

1. $\mathcal{Q}(U|\psi_j\rangle) \neq c_{\psi_j}$
2. $|\psi_j\rangle$ and $U|\psi_j\rangle$ are "close"

for a significant fraction of ψ_j. Again, the effectiveness of U is measured by the misclassification rate. In the existing literature, closeness of $|\psi_j\rangle$ and $U|\psi_j\rangle$ is ensured by constraining U to be close to the identity matrix (Gong and Deng 2022); in this work, we explore the effect of using a fidelity-based loss function instead.

Encoding Schemes To use quantum classifiers for classifying classical data, it is necessary to encode the data into quantum states. There are a number of such proposed schemes for encoding classical data (LaRose and Coyle 2020). Amplitude encoding is one of the most commonly used data encoding schemes, due to the fact that it offers an exponential advantage in number of qubits required to encode classical data (compared to other quantum encoding schemes); to encode d-dimensional classical data we require only $\lceil \log_2 d \rceil$ qubits. The amplitude encoded state $|\psi_x\rangle$ for a classical sample x can be computed as $|\psi_x\rangle = \sum_{k=0}^{d-1} x^{(k)}|k\rangle$ where $x^{(k)}$ is the k^{th} element of the *normalized* $d-$dimensional vector x and $\{|k\rangle\}_{k=0}^{d-1}$ is the set of computational basis states. While in general the amplitudes corresponding to the computational basis states of a quantum state may be complex, in practice, for classical data, the amplitudes are constrained to be real values.

Adversarial Loss In order to train generative models for adversarial sample generation, we define an adversarial or fooling loss inspired by the formulations in (Xiao et al. 2019; Poursaeed et al. 2018).

For targeted attacks, the fooling loss is defined as:

$$\mathcal{L}_{fool,targeted} = \sum_{x \in \mathcal{D}} \mathcal{L}_{CE}(\hat{y}_x, t)$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss, $\hat{y}_x$ gives the prediction probabilities for input x and t denotes the target class. For untargeted attacks, two formulations exist: either the target class can be set as the least-likely class and targeted attack can be used or a separate fooling loss can be defined:

$$\mathcal{L}_{fool,untargeted} = -\sum_{x \in \mathcal{D}} \mathcal{L}_{CE}(\hat{y}_x, c_x)$$

where c_x is the label of input x. Both formulations were observed to have competitive performance in (Poursaeed et al. 2018). In our experiments, we use the latter formulation for untargeted attacks. We use $\mathcal{L}_{fool}$ to denote fooling loss in general. We also reiterate that the effectiveness of an attack on a dataset is measured by its misclassification rate: percentage of correctly predicted samples which are misclassified after the attack. This metric will be used throughout the paper to measure the effectiveness of adversarial attacks.

Additive UAPs

Motivation A proposed application of quantum classifiers is to carry out classification tasks on classical data encoded into quantum states (Benedetti et al. 2019). In such cases, we propose to generate additive UAPs for classical data motivated by the following reasons:

- Unlike a quantum dataset where the quantum states may be directly accessible, an adversary may not have access to the quantum states after encoding classical data as the encoded data is directly fed to the quantum classifier.
- The classical notion of additive UAPs does not directly carry over to the quantum domain because quantum states, unlike classical data, can only be perturbed using unitary transformations.

It is thus valuable to study the effect of classical additive attacks on quantum classifiers. For the rest of the discussion in this section, we assume amplitude encoding of classical data. A justification for this choice may be found in the previous section.

Existence of Additive UAPs

The key idea behind using additive perturbations is to take advantage of the normalization step in the amplitude encoding scheme. By applying a large enough perturbation to a sample and re-normalizing the resulting vector, it might be possible to project the sample to a different decision region of the quantum classifier. We provide an intuitive demonstration of this idea in the supplementary material.

To develop a theoretical framework for formally proving the existence of additive UAPs, we make the following assumptions:

Assumption 1 *The dataset under consideration is a classical dataset $\{(x_i, c_{x_i})\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^d$ are $d-$dimensional data samples, $c_{x_i} \in \{0, 1, ..., k-1\}$ are the labels for the $k-$class classification problem and N is the total number of labelled data samples.*

Assumption 2 *Encoding of classical data into quantum states is carried out using the amplitude encoding scheme. We abuse notation and use $|x\rangle$ to denote the amplitude encoded quantum state corresponding to the data sample x. We also assume x is normalized since this is necessary for amplitude encoding.*

Assumption 3 *The trained quantum classifier, $\mathcal{Q}$, is a noiseless PQC-based classifier with a total of $D+K$ qubits. $D = \lceil \log_2 d \rceil$ data qubits are used for encoding classical data and $K = \lceil \log_2 k \rceil$ ancillary qubits are used for measuring output probabilities. Inputs are padded with zeros to ensure $d = 2^D$.*

Assumption 4 *$\mathcal{Q}$ assigns a prediction $\hat{c}_x$ to an input sample x as follows: first, a global unitary transformation $U \in \mathbb{C}^{n \times n}$, where $n = 2^{D+K}$, is applied to $|x\rangle \otimes |0\rangle^{\otimes K}$ to obtain a state $|y\rangle$. The prediction is then computed as $\hat{c}_x = \arg\max\{|\langle \mathbf{1}_{2^D} \otimes i | y\rangle|^2\}_{i=0}^{k-1}$, where $\mathbf{1}_{2^D}$ is a vector in $\mathbb{C}^{2^D}$ with all its elements as 1 and $\{|i\rangle\}_{i=0}^{k-1}$ are the first k standard basis states of space $\mathbb{R}^{2^K}$ (j^{th} standard basis state is a vector with its j^{th} element equal to 1 and all other elements equal to 0).*

A note regarding notation: all vectors and matrices are zero-indexed. For instance, the first entry of vector x will be denoted as x_0 and the top-left entry of matrix U will be denoted as $U_{0,0}$. Also, $\|.\|$ denotes L_2 norm unless specified otherwise. With these assumptions and notations in place, we now state a few important lemmas and the main theorems. All detailed proofs are provided in the supplementary. First, we prove the following lemma:

Lemma 1 *The probability of an input x being classified as class c is given by $P(\hat{c}_x = c) = x^\dagger M^c x$ where, $x^\dagger$ denotes the conjugate transpose of x and $M^c \in \mathbb{C}^{d \times d}$ is a positive semi-definite matrix given by: $M^c_{ij} = \sum_{t=0}^{d-1} U^*_{k't+c,k'i} \; U_{k't+c,k'j}$ with * denoting the complex conjugate and $k' = 2^K$.*

Proof sketch: Straightforward to prove by expanding out and then rewriting the expression in Assumption 4.

Now, our objective is to examine whether a perturbation, when added to input samples, can cause the classifier to misclassify most of the input samples. Therefore, it is necessary to examine the effect of such a perturbation on output prediction probabilities. Towards this, we prove the following lemma:

Lemma 2 *Let $x, y \in \mathbb{R}^d$ such that $\|x\| = 1$ and $\|y\| = 1$. If $\|x - y\| \leq \epsilon$, where $\epsilon \in \mathbb{R}$, the probability of the point y being classified as c is bounded as $|P(\hat{c}_y = c) - P(\hat{c}_x = c)| \leq d \cdot \left(\epsilon^2 + 2\epsilon\right)$.*

Proof sketch: The key idea is to write y as $x + \delta$ for some δ and use Lemma 1 to express $P(\hat{c}_y = c)$ in terms of $P(\hat{c}_x = c)$ and some additional terms involving M^c. The expression can then be simplified using the triangle inequality and the fact that M^c is PSD. The result then follows from the fact that $\text{Tr}(M^c) \leq d$, where $\text{Tr}()$ denotes the trace operator.

Next, we explore how introducing a perturbation impacts the amplitude encoding of data samples. The following lemma is established for large perturbations:

Lemma 3 *For any $x \in \mathbb{R}^d$ with $\|x\| = 1$ on which a perturbation p is applied, the resultant vector is given by $p + x$. If we normalize this vector to obtain $y = \frac{p+x}{\|p+x\|}$, then we have $\left\| \frac{p}{\|p\|} - y \right\| \leq \sqrt{2 - 2\sqrt{1 - \frac{1}{\|p\|^2}}}$ whenever $\|p\| > 1$.*

Proof sketch: The expression of norm is first expanded out in terms of p and x. The bound can then be computed by maximizing the expression and imposing the condition that $\|p\| > 1$.

With Lemmas 2 and 3 in place, we are ready to state the following theorem for perturbations with norm greater than 1:

Theorem 1 *For an additive universal adversarial perturbation p applied on inputs of classifier $\mathcal{Q}$, a strength of perturbation $\|p\| \in \mathbb{R}$ will cause $\mathcal{Q}$ to classify all inputs as c (class to which $p/\|p\|$ belongs) if:*

$$\|p\| \geq \frac{2}{\epsilon_c \sqrt{4 - \epsilon_c^2}}$$

where ϵ_c is given by:

$$\epsilon_c = \sqrt{1 + \frac{1}{2d} \cdot \left(\hat{p}^T M^c \hat{p} - \hat{p}^T M^{c'} \hat{p} \right)} - 1$$

where $\hat{p} = p/\|p\|$ and c' is the class with highest output probability for $\hat{p}$ after c.

Proof sketch: A lower bound is first placed on the probability of classifying the input sample after perturbation as class c. Lemma 2 is then invoked to rewrite the bound in terms of ϵ_c. The bound is then related to $\|p\|$ by using Lemma 3.

By using Theorem 1, it can be concluded that a sufficiently large perturbation can cause the classifier $\mathcal{Q}$ to classify all input samples as belonging to the class of the perturbation. This result is then directly applicable to targeted UAP generation; a sample belonging to the target class can be chosen as the perturbation. This perturbation, when appropriately scaled, can cause $\mathcal{Q}$ to misclassify all samples as belonging to the target class. For untargeted attacks, the target class can be chosen as the least probable class in the dataset.

Next, we focus on the case where the norm of the perturbation is constrained. In such cases, it may not be possible to guarantee a result as strong as Theorem 1. However, it is still possible to characterize the set of inputs for which we can ensure a required classification. To establish such a characterisation, we state a more general version of Lemma 3:

Lemma 4 *For any $x \in \mathbb{R}^d$ with $\|x\| = 1$ on which a perturbation p, such that $\|p\| \geq \delta$, is applied, the resultant vector is given by $p + x$. If we normalize this vector to obtain $y = \frac{p+x}{\|p+x\|}$, then we have $\|\hat{p} - y\| \leq \sqrt{2 - 2\left(\frac{\delta + \hat{p}^T x}{\sqrt{\delta^2 + 2\delta \hat{p}^T x + 1}}\right)}$ where $\hat{p} = \frac{p}{\|p\|}$.*

Proof sketch: Similar to the proof of Lemma 3.

With Lemma 4 in place, we are ready to state the following theorem:

Theorem 2 *For an additive universal adversarial perturbation p applied on inputs of classifier $\mathcal{Q}$, with the constraint $\|p\| \leq \delta$, an input x is predicted as belonging to class c (class to which $p/\|p\|$ belongs) if any one of the following conditions hold true:*

1. *If $\delta \geq \frac{2}{\epsilon_c \sqrt{4 - \epsilon_c^2}}$*
2. *If $1 \leq \delta < \frac{2}{\epsilon_c \sqrt{4 - \epsilon_c^2}}$ and ($\hat{p}^T x \leq t_1$ or $\hat{p}^T x \geq t_2$)*
3. *If $\delta < 1$ and $\hat{p}^T x \geq t_2$*

where $\hat{p} = p/\|p\|$, thresholds t_1, t_2 are in $[-1, 1]$, $t_1 \leq t_2$ and t_1, t_2 are the solutions of the quadratic equation:

$$t^2 + 2\delta\epsilon' t + (\epsilon'(\delta^2 + 1) - 1) = 0$$

where $\epsilon' = \epsilon_c^2 - \frac{\epsilon_c^4}{4}$ (ϵ_c defined in Theorem 1).

Proof sketch: Similar to Theorem 1, except that now we place a bound on $\|p\|$. The bound is then used to compute allowed values of $\hat{p}^T x$ using Lemma 4.

With Theorem 1, we have established the existence of additive UAPs which can cause a quantum classifier to classify all samples as belonging to a target class. Further, Theorem 2 establishes a connection between the perturbation strength and the subset of input space which gets classified as belonging to a target class. Theorem 2 implies that for any given δ, there is a subset of vectors in $\mathbb{R}^d$ which get classified as belonging to class c.

Before moving forward, we would like to emphasize that the statements of both Theorem 1 and Theorem 2 involve sufficient conditions and not necessary conditions; it is possible that other perturbations exist which are "better" than the ones which satisfy conditions in Theorem 1 or 2. The objective of the developed theoretical framework is to prove the existence of additive UAPs for quantum classifiers at various perturbation strengths; no claim is made with respect to the optimality of the perturbations. A strategy for generating effective additive UAPs is discussed in the next section.

Generative Framework

As described above, it is not straightforward to determine the perturbation p, which causes the highest misclassification under a given norm constraint. The generation of such UAPs is further complicated by the fact that the effectiveness of a UAP also depends on how the samples are distributed and therefore on the dataset under consideration. Therefore, in this section, we introduce an effective framework for obtaining additive UAPs in a constrained-norm setting.

The proposed strategy for generating additive UAPs, denoted henceforth as QuGAP-A, for an amplitude-encoded classical dataset is illustrated in Figure 1. A brief description of the training procedure is given in the caption.

The key idea is to train the classical generator $\mathcal{G}$ to convert a given random vector z to an additive UAP z' for the dataset under consideration. Note that the generated UAP z' is scaled to ensure that the L_p norm is below a fixed threshold. The training is done by performing backpropagation and updating the parameters of $\mathcal{G}$ using the fooling loss $\mathcal{L}_{fool}$. Typically, $p = 2$ or $p = \infty$ is used. The gradient computation can either be done completely classically by simulating

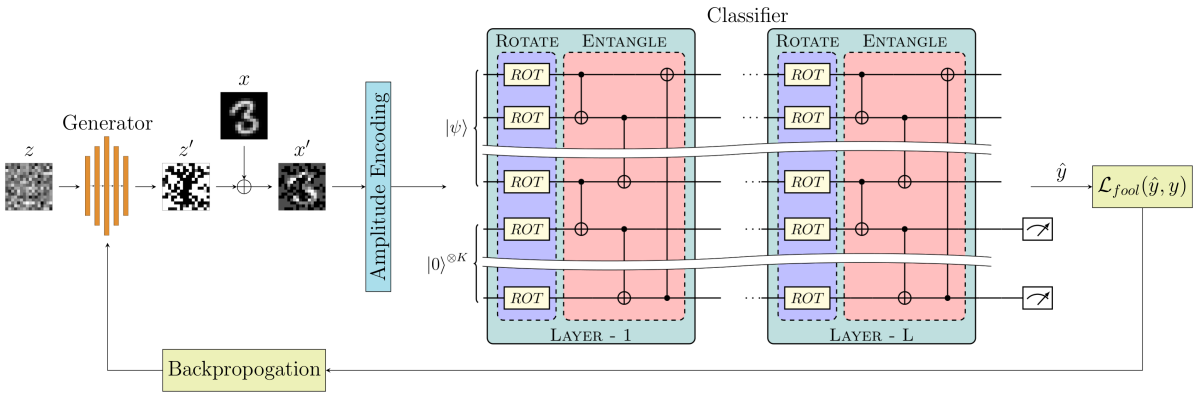
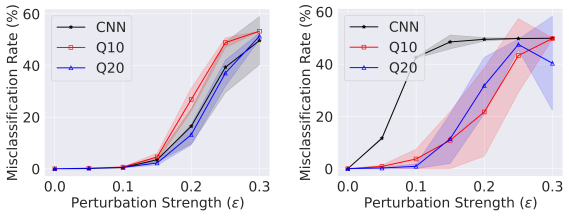


Figure 1: QuGAP-A: A framework for generating additive UAPs for quantum classifiers. A random vector z sampled from $\mathbb{R}^m$ is passed through a classical generative network. The generated perturbation z' is then scaled to impose the norm constraint and then added to an input sample x. The perturbed input sample x' is then amplitude-encoded and passed through the trained quantum classifier $\mathcal{Q}$. Output predictions from $\mathcal{Q}$ are then used to compute the fooling loss $\mathcal{L}_{fool}$. Gradients computed are backpropagated to update the generator parameters. The process is repeated for all input samples over multiple epochs.

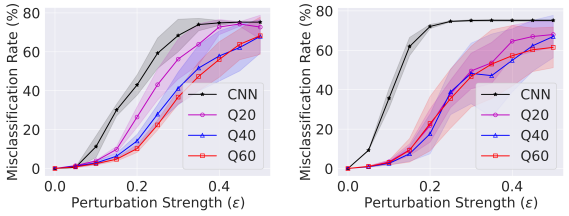
$\mathcal{Q}$ (straightforward once the parameters of $\mathcal{Q}$ are known), or it can be done in a hybrid fashion where gradients from $\mathcal{Q}$ can be directly computed using the parameter-shift method. Additional details such as the structure of $\mathcal{G}$, hyperparameters for training and software packages used as well as experiments for targeted attacks are detailed in the supplementary.

Once the generator $\mathcal{G}$ training is complete, we no longer require access to the quantum classifier to generate adversarial samples. This makes our attack a semi-whitebox attack (Xiao et al. 2019). The noise z used during training is passed to the trained generator to generate the attack. The generated perturbation is added to a clean input to generate an adversarial sample.



(a) MNIST: 2 Class

(b) FMNIST: 2 Class



(c) MNIST: 4 Class

(d) FMNIST: 4 Class

Figure 2: The misclassification rates for 16×16 MNIST and FMNIST using additive untargeted UAPs. We report results for binary classification between classes 0 and 1 and 4-class classification between classes $0, 1, 2$ and 3.

Experimental Results

We test the generative framework by attacking quantum classifiers of different depths trained on two tasks: binary classification and four-class classification, and two datasets: MNIST (LeCun, Cortes, and Burges 2010) and FMNIST (Xiao, Rasul, and Vollgraf 2017). We downsample both datasets to 16×16 pixels due to computational limitations. We constrain the L_∞ norm of the attack to be less than ϵ. UAPs are generated with different values of the bound ϵ for the classifier. The UAP generation is stochastic in nature as we sample a random vector z initially. To enable a fair evaluation, we test each setting 10 times to ensure reasonably small standard deviations. While dealing with images, an additional step of clipping the data after perturbation is required to ensure that the pixels are in the range $[0, 1]$. The implications of this additional step on the UAP generation as well as more specifics regarding the experiments, such as the training procedure and hyperparameters used, are detailed in the supplementary.

To illustrate the effectiveness of our framework, we plot the misclassification rates (along with standard deviation) for the experiments in Figure 2. CNN represents a classical convolutional neural network, whereas Q10, Q20, Q40 and Q60 represent PQCs with depths 10, 20, 40 and 60, respectively. As expected, the misclassification rates increase with an increase in ϵ. Note that the misclassification rate plateaus around 50% for binary classification and around 75% for 4-class classification. This is in line with the developed theory; for a $k-$class classification problem, assuming even distribution of samples across all classes and an ideal classifier, the misclassification rate when all samples are classified as one particular class will be $\frac{k-1}{k}$. A naive approach to generate additive UAPs for quantum classifiers would be to generate perturbations for a classical classifier and transfer them to quantum classifiers. However, empirically such attacks do not transfer well. Transferability studies are presented in the supplementary material.

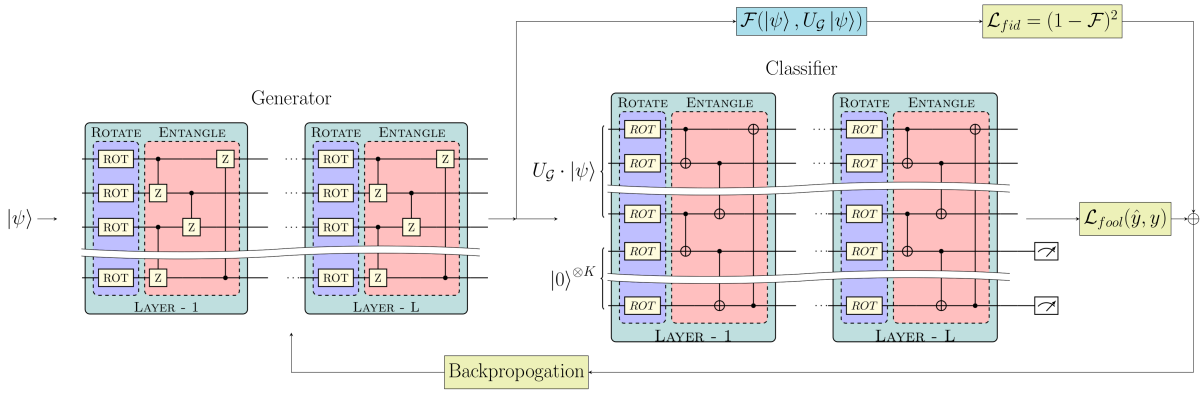


Figure 3: QuGAP-U: A framework for generating unitary UAPs for quantum classifiers. The quantum generator $\mathcal{G}_{\mathcal{Q}}$ takes in an input state $|\psi_i\rangle$ and transforms it into a perturbed state $|\phi_i\rangle = U_{\mathcal{G}}|\psi_i\rangle$. The fidelity between $|\psi_i\rangle$ and $|\phi_i\rangle$ is computed from which $\mathcal{L}_{fid}$ is calculated. $|\phi_i\rangle$ is also passed through a trained quantum classifier $\mathcal{Q}$ to compute $\mathcal{L}_{fool}$. Gradients are computed using the total loss $\mathcal{L}_{fool} + \alpha\mathcal{L}_{fid}$ and used to update the parameters of $\mathcal{G}_{\mathcal{Q}}$ over all training samples for multiple epochs

Unitary UAPs

In this section, we propose a strategy for generating unitary transformations in the quantum domain which can act as UAPs. In contrast to the previous section where additive perturbations are added to classical data before encoding, in this section we focus on the case where adversarial manipulations directly transform the quantum state. This enables us to attack encoded classical data as well as quantum data.

Motivation

The existence of quantum UAPs has been theoretically established in (Gong and Deng 2022); furthermore they employ the iterative qBIM (Lu, Duan, and Deng 2020) algorithm to generate quantum UAPs. However, a drawback of the algorithm is that the search space for the global unitary U is limited as U is constrained to be a product of local unitaries near the identity matrix. This limitation is further worsened by the fact that only a single variational layer is used to generate the UAP.

We propose a novel strategy to overcome these limitations; instead of constraining U to be a product of local unitaries near identity, we implement a fidelity-based loss function to control the perturbation strength. Further, a PQC-based generative network is used instead of a single variational layer to search over a larger space of unitaries.

Proposed Framework

The proposed framework for generating unitary UAPs, denoted as QuGAP-U, is illustrated in Figure 3. A brief description of the training procedure is given in the caption.

We introduce a novel loss function $\mathcal{L}_U$ of the form:

$$\mathcal{L}_U = \mathcal{L}_{fool} + \alpha\mathcal{L}_{fid}$$

where $\mathcal{L}_{fool}$ is the fooling loss and $\mathcal{L}_{fid}$ is a fidelity-based loss of the form:

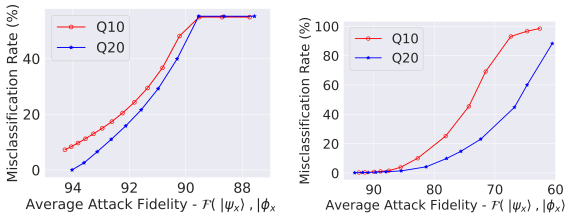
$$\mathcal{L}_{fid} = (1 - \mathcal{F}(|\psi\rangle, |\phi\rangle))^2$$

$\mathcal{F}(|\psi\rangle, |\phi\rangle) = |\langle\psi|\phi\rangle|^2$ is the fidelity between the input state and the perturbed state. α is a hyper-parameter controlling the trade-off between misclassification and fidelity; a higher value of α generates UAPs which ensure a higher fidelity of perturbed states but may have lower misclassification rates. Some details regarding implementation on quantum hardware: since $|\psi_i\rangle$ and $|\phi_i\rangle$ are pure states, the fidelity can be computed exactly using the SWAP test (Stein et al. 2021). The gradients can also be computed using parameter-shift rule (Schuld et al. 2019; Mitarai et al. 2018).

Classical Simulation

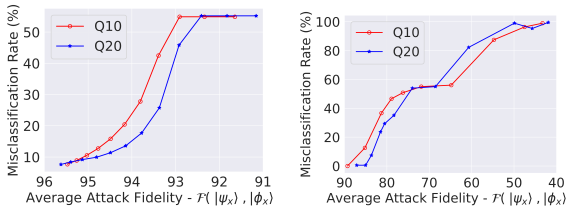
To empirically verify the viability of the proposed framework, we simulate it classically by optimizing for a unitary $U_{\mathcal{G}_{\mathcal{Q}}}$ which acts as a proxy for the quantum generator $\mathcal{G}_{\mathcal{Q}}$. The objective then is to learn $U_{\mathcal{G}_{\mathcal{Q}}}$ which minimizes the loss $\mathcal{L}_{\mathcal{Q}}$. To perform the optimization, we take inspiration from (Kiani et al. 2022) and project the matrix learned after each step of gradient descent into the space of unitary matrices. The simulation is done on two datasets: MNIST and Transverse-field Ising Model (TIM) Dataset. The synthetic TIM dataset maps the states of the transverse-field Ising model described in (Pfeuty 1970) to the phase of the system (ferromagnetic or paramagnetic). We model this physical system as a binary classification task on pure quantum data. More details regarding the TIM dataset as well as the complete pseudocode for classical optimization are given in the supplementary.

The results are illustrated in Figure 4. The plots validate the hypothesis that by varying the value of α, we can achieve fine-grained control over the fidelity of the perturbed states produced by the learned UAP. We also note that while the misclassification rate approaches 100% for MNIST classification as you decrease α, misclassification rate for TIM classification stays at around 54%. An explanation for this disparity is given in (Gong and Deng 2022); the maximum allowed misclassification rate has an upper bound which depends on the dataset distribution.



(a) TIM: 2 Class (b) MNIST: 2 Class

Figure 4: Classical simulation of a framework for generating unitary UAPs. Misclassification rates for TIM binary classification are given in (a), and for 8x8 downsampled MNIST binary classification are given in (b). Achieving competitive misclassification rates for MNIST results in much lower attack fidelities.



(a) TIM: 2 class (b) MNIST: 2 class

Figure 5: Misclassification rate evolution of QuGAP-U on classifiers with depths 10(Q10) and 20(Q20). (a) TIM dataset; (b) 8×8 downsampled MNIST dataset

PQC Simulation

In practice, quantum UAPs are implemented using PQCs. While we have demonstrated the effectiveness of the proposed approach through classical simulation in the previous section, it must be noted that PQCs, by construction, have access to only local unitaries. In such a scenario, it has been shown that for constructing an arbitrary unitary in a Hilbert space of dimension d, one would require $\mathcal{O}(d^2)$ gates (Shende, Bullock, and Markov 2005). Therefore, we expect the depth of the PQC to have a significant impact in determining the quality of the generated UAP.

Fidelity constraint	TIM		MNIST	
	qBIM	QuGAP-U	qBIM	QuGAP-U
0.95	3.18	**10.54**	0.00	0.00
0.90	6.37	**54.88**	0.00	0.00
0.85	9.44	**54.88**	0.00	**12.56**
0.80	13.39	**54.88**	0.00	**36.65**
0.75	16.68	**54.88**	0.05	**50.78**
0.70	23.27	**54.88**	0.15	**54.97**

Table 1: Comparison of misclassification rates for a PQC classifier of depth 10 trained on TIM and MNIST datasets after qBIM and QuGAP attacks. Fidelity constraint is the minimum average fidelity to be maintained while attacking.

We repeat the classical simulation experiments with a PQC-based quantum generative model, implemented using the Pennylane library (Bergholm et al. 2022). For benchmarking performance, we compare the performance of our method with the qBIM-based approach proposed in (Gong and Deng 2022). Tests are run for binary classification on the TIM dataset and an 8×8 downsampled version of MNIST. QuGAP-U uses a quantum generator with 30 layers for the TIM dataset and 200 layers for the MNIST dataset. The number of parameters in the generative models is chosen to be around d^2 in each case. Here d denotes the Hilbert space dimension or equivalently the number of input features. The effect of varying generator depth on the performance of QuGAP-U is detailed in the supplementary. Note that qBIM can utilize only a single variational layer; deeper circuits will significantly degrade the fidelity of the generated samples as the method does not explicitly rely on fidelity constraints. The benchmarking results are presented in Table 1. We observe that the performance of qBIM is much poorer than QuGAP-U, possibly because QuGAP-U utilizes deeper quantum generators and hence has a significantly larger search space. A more detailed analysis, as well as further discussion, can be found in the supplementary.

QuGAP-U clearly outperforms qBIM, the only other existing method for generating unitary UAPs, on both tasks, thereby achieving state-of-the-art performance in unitary UAP generation. We also observe that for the TIM dataset, misclassification saturates at 90% fidelity. This is because the misclassification rate rises drastically in the 90% - 95% fidelity range, as evidenced in Figure 5 (a). We also note that the performance closely matches the classical simulation (Figure 4 (a)) for the TIM dataset. We plot the performance of QuGAP-U on MNIST in Figure 5 (b). While TIM can be attacked with a quantum generator depth of just 30, a generator depth of around 200 is required to achieve good results for MNIST. Even then, the attacks are weaker than the classical simulation (Figure 4 (b)). This observation further supports our hypothesis that higher dimensional datasets require deeper PQCs for generating effective unitary attacks.

Discussion and Conclusion

With the rising prominence of quantum classifiers, it is necessary to study and mitigate the effect of adversarial attacks on such classifiers. In this work, we have analyzed universal adversarial perturbations in the context of quantum classifiers. We theoretically proved existence of additive UAPs and proposed a framework for UAP generation. We then established a novel framework for generating unitary UAPs and empirically demonstrated its advantages over existing methods.

A natural extension to our work would be to examine the effect of additive perturbations on encoding schemes other than amplitude encoding. Moreover, while we have designed the framework with considerations for practical implementations, our experiments are limited to simulations. Implementation of the proposed schemes on actual quantum computers may also be a worthwhile avenue for future research. Finally, it might be interesting to analyze the impact of quantum noise on the performance of the generated attacks.

Ethics Statement

In this work, we introduce algorithms to generate Universal Adversarial Perturbations (UAPs) for quantum classifiers which act on amplitude-encoded classical, and quantum datasets. Using the methods we propose, adversaries may tailor adversarial attacks which show reasonable efficiency even on real-world noisy quantum classifiers. However, due to the limited presence of PQC-based quantum classifiers in real-world applications at the time of publication of this work, we believe that our work does not pose any immediate security threats. On the other hand, the theoretical and empirical demonstrations of the efficacy of such attacks in our research, pre-emptively highlights the need to develop effective defense strategies capable of mitigating such attacks. We also concretely conceptualize the two types of UAPs for quantum classifiers: Additive UAPs (generated using QuGAP-A) and Unitary UAPs (generated using QuGAP-U); allowing future work in developing defense strategies to be broadly centered around these areas. To the best of our knowledge our work does not raise any ethical concerns, other than those addressed above.

Acknowledgements

The research presented in this work was done partially while the authors were at the Intelligent Data Science Lab, University of British Columbia. We thank the anonymous reviewers for their feedback. We also thank the Digital Research Alliance of Canada for access to computational resources. This work was supported by the MITACS Globalink Research Internship award 2022 and by the NSERC Discovery Grant No. RGPIN-2019-05163. All source code used for this research may be found at: https://github.com/Idsl-group/QuGAP along with links to the supplementary material.

References

Abbas, A.; Sutter, D.; Zoufal, C.; Lucchi, A.; Figalli, A.; and Woerner, S. 2021. The power of quantum neural networks. *Nature Computational Science*, 1(6): 403–409.

Akhtar, N.; and Mian, A. 2018. Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey. arXiv:1801.00553.

Bai, T.; Zhao, J.; Zhu, J.; Han, S.; Chen, J.; Li, B.; and Kot, A. 2021. Ai-gan: Attack-inspired generation of adversarial examples. In *2021 IEEE International Conference on Image Processing (ICIP)*, 2543–2547. IEEE.

Benedetti, M.; Lloyd, E.; Sack, S.; and Fiorentini, M. 2019. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4): 043001.

Bergholm, V.; Izaac, J.; Schuld, M.; Gogolin, C.; Ahmed, S.; Ajith, V.; Alam, M. S.; Alonso-Linaje, G.; AkashNarayanan, B.; Asadi, A.; Arrazola, J. M.; Azad, U.; Banning, S.; Blank, C.; Bromley, T. R.; Cordier, B. A.; Ceroni, J.; Delgado, A.; Matteo, O. D.; Dusko, A.; Garg, T.; Guala, D.; Hayes, A.; Hill, R.; Ijaz, A.; Isacsson, T.; Ittah, D.; Jahangiri, S.; Jain, P.; Jiang, E.; Khandelwal, A.; Kottmann, K.; Lang, R. A.; Lee, C.; Loke, T.; Lowe, A.; McKiernan, K.; Meyer, J. J.; Montañez-Barrera, J. A.; Moyard, R.; Niu, Z.; O'Riordan, L. J.; Oud, S.; Panigrahi, A.; Park, C.-Y.; Polatajko, D.; Quesada, N.; Roberts, C.; Sá, N.; Schoch, I.; Shi, B.; Shu, S.; Sim, S.; Singh, A.; Strandberg, I.; Soni, J.; Száva, A.; Thabet, S.; Vargas-Hernández, R. A.; Vincent, T.; Vitucci, N.; Weber, M.; Wierichs, D.; Wiersema, R.; Willmann, M.; Wong, V.; Zhang, S.; and Killoran, N. 2022. PennyLane: Automatic differentiation of hybrid quantum-classical computations. arXiv:1811.04968.

Bharti, K.; Cervera-Lierta, A.; Kyaw, T. H.; Haug, T.; Alperin-Lea, S.; Anand, A.; Degroote, M.; Heimonen, H.; Kottmann, J. S.; Menke, T.; et al. 2022. Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics*, 94(1): 015004.

Biamonte, J.; Wittek, P.; Pancotti, N.; Rebentrost, P.; Wiebe, N.; and Lloyd, S. 2017. Quantum machine learning. *Nature*, 549(7671): 195–202.

Cao, Y.; Romero, J.; Olson, J. P.; Degroote, M.; Johnson, P. D.; Kieferová, M.; Kivlichan, I. D.; Menke, T.; Peropadre, B.; Sawaya, N. P.; et al. 2019. Quantum chemistry in the age of quantum computing. *Chemical reviews*, 119(19): 10856–10915.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (sp)*, 39–57. IEEE.

Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S. C.; Endo, S.; Fujii, K.; McClean, J. R.; Mitarai, K.; Yuan, X.; Cincio, L.; and Coles, P. J. 2021. Variational quantum algorithms. *Nature Reviews Physics*, 3(9): 625–644.

Dallaire-Demers, P.-L.; and Killoran, N. 2018. Quantum generative adversarial networks. *Physical Review A*, 98(1).

Du, Y.; Hsieh, M.-H.; Liu, T.; Tao, D.; and Liu, N. 2021. Quantum noise protects quantum classifiers against adversaries. *Physical Review Research*, 3(2): 023153.

Gong, W.; and Deng, D.-L. 2022. Universal adversarial examples and perturbations for quantum classifiers. *National Science Review*, 9(6): nwab130.

Gong, W.; Yuan, D.; Li, W.; and Deng, D.-L. 2022. Enhancing Quantum Adversarial Robustness by Randomized Encodings. arXiv:2212.02531.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. arXiv:1412.6572.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, 1026–1034.

Huang, H.-Y.; Broughton, M.; Mohseni, M.; Babbush, R.; Boixo, S.; Neven, H.; and McClean, J. R. 2021. Power of data in quantum machine learning. *Nature Communications*, 12(1): 2631.

Huang, J.-C.; Tsai, Y.-L.; Yang, C.-H. H.; Su, C.-F.; Yu, C.-M.; Chen, P.-Y.; and Kuo, S.-Y. 2023. Certified Robustness of Quantum Classifiers against Adversarial Examples through Quantum Noise. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

Jordan, M. I.; and Mitchell, T. M. 2015. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245): 255–260.

Kiani, B.; Balestriero, R.; LeCun, Y.; and Lloyd, S. 2022. projUNN: Efficient method for training deep networks with unitary matrices. *Advances in Neural Information Processing Systems*, 35: 14448–14463.

LaRose, R.; and Coyle, B. 2020. Robust data encodings for quantum classifiers. *Physical Review A*, 102(3): 032420.

Lau, J. W. Z.; Lim, K. H.; Shrotriya, H.; and Kwek, L. C. 2022. NISQ computing: where are we and where do we go? *AAPPS Bulletin*, 32(1): 27.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *nature*, 521(7553): 436–444.

LeCun, Y.; Cortes, C.; and Burges, C. 2010. MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2.

Liu, N.; and Wittek, P. 2020. Vulnerability of quantum classification to adversarial perturbations. *Physical Review A*, 101(6).

Liu, Y.; Chen, X.; Liu, C.; and Song, D. 2017. Delving into Transferable Adversarial Examples and Black-box Attacks. arXiv:1611.02770.

Lloyd, S.; Mohseni, M.; and Rebentrost, P. 2014. Quantum principal component analysis. *Nature Physics*, 10(9): 631–633.

Lu, S.; Duan, L.-M.; and Deng, D.-L. 2020. Quantum adversarial machine learning. *Physical Review Research*, 2(3).

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2019. Towards Deep Learning Models Resistant to Adversarial Attacks. arXiv:1706.06083.

Mitarai, K.; Negoro, M.; Kitagawa, M.; and Fujii, K. 2018. Quantum circuit learning. *Physical Review A*, 98(3): 032309.

Moosavi-Dezfooli, S.-M.; Fawzi, A.; Fawzi, O.; and Frossard, P. 2017. Universal Adversarial Perturbations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 86–94.

Nielsen, M. A.; and Chuang, I. L. 2010. *Quantum computation and quantum information*. Cambridge university press.

Pfeuty, P. 1970. The one-dimensional Ising model with a transverse field. *Annals of Physics*, 57(1): 79–90.

Poursaeed, O.; Katsman, I.; Gao, B.; and Belongie, S. 2018. Generative Adversarial Perturbations. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4422–4431.

Preskill, J. 2018. Quantum Computing in the NISQ era and beyond. *Quantum*, 2: 79.

Rebentrost, P.; Mohseni, M.; and Lloyd, S. 2014. Quantum Support Vector Machine for Big Data Classification. *Physical Review Letters*, 113(13).

Ren, W.; Li, W.; Xu, S.; Wang, K.; Jiang, W.; Jin, F.; Zhu, X.; Chen, J.; Song, Z.; Zhang, P.; et al. 2022. Experimental quantum adversarial learning with programmable superconducting qubits. *Nature Computational Science*, 2(11): 711–717.

Schuld, M.; Bergholm, V.; Gogolin, C.; Izaac, J.; and Killoran, N. 2019. Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3): 032331.

Schuld, M.; Bocharov, A.; Svore, K. M.; and Wiebe, N. 2020. Circuit-centric quantum classifiers. *Physical Review A*, 101(3).

Shende, V. V.; Bullock, S. S.; and Markov, I. L. 2005. Synthesis of quantum logic circuits. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, 272–275.

Shor, P. W. 1994. Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In *Proceedings 35th annual Symposium on Foundations of Computer Science*, 124–134. IEEE.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature*, 529(7587): 484–489.

Stein, S. A.; Baheri, B.; Chen, D.; Mao, Y.; Guan, Q.; Li, A.; Fang, B.; and Xu, S. 2021. Qugan: A quantum state fidelity based generative adversarial network. In *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 71–81. IEEE.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2014. Intriguing properties of neural networks. arXiv:1312.6199.

West, M. T.; Erfani, S. M.; Leckie, C.; Sevior, M.; Hollenberg, L. C. L.; and Usman, M. 2023. Benchmarking adversarially robust quantum machine learning at scale. *Physical Review Research*, 5(2).

Xiao, C.; Li, B.; Zhu, J.-Y.; He, W.; Liu, M.; and Song, D. 2019. Generating Adversarial Examples with Adversarial Networks. arXiv:1801.02610.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. arXiv:1708.07747.

Xu, H.; Ma, Y.; Liu, H.-C.; Deb, D.; Liu, H.; Tang, J.-L.; and Jain, A. K. 2020. Adversarial attacks and defenses in images, graphs and text: A review. *International Journal of Automation and Computing*, 17: 151–178.

Zhang, W. E.; Sheng, Q. Z.; Alhazmi, A.; and Li, C. 2020. Adversarial Attacks on Deep-Learning Models in Natural Language Processing: A Survey. *ACM Transactions on Intelligent Systems and Technology*, 11(3): 1–41.

Zhao, W. X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; Du, Y.; Yang, C.; Chen, Y.; Chen, Z.; Jiang, J.; Ren, R.; Li, Y.; Tang, X.; Liu, Z.; Liu, P.; Nie, J.-Y.; and Wen, J.-R. 2023. A Survey of Large Language Models. arXiv:2303.18223.