CODE AVAILABILITY

All the code that was used in this project is available following the anonymous link `https://osf.io/by5dk/?view_only=5688cba7b13d44479f76e13e01d28d75`

A RELATED WORK

A recent work Schreiber et al. (2022) independently proposed a similar approach where classical surrogate methods approximate VQCs. The difference with this work is the necessity of having access to all Ω, the totality of the frequencies of the VQC considered, without sampling from them.

Indeed, if Ω is known, the coefficients a_ω and b_ω of the VQC function (see Eq.10) can be easily fitted by solving the classical least square problem. Namely, one determines $\mathbf{w}^*$ such that

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{M} \sum_{i=1}^{M} |\mathbf{w}^T \phi(x_i) - y_i|^2 + \lambda_0 ||\mathbf{w}||^2 \tag{2}$$

where $\phi(x) = \begin{bmatrix} cos(\omega x) \\ sin(\omega x) \end{bmatrix}_{\omega \in \Omega}$, and λ_0 is the regularisation parameter. As explained in the previous section, with a dataset of M points, this can be solved exactly using matrix inversion in $\mathcal{O}(M|\Omega|^2 + |\Omega|^3)$ operations if $M \geq 2|\Omega|$. If the inequality is not fulfilled or if $|\Omega|^3$ is too big, one would use stochastic gradient descent instead of matrix inversion.

However, this method assumes that Ω, and is not too large, which will usually be the case as we shown in section B.2. One should also be able to enumerate all individual frequencies $\omega \in \Omega$. Moreover, as we will show the redundancy of some frequencies in Ω has a key importance, which is not captured by such a method.

For completeness, we note from the seminal work Schuld (2021) that the author briefly mentions the idea of approximating kernels with RFF. Similarly, in a more recent work Peters & Schuld (2022), the authors mention RFF as a sampling strategy on VQCs with shift invariant kernels, without further details.

B PRELIMINARIES ON VARIATIONAL QUANTUM CIRCUIT FOR MACHINE LEARNING

B.1 DEFINITIONS

We consider a standard ML task where a function f, named *model*, must be optimized to map data points to their target values. The data used of the training is made of M points $x = (x_1, \ldots, x_d)$ in $\mathcal{X} = \mathbb{R}^d$ along with their target values y in $\mathcal{Y} = \mathbb{R}$. We define a *quantum model* as the family of parametrized functions $f : (\mathcal{X}, \Theta) \longrightarrow \mathcal{Y}$, such that

$$f(x;\theta) = \langle 0|U(x;\theta)^\dagger O U(x;\theta)|0\rangle \tag{3}$$

where $U(x;\theta)$ is a unitary that represents the parametrized quantum circuits, θ represents the trainable parameters from a space Θ, and O is an observable. We can always describe the parametrized quantum circuit as a series of two alternating layers. The first are called *encoding* layers as they only depend on input data values, whereas the *trainable* layers depend on internal parameters that are optimized during training. A typical instance of a layer is illustrated in figure 2. Note that an actual layer structure is not mandatory, since any circuit can be sliced into alternating sequences of encoding and training blocks (even if containing a single gate).

Any quantum unitary implements the evolution of a quantum system under a Hamiltonian. Thus, we choose to write the ℓ^{th} encoding gates as $\exp(-ix_i H_\ell)$, where x_i is one of the d components of x, and H_ℓ is a Hamiltonian matrix of size 2^p if p is the number of qubits this gates acts on. We will note L the number of encoding gates for each dimension of x (the same for each dimension, for notation simplicity).

In this framework, the aim is to find the optimal mapping between data points and their target values. This is done by optimizing the parameters θ to find the best guess f^* such that

$$f^* = \arg\min_{\theta} \frac{1}{M} \sum_{i=1}^{M} l(f(x_i; \theta), y_i) \tag{4}$$

where l is a cost function adapted to the task. For a standard regression tasks, we can choose $l(z, y) = |z - y|^2$.

B.2 Quantum Models are Large Fourier Series

We know since Schuld et al. (2021) that the family of quantum models defined in Eq.(3) can be rewritten as a Fourier series:

$$f(x; \theta) = \sum_{\omega \in \Omega} c_\omega e^{i\omega x} \tag{5}$$

where the spectrum Ω of frequencies is determined by the ensemble of eigenvalues of the encoding Hamiltonians and the coefficients c_ω depend on the parametrized ansatz, as pictured in figure 2.

In order to familiarize the reader with the structure of the spectrum, we explicitly build Ω in the case of a one dimensional data input ($\mathcal{X} = \mathbb{R}$) and with a variational circuit containing only L encoding gates. The accessible frequency spectrum Ω is the ensemble of all the differences between all possible sums of the eigenvalues of the encoding gates as shown in figure 2.

We note λ_ℓ^k the k^{th} eigenvalue of the ℓ^{th} encoding Hamiltonian H_ℓ. We use the multi-index $\boldsymbol{i} = (i_1, \ldots, i_L)$ indicating which eigenvalue is taken from each encoding Hamiltonian. We define $\Lambda_{\boldsymbol{i}}$ as

$$\Lambda_{\boldsymbol{i}} = \lambda_1^{i_1} + \cdots + \lambda_L^{i_L} \tag{6}$$

Finally, we can express the set of frequencies as:

$$\Omega = \left\{ \Lambda_{\boldsymbol{i}} - \Lambda_{\boldsymbol{j}}, \boldsymbol{i}, \boldsymbol{j} \in \{0, 1\}^L \right\}, \tag{7}$$

The simplest case is called Pauli encoding, when all encoding Hamiltonians are in fact Pauli matrices (e.g. encoding gates $R_Z(x) = e^{-i\frac{x}{2}\sigma_Z}$) as in Schuld et al. (2021); Caro et al. (2021). In this case, all the eigenvalues are $\lambda = \pm 1/2$, and therefore, the $\Lambda_{\boldsymbol{i}}$ are all integers (or half-integers, if L is odd) in $[-L/2, L/2]$. It follows that the set of distinct values in Ω is simply the set of integers in $[\![-L, L]\!]$. Indeed, in this case, there are many redundant frequencies, due to the fact that all Pauli eigenvalues are the same. As shown in figure 2, various eigenvalues would create more distinct frequencies in the end. In the rest of the paper, Ω will denote the set of unique frequencies, without redundancy.

Note that in section 4, we observe an unexpected phenomenon: it seems that redundant frequencies are likely to have high coefficients (for both random and trained VQCs). Unique frequencies might often have in contrast small coefficients, reducing the potential expressivity of the VQC. We see that the redundancy might therefore play an important role in the expressivity of VQCs, and leave theoretical proof for future work.

These arguments give some intuition on why one should use encoding gates from Hamiltonians with rich and various eigenvalues, by taking complex interactions over many qubits. A global Hamiltonian over n qubits, hard to implement, could potentially have 2^n distinct eigenvalues, thus enlarging Ω and avoid redundancy. Another approach from Shin et al. (2022) consists in adding scaling factors in the Pauli encoding gate to modify their eigenvalues and avoid redundancy. It results in an exponential number of integer frequencies, with respect to L, with many very high frequencies.

We can now generalize, if we now suppose that $\mathcal{X} = \mathbb{R}^d$, such that we encode a vector $x = (x_1, \ldots, x_d)$ in our quantum model, then Ω becomes the following $d-$dimensional Cartesian product $\Omega = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_d$, where each Ω_κ is defined as in Eq.7 on its own set of Hamiltonians.

In this context, note that the frequencies ω are now vectors in $\mathbb{R}^d$ and there are d different trees to build Ω (see figure 2). Note that for notation simplicity, we assumed that L gates were applied on each input's component, but it can be generalized to any number of gates per dimension.

We therefore see that the size of the spectrum $|\Omega|$ can potentially grow exponentially with the number of encoding gates and the dimension of the input data. For instance, consider a d-dimensional vector x and L Pauli-encoding gates for each dimension in such a way that there are Ld encoding gates in the VQC. The size of the spectrum Ω would scale as $O(L^d)$, which becomes quickly intractable as d increases.

As an example, the spectrum associated to a VQC with $L = 20$ encoding gates and $d = 16$ would require more than one hundred times the world's storage data capacity available in 2007 to be stored Hilbert & López (2011). We therefore wonder if it is possible to build a classical approximator $\tilde{f}(x) = \sum_{\omega \in \tilde{\Omega}} \tilde{c}_\omega e^{i\omega x}$, such that $\tilde{\Omega}$ is of tractable size and $\sup_{x \in \mathcal{X}} \left\| f(x) - \tilde{f}(x) \right\| \leq \varepsilon$.

B.3 Quantum models are shift-invariant kernel methods

As the quantum model is a real-valued function, it follows that $\omega \in \Omega$ implies $-\omega \in \Omega$ and $c_\omega = c^*_{-\omega}$. We express the Fourier series of the quantum model as a sum of trigonometric functions by defining for every $\omega \in \Omega$:

$$a_\omega := c_\omega + c_{-\omega} \in \mathbb{R} \tag{8}$$

$$b_\omega := \frac{1}{i}(c_\omega - c_{-\omega}) \in \mathbb{R} \tag{9}$$

such that

$$\begin{aligned} f(x;\theta) &= \sum_{\omega \in \Omega_+} c_\omega e^{i\omega x} + c_{-\omega} e^{-i\omega x} \\ &= \sum_{\omega \in \Omega_+} a_\omega \cos(\omega x) + b_\omega \sin(\omega x) \end{aligned} \tag{10}$$

where Ω_+ contains only half of the frequencies from Ω. Considering only Pauli matrices, if $d = 1$, we simply have $\Omega = [\![-L, L]\!]$ and $\Omega_+ = [\![0, L]\!]$. In dimension d, we have $\Omega = [\![-L, L]\!]^d$ and Ω_+ is built by keeping half of the frequencies (after removing those of opposite sign), plus the null vector. In the end, we have

$$|\Omega_+| = \frac{(2L+1)^d - 1}{2} + 1 \tag{11}$$

With a more general encoding scheme, if there is a different number of distinct positive frequencies per dimension, the formula is different but is built similarly.

In the following parts, we will focus solely on Ω_+ and conveniently drop the $+$ subscript.

Given this formulation of the quantum model, we define the feature map of the quantum model as

$$f(x;\theta) = \langle \psi(x;\theta) | O | \psi(x;\theta) \rangle = \mathbf{w}(\theta)^T \phi(x) \tag{12}$$

where $\phi(x)$ is the *feature vector*, the mapping of the initial input into a larger *feature space*, where the new distribution of the data is supposed to make the classification (or regression) solvable with only a linear model. This linear model is in fact the inner product between $\phi(x)$ and a trainable *weight vector* $\mathbf{w}$. In the case of VQCs, we can explicitly express them as:

$$\phi(x) = \frac{1}{\sqrt{|\Omega|}} \begin{bmatrix} cos(\omega^T x) \\ sin(\omega^T x) \\ \vdots \end{bmatrix}_{\omega \in \Omega}, \quad \mathbf{w}(\theta) = \begin{bmatrix} a_\omega \\ b_\omega \\ \vdots \end{bmatrix}_{\omega \in \Omega} \tag{13}$$

If the spectrum Ω is known and accessible, one can fit the quantum model by retrieving the coefficients a_ω, b_ω associated to each frequency ω. This can be done by using general linear ridge regression techniques. Interestingly, there exists a dual formulation of the linear ridge regression that depends entirely on the kernel function associated to the model Bishop & Nasrabadi (2006). The related kernel function is defined by:

$$\begin{aligned} k(x, x') &= \langle \phi(x), \phi(x') \rangle \\ &= \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \cos(\omega x)\cos(\omega x') + \sin(\omega x)\sin(\omega x') \\ &= \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \cos(\omega(x - x')) \end{aligned} \tag{14}$$

which is a shift-invariant kernel, meaning that $k(x, x') = k(x - x')$.

It is known that quantum models from VQCs are equivalent to kernel methods Schuld (2021), which means that it is equally possible to fit the quantum model by approximating the related kernel function. These kernels are high dimensional (since the frequencies in Ω can be numerous) which makes it hard to simulate classically in practice. But due to their shift-invariance, we propose to study their classical approximation using Random Fourier Features (RFF), a seminal method known to be powerful approximator of high-dimensional kernels Rahimi & Recht (2009).

C Definitions of Linear Ridge Regression (LRR) and Kernel Ridge Regression (KRR)

We present in this section the Linear Ridge Regression (LRR) and Kernel Ridge Regression (KRR) problem Bishop & Nasrabadi (2006). The problem of regression is to predict continuous label values from feature vectors. We are given a dataset $\{(x_i, y_i), i \in [\![1, M]\!] x_i \in \mathbb{R}^d, y_i \in \mathbb{R}\}$, and to each data point x an associated feature vector $\phi(x) \in \mathbb{R}^p$. The goal of LRR is to construct a parameterized model f such that $f(x) = y$. The model is parameterized by a weight vector $\mathbf{w}$ of size p such that $f(x; w) = \mathbf{w}^T \phi(x)$. Training the model consists of finding the vector $\mathbf{w}*$ that minimizes the loss function

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{M} \sum_{i=1}^{M} |\mathbf{w}^T \phi(x_i) - y_i|^2 + \lambda ||\mathbf{w}||^2 \tag{15}$$

$$= \arg\min_{\mathbf{w}} \frac{1}{M} ||\mathbf{\Phi w} - \boldsymbol{y}||^2 + \lambda ||\mathbf{w}||^2 \tag{16}$$

$$\tag{17}$$

where $\mathbf{\Phi}$ is a matrix of size $M \times p$ with each row i corresponds to $\phi(x_i)^T$ and $\boldsymbol{y}$ is the vector of all the labels y_i. The first term of the loss is the Mean Square Error (MSE) and corresponds to the difference between the prediction and the ground truth. The second term is the ridge regularization, and prevents the weights from exploding. The magnitude of the regularization is controlled by the hyperparameter $\lambda > 0$.

When $p < M$, an analytic solution to this problem is given by $\mathbf{w}^* = (\mathbf{\Phi}^T \mathbf{\Phi} + M\lambda I_p)^{-1} \mathbf{\Phi}^T \boldsymbol{y}$.

As a consequence, to make the LRR possible and have a single solution, the number of training points must be larger than the number of features in the feature space ($\phi(x)$). Otherwise, one can perform a gradient descent.

The dual formulation of this problem is given by expressing $\mathbf{w}$ as a linear combination of the data points $\mathbf{w} = \mathbf{\Phi}^T \boldsymbol{\alpha}$. The minimization on $\mathbf{w}$ become a minimization on $\boldsymbol{\alpha}$ and can be expressed as

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \frac{1}{M} ||\mathbf{\Phi \Phi}^T \boldsymbol{\alpha} - \boldsymbol{y}||^2 + \lambda \boldsymbol{\alpha}^T \mathbf{\Phi \Phi}^T \boldsymbol{\alpha} \tag{18}$$

$$\tag{19}$$

The solution of this problem is $\boldsymbol{\alpha} = (\mathbf{\Phi \Phi}^T + M\lambda I_M)^{-1} \boldsymbol{y}$.

Note that the dual solution only depends on the matrix of scalar products between feature vectors $\mathbf{\Phi}\mathbf{\Phi}^T$. One can then replace this matrix by a kernel matrix K and the obtained model is a Kernel Ridge Regression.

D Approximation results for Random Fourier Features

We give here two useful results about the bounds of the error of the RFF method. RFFs are supposed to approximate a certain kernel k by using fewer features. Intuitively, not enough features would lead to imprecise solutions. The following theorems Rahimi & Recht (2009); Sutherland & Schneider (2015) bounds the error obtained when comparing the kernel $k(x, y)$ by the RFF approximator $\phi(x)^T\phi(y)$ using D samples.

We recall that the condition on the kernel k is for it to be expressed as

$$k(\delta) = \int_{\omega \in \mathcal{X}} p(\omega) e^{-i\omega^T \delta} d\omega \tag{20}$$

where $p(\omega)$ is the distribution of the frequencies ω.

Theorem 2. *Let $\mathcal{X}$ be a compact set of $\mathbb{R}^d$, and $\epsilon > 0$.*

$$\begin{aligned} \mathbb{P}(\sup_{x,y \in \mathcal{X}} |k(x-y) - \phi(x)^T\phi(y)| \geq \epsilon) \leq \\ 66(\frac{\sigma_p |\mathcal{X}|}{\epsilon})^2 exp(-\frac{D\epsilon^2}{4(d+2)}) \end{aligned} \tag{21}$$

with $\sigma_p^2 = \mathbb{E}_p(\omega^T\omega)$, the variance of the frequencies' distribution, and $|\mathcal{X}| = max_{x,x' \in \mathcal{X}}(\|x - x'\|)$ the diameter of $\mathcal{X}$.

The following theorem Sutherland & Schneider (2015) bounds the actual prediction error when using RFF compared to the KRR estimate. The formula in the original reference contains a sign error and we correct it here.

Theorem 3. *Let $\mathcal{X}$ be a compact set of $\mathbb{R}^d$, and $\epsilon > 0$. We consider a training set $D\{(x_i, y_i)\}_{i=1}^M$. Let f be the KRR model obtained with the true kernel k and regularization $\lambda = M\lambda_0$ for $\lambda_0 > 0$, and $\tilde{f}$ the KRR model obtained with the approximate kernel and the same regularization. Then we can guarantee $|f(x) - \tilde{f}(x)| \leq \epsilon$ with probability $1 - \delta$ for a number D of samples given by:*

$$D = \Omega\left(d\left(\frac{(\lambda_0 + 1)\sigma_y}{\lambda_0^2 \epsilon}\right)^2 \left[log(\sigma_p |\mathcal{X}|) + log\frac{(\lambda_0 + 1)\sigma_y}{\lambda_0^2 \epsilon} - log\delta\right]\right) \tag{22}$$

with $\sigma_y^2 = \frac{1}{M}\sum_{i=1}^M y_i^2$ and σ_p, $|\mathcal{X}|$ being defined in theorem 2. We recall that in Eq.22 the notation Ω stands for the computational complexity "Big-Ω" notation.

E Approximation results for RFF in the context of VQCs

E.1 Distinct sampling in the Pauli encoding case

In the case of Pauli encoding only, we know that $\Omega = [\![-L, L]\!]^d$ (considered here to be the full spectrum, not Ω_+ defined in section B.3, which would have been equivalent). In one dimension, we simply have $\sigma_p = 1/L \sum_{\ell=-L,\cdots,L} \ell^2 = O(L^2)$. In dimension d, a frequency ω is given by its values on each dimension $(j_1, \cdots, j_d)$ with $j_k \in [|-L, L|]$. We similarly have

$$\sigma_p = \frac{1}{(2L+1)^d} \sum_{j_1,\cdots,j_d} j_1^2 + \cdots + j_d^2 \tag{23}$$

Note that $\sum_{j_1,\cdots,j_d} j_1^2 + \cdots + j_d^2$ is $d(2L+1)^{d-1}$ times the sum of all squares,

$$\begin{aligned} \sigma_p = \frac{d(2L+1)^{d-1}}{(2L+1)^d} \sum_{\ell=-L}^{L} \ell^2 = \frac{d}{2L+1} \frac{2L(L+1)(2L+1)}{6} \\ = O(dL^2) = O(d|\Omega|^{2/d}) \end{aligned} \tag{24}$$

The expression is then obtained by replacing the value of σ_p in theorem 3.

We note that we can generalize this results to scaled Pauli encoding, as done in Shin et al. (2022), by replacing L by a term growing as c^L where c is a constant. D would grow linearly in L and not logarithmically anymore.

E.2 GRID SAMPLING WITH A GENERAL HAMILTONIAN

We provide here a bound on the minimum of samples required to achieve a certain error between the RFF model and the complete model in the case of a general encoding in the gird sampling strategy. The proof and details for this theorem is shown in Appendix section E.2.

Theorem 4. *Let $\mathcal{X}$ be a compact set of $\mathbb{R}^d$, and $\epsilon > 0$. We consider a training set $D\{(x_i, y_i)\}_{i=1}^M$. Let f be a VQC model with any hamiltonian encoding, with a maximum individual frequency $\omega_{\max}$, trained with a regularization λ. Let $\sigma_y^2 = \frac{1}{M}\sum_{i=1}^M y_i^2$ and $|\mathcal{X}|$ the diameter of $\mathcal{X}$. Let $\tilde{f}$ be the RFF model with D samples in the grid strategy trained on the same dataset and the same regularization. Let $C = |f|_\infty|\mathcal{X}|$ and s the sampling rate defined in the grid sampling strategy. Then we can guarantee $|f(x) - \tilde{f}(x)| \leq \epsilon$ for $0 < s < \frac{1}{C}$ with probability $1-\delta$ for a number D of samples given by:*

$$D = \Omega\left(\frac{dC_1}{(\epsilon - sC)^2}\left[log(\omega_{\max}|\mathcal{X}|) + log\frac{C_2}{\epsilon - sC} - log\delta\right]\right) \tag{25}$$

with C_1 and C_2 being constants depending on σ_y, $d(X)$ and λ. We recall that in Eq.25 the notation Ω stands for the computational complexity "Big-Ω" notation.

Proof. The following theorem bounds the approximation between a function defined by its Fourier series and another function with frequencies distant by at most a constant s of the original frequencies.

Let $\mathcal{X}$ a compact set of $\mathbb{R}^d$ with diameter $|\mathcal{X}|$ and Ω a finite subset of $\mathcal{X}$. Let $f(x) = \sum_{\omega\in\Omega} a_\omega cos(\omega^T x) + b_\omega sin(\omega^T x)$. Let Ω' a subset of $\mathcal{X}$ and $s > 0$ such that $\forall\omega \in \Omega$, $\exists\omega' \in \Omega$, $|\omega - \omega'| \leq s$.

Let $\mathcal{F}_{\Omega'} = \left\{ \sum_{\omega\in\Omega'} a_\omega cos(\omega^T x) + b_\omega sin(\omega^T x), a_\omega, b\omega \in \mathbb{R}\right\}$.

Theorem 5. *It exists $f' \in \mathcal{F}_{\Omega'}$ such that*

$$\sup_{x\in\mathcal{X}} |f'(x) - f(x)| \leq sC \tag{26}$$

with $C = |\mathcal{X}||f|_\infty$.

Proof. For each $\omega \in \Omega$ let $b(\omega) \in \Omega'$ be such that $|\omega - b(\omega)| \leq s$. Such element exists by definition but is not necessarily unique. Let $f'(x) = \sum_{\omega\in\Omega} a_\omega cos(b(\omega)^T x) + b_\omega sin(b(\omega)^T x)$. The $b(\omega)$s are not necessarily different therefore there might be less frequencies in f' than in f.

$$|f(x) - f'(x)| = 2\Bigg|\sum_{\omega\in\Omega} sin(\frac{(\omega - b(\omega))^T}{2}x) \tag{27}$$

$$[b_\omega sin(\frac{(\omega + b(\omega))^T}{2}x) - a_\omega cos(\frac{(\omega + b(\omega))^T}{2}x)]\Bigg| \tag{28}$$

$$\leq 2\sum_{\omega\in\Omega} \big|\frac{(\omega - b(\omega))^T}{2}\big||x|[|b_\omega| + |a_\omega|] \tag{29}$$

$$\leq s|x|\sum_{\omega\in\Omega} |b_\omega| + |a_\omega| \tag{30}$$

$$\leq s|\mathcal{X}||f|_\infty \tag{31}$$

□

We shall here extend the proof where we sample from the grid described above. Let us note $\hat{f}_s$ the RFF model with the whole grid and $\tilde{f}$ the RFF model with D samples from the grid below. For all $x \in \mathcal{X}$ we have

$$|\tilde{f}(x) - f(x)| \leq |\tilde{f}(x) - \hat{f}_s| + |\hat{f}_s - f(x)| \tag{32}$$

$$\leq |\tilde{f}(x) - \hat{f}_s| + sC \tag{33}$$

Then

$$\mathbb{P}(|\tilde{f}(x) - f(x)| \geq \epsilon) \leq \mathbb{P}(|\tilde{f}(x) - \hat{f}_s| \geq \epsilon - sC) \tag{34}$$

for $s < \epsilon/C$.

In this case $|\Omega| = (\omega_{\max}/s)^d$ Using the expression of section E.1, we can guarantee that $|f(x) - \tilde{f}(x)| \leq \epsilon$ with probability $1 - \delta$ if

$$D = \Omega\left(d\frac{1}{(\epsilon - sC)^2}\left[log(\omega_{\max}/s) + log\frac{1}{\epsilon - sC} - log\delta\right]\right) \tag{35}$$

□

F LIMITATIONS OF RFF FOR APPROXIMATING VQCS

In section **??** have seen the theoretical power of Random Fourier Features and three different adaptations to approximate VQCs in practice. Since many parameters are to be taken into account (size and structure of Ω, number of qubits, circuit depth, number of training points, input dimension, encoding Hamiltonians, etc.), it is natural to ask ourselves in which of the three strategy is recommended given a use case, and are there any use cases for which none of them work.

As seen in section 3.4, we know the lower bound on the number of samples to draw in RFF, to reach a specific error. This bound grows linearly with the input dimension d, and logarithmically with the size of Ω (itself depending exponentially on L^d). Nonetheless, in practice, we could see very large spectrum to be harder to approximate, simply because it would require much more samples. This scaling will be judged once such VQCs will be actually implemented on large enough quantum computers (with enough qubits and/or long coherence).

Ω increases as well when the encoding Hamiltonians have distinct eigenvalues and are acting on many qubits. Therefore, quantum computers allowing for many qubits and various high locality Hamiltonians would be a plus for enlarging the spectrum.

As the Hamiltonians become larger and their eigenvalues complex, we could reach a limit where it becomes impossible to diagonalize them. In such a case, without sampling access to Ω, the *Distinct* and *Tree* sampling strategies would be unavailable. The *Grid* sampling scheme would suffice until suffering from the high dimensionality or other factors detailed above.

Finally, having a small dataset would limit the trainability of our classical RFF methods. Note that this would probably constrain the training of the VQC as well.

Overall, some limits for our classical methods can be guessed and observed already, but the main ones remain to be measured on real and larger scale quantum computers. We leave this research for future work. On another hand, one could want to understand better the relation between the available frequencies and their amplitude in practice, to find potential singularities that could help, or not, the VQCs.

Finally, we want to insist on the fact that the assumptions on VQCs are crucial on the whole construction that we propose, and that some of them could be questioned, especially concerning the encoding. For instance, when encoding vectors $x = (x_1, \cdots, x_d)$, not having encoding gates expressed as $exp(-x_i H)$ could potentially change the expression of $f(x;\theta)$ (Eq.5) and therefore could change the fact that the associated kernel would be easily expressed as a Fourier series, with shift-invariance. For instance, in Kyriienko et al. (2021), the authors use $exp(-arcsin(x_i)H)$ to encode

data, resulting in f being expressed in the Chebyshev basis instead of the Fourier one. More generally, understanding what happens with encodings of the form $exp(-g(x_i)H)$, and whether we can still use our classical approximation methods, remain an open question. Similar questions arise if we use simultaneous components encoding $exp(-x_i x_j H)$, or other alternative schemes.

G Numerical Simulations: Additional Details

G.1 Methods and definitions

As shown in figure 6, a typical random VQC instance is built from a list of general encoding Hamiltonians $\{H_1, \cdots, H_k\}$, applied to randomly selected qubits according to their locality. The number of qubits is fixed to 5 in all the experiments (Note that the number of qubits has no impact on the expressivity a priori).

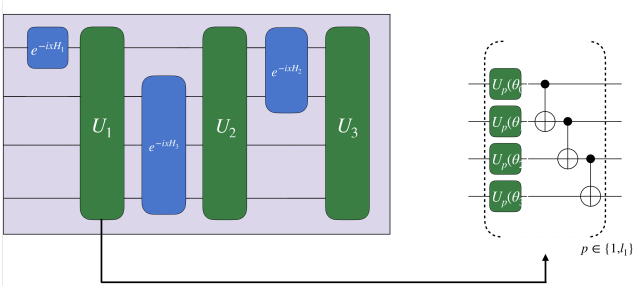


Figure 6: **Random instance of a VQC.** In this example, three encoding Hamiltonians $\{H_1, H_2, H_3\}$ are randomly assigned over four qubits, and load a 1-dimensional vector x. Following each encoding gate H_i, an ansatz with trainable parameters and a ladder of CNOTs is applied, l_i times in a row.

G.2 RFFs approximation on other types of random VQCs

We introduce this section by adding some details on the experimental framework described in section 4.1. The random VQCs follow the structure shown in figure 6. The training dataset we use is $\{X_{grid}, Y_{grid}\}$, with X_{grid} being a set of d-dimensional data points spaced uniformly on the interval $\prod_{i=1}^{d}[0, x_{max_i}]$ and Y_{grid} the evaluation of the quantum circuit on the input dataset X_{grid}.

We note that the number of data points in X_{grid} needed to efficiently learn the quantum function is $N > \prod_{i=1}^{d} \frac{x_{max_i} w_{max_i}}{\pi}$. This choice is basically related to the *Shannon criterion* for effective sampling in order to reconstruct the full function covering all of its frequencies. Moreover, it is better for the solution to be unique and hence for the least square problem introduced in Eq.2 to be well defined, we choose N to be bigger than the number of features in the regression problem (these two criteria coincide in the case of Pauli encoding).

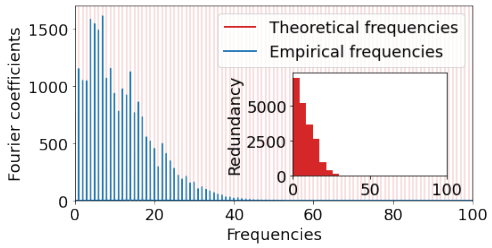
G.2.1 Pauli encoding

We first consider a quantum model with L Pauli encoding gates per feature resulting in an integer-frequency spectrum (half of $[\![| - L : L|]\!]^d$). In this case, the corresponding quantum model is a periodic function of period $T = (2\pi)^d$ and thus, we choose $x_{max} = 2\pi$ for X_{grid} construction.

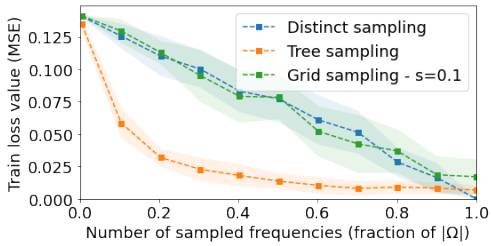
In figure 7, we implement VQCs with L=200 Pauli encoding gates, for a 1-dimensional input. We observe that our classical approximation methods are indeed able to reproduce such VQCs. On average, the RFF training error for *Distinct* and *Grid* sampling is a linear function of the number D of samples taken from Ω. On the other hand, the error using *Tree* sampling exhibits a faster decreasing trend, reaching relatively low errors with only 20% of the spectrum size. Indeed, the redundancy of Pauli encoding is extremely high, since with $L = 200$ gates, Ω can potentially have 3^{200} frequency, but only have 200 distinct ones, concentrated in the lower part.

We conjecture that the efficiency of *Tree* sampling is closely related to the redundancy in the discrete frequency distribution over Ω. In fact, as shown in figure 7, Fourier coefficients of the VQC are, on

average, correlated to the frequency redundancy in the empirical quantum spectrum. Frequencies above a certain threshold $\omega_{\text{effective}}$ are merely redundant for this particular encoding scheme, and we observe that they are cut from the quantum model empirical spectrum. The effective spectrum of the VQC is therefore smaller than what the theory predicts. Consequently, the fast decreasing trend of the *Tree* sampling stems from the fact that we sample accordingly to the redundancy, therefore requiring less frequency samples. We see that $0.2 \times |\Omega|$ samples are sufficient to sample approximately all frequencies in $[|0, \omega_{\text{effective}}|]$.



(a) Average Fourier Transform of the VQC's quantum models. The frequencies with high coefficients are the ones with high redundancy in Ω (seen in the inner red histogram). Frequencies over 100 have negligible coefficients and redundancy, and therefore are not shown.



(b) Evolution of RFF train loss as a function of the relative number of frequencies sampled. The *Tree* sampling strategy takes advantage of the high redundancy to sample less frequencies to reach a good approximation.

Figure 7: Random 1d VQCs with L=200 Pauli encoding gates, averaged over 10 different random initialization.

In figure 8, we show a similar simulations with a d-dimensional input ($d = 4$) and $L = 5$ Pauli gates per dimension. According to Eq.11, the theoretical number of distinct positive frequencies is 7321. In this case in the tree sampling procedure, we can sample both a frequency and its opposite without removing one of them. Therefore the scheme is a bit less performant than in dimension 1.

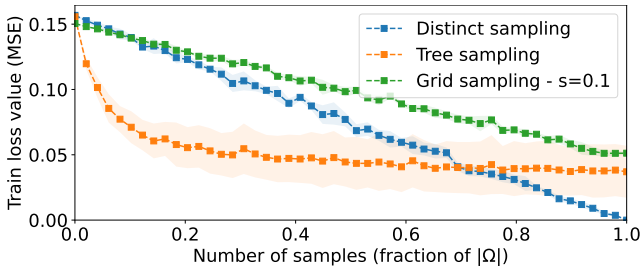


Figure 8: RFF performance for $L = 5, d = 4$, to approximate random VQCs with Pauli encoding.

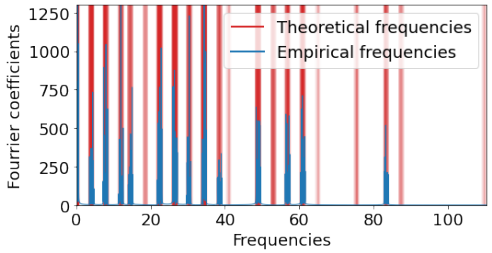
G.2.2 More complex Hamiltonian encodings

For Pauli encoding, we have seen that *Tree* sampling is highly effective for approximating the quantum model. Consequently, we designed VQCs with different spectrum distributions to study the RFF approximation performance in these cases.

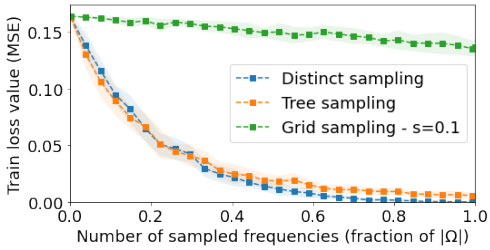
As explained in section 1, we consider encoding gates of the form $exp(-ix_iH)$ for each dimension i. One way to alter the spectrum distribution is the use of more general Hamiltonians H. To obtain exotic Hamiltonians while maintaining their physical feasibility (involving only two-bodies interactions), we use the generic expression

$$H_{XYZ} = \sum_{\langle i,j \rangle} \alpha_{ij} X_i X_j + \beta_{ij} Y_i Y_j + \gamma_{ij} Z_i Z_j + \sum_i \delta_i P_i \tag{36}$$

with the first term describing the interactions: $\langle i, j \rangle$ indicates a pair of connected particles and the second term describing a single particle's energy ($P_i = \{X_i, Y_i \text{ or } Z_i\}$).



(a) Fourier transform averaged over different random initialization of the VQC. The intensity of the vertical red lines indicates the concentration of the theoretical frequencies in Ω.



(b) RFF train loss with different sampling methods on the random VQCs. The Distinct sampling benefit from the concentration of frequencies in packets to approximate with less samples.

Figure 9: Random 1d VQCs with 4 scaled Paulis and and a 3-qubits H_{XYZ} Hamiltonian

In figure 9, we construct VQCs mixing both such Hamiltonians[1] and scaled Pauli[2] as encoding gates, on 1-dimensional inputs. In these cases, the corresponding quantum model is no longer 2π-periodic, thus we have to find empirically a good value for x_{max} (by increasing it until the performance reaches a limit).

With such complex encoding, we witness a different behavior for the *Distinct* sampling method, in comparison to the previous basic Pauli encoding scenario. Essentially, *Distinct* sampling has a faster than linear scaling, showing a clear and unexpected efficiency of RFFs in this case. We also notice that the *Tree* sampling method as a similar scaling. This observation points to the fact that, with the chosen encoding strategies, the frequencies in the spectrum Ω are concentrated in many packets or groups. This behavior is displayed with the concentrated red lines in figure **??**. Therefore, even though the frequencies in Ω have a low redundancy (545 distinct frequencies out of 2017), sampling just one of the many frequencies in a narrow packet is enough for the RFF to approximate it all. To put it differently, we can consider that there is a $\Omega_{\text{effective}}$ where each packet can be replaced by its main frequency, and the RFF manage to approximate it with fewer samples than the actual size of Ω. To conclude, many distinct frequencies is not a guarantee of high expressivity.

As for *Grid* sampling, the choice of s seemed too high for this solution to work in this case, in line with the theoretical bounds for this sampling method given in Theorem 1.

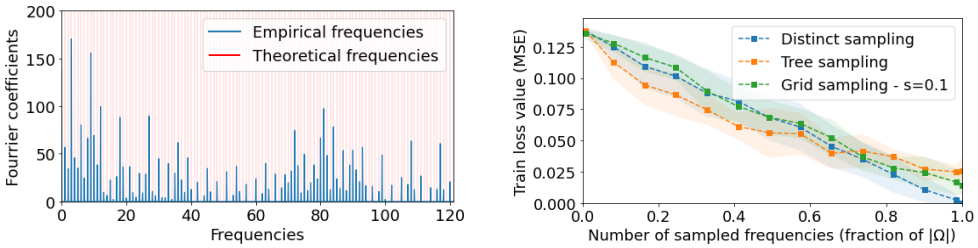
G.2.3 Exponential Pauli Encoding

In order to obtain VQCs with a large number of frequencies, but low redundancy and no concentrated packets, we exploit the exponential encoding scheme proposed in Shin et al. (2022), resulting in a non degenerate quantum spectrum with zero redundancies and thus a uniform probability distribution over integers. In this encoding strategy, encoding Pauli gates are enhanced with a scaling coefficient β_{mn} for the n^{th} Pauli rotation gate encoding the component x_m. This gives us a total of 3^{Ld} positive and negative frequencies. These frequencies can be all distinct with the particular choice of $\beta_{mn} = 3^{n-1}$, resulting in an exponentially large and uniform Ω. Note however that Ω is analytically known and contain only integer frequency, mostly very high frequencies for which the usefulness in practice remain to be studied.

We have tested our classical RFF approximation, shown in figure 10, and obtain again the confirmation that RFF can approximate such an exponential feature space with a fraction of $|\Omega|$. This fraction might however be too large in practice. We also observe as expected that all three strategies have a linear scaling, in line with the absence of redundancy and frequency packets.

[1] in figure 9, we used a 3-qubits Hamiltonian defined by: $H_{XYZ} = 7X_0X_1 + 7X_1X_0 + 0.11X_0X_2 + 0.1X_2X_0 + 8[Y_1Y_2 + Y_2Y_1 + Z_0Z_2 + Z_2Z_0]$

[2] Scaling factors are $[26.4309, 34.4309, 22.4309, 0.4309]$



(a) Fourier transform averaged over different random initializations of the exponential encoding VQC with $L = 5$.

(b) RFF approximation performance of an the exponential encoding VQC with $L = 5$.

Figure 10: Random VQCs with exponentially large Spectrum, using scaled Pauli encoding as in Shin et al. (2022).

G.3 Artificial target function

We add here the training curve obtained during the training of the VQC with $L = 200$ Pauli encoding gates, on the artificial function $s(x) = \sum_{\omega \in \{4,10,60\}} cos(\omega x) + sin(\omega x)$. Despite the potential large number of frequencies available in Ω, we have observed that the effective maximal frequency of the VQC was lower than 60, making it impossible for it to fit the high frequency of the target function.

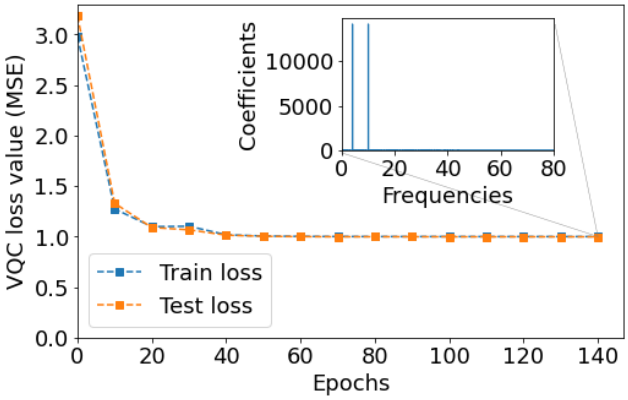


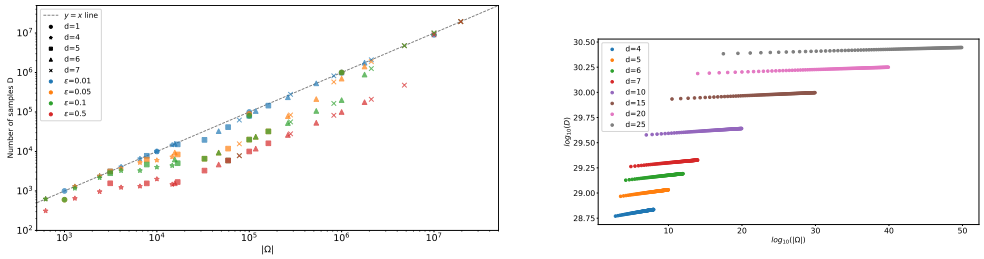
Figure 11: Predictions of the target function $s(x) = \sum_{\omega \in \{4,10,60\}} cos(\omega x) + sin(\omega x)$ with the quantum model and its corresponding RFF classical approximator using *Distinct* sampling.

G.4 Real Datasets

For the binary classification task, we used the *PyTorch* Fashion MNIST dataset with the classes coat and dress (3 and 4). We divided the 12000 input data points into train and test datasets with $N_{train} = 9600$ and $N_{test} = 2400$. For the pre-processing, we downscaled the input dataset by first rescaling the flattened input images between 0 and 1 and subtracting the mean then performing a $d = 5$ PCA transformation fitted on the train data and applied on the test data. Finally, the 5-dimensional input vectors are rescaled between $-\pi$ and π. The final VQC predictions are obtained after 60 epochs using Adam optimizer with learning rate $= 0.01$. For Tree sampling RFF training, for each fixed number of sampled frequencies p, we perform the regression on the corresponding fourier features using a *PyTorch* logistic regression model (linear layer + sigmoid layer, loss : binary cross entropy with logits, metric : accuracy) trained for 2000 epochs with early stopping using Adam optimizer with learning rate $= 0.05$. The final accuracy for the fixed number of samples p is the average score over 10 different such trained models with different random seeds for frequency sampling.

For the regression task, we used the *Scikit-learn* California housing dataset and kept only the first 5 features. We chose $N_{train} = 5000$ and $N_{test} = 1000$ and we scaled the 5-dimensional dataset between $-\pi$ and π. The final VQC predictions are obtained after 100 epochs using Adam optimizer with learning rate $= 0.01$. For Tree sampling RFF training, the same steps as in the case of the Fashion MNIST dataset are performed with a *PyTorch* regression model (linear layer, loss and metric: mean squared error).

G.5 Number of Samples and Size of Ω



(a) Experimental bounds for different values of L, d. (b) Theoretical bound for different values of L, d.

Figure 12: **Evolution of D as a function of input dimension d and of L encoding gates per dimension, and theoretical bounds.** In agreement with the theoretical bound, the number of samples D given as a fraction of $|\Omega|$ decreases with the growth of the data input dimension and the number of encoding gates.

In this section, we test the theoretical bound provided by the theorem 4. Given a spectrum $\Omega = [\![0, L]\!]^d$, a Fourier series model trained on a specific dataset, the theorem bounds the necessary number of samples for a RFF model to approximate the original model with an ϵ error. This is an approximation to the Pauli encoding VQCs where the spectrum is $\Omega = [\![-L, L]\!]^d$. For fixed values of L, d, and a spectrum $\Omega = [\![0, L]\!]^d$, we implement the following protocol to test this bound:

- Generate a dataset of 10^5 points sampled uniformly from $[0, 1]^d$ and a labels coming from a Fourier series on Ω with coefficients chosen uniformly from $[0, 1/\sqrt{|\Omega|}]$. Split in a train set and a test set with respective fractions .9 and .1
- For each value of D in $\{1, k|\Omega|/10$ for $k \in [\![1, 10]\!]\}$, sample D frequencies from Ω without replacement, and train a linear ridge regression with $\lambda = 10^{-6}$ on the train set. We performed the training with a Adam optimizer, a learning rate of .001, and between 50 and 200 epochs depending on teh size of the dataset. Compute the output on the test set.
- Compute the mean absolute error between the output of the trained model with all the frequencies and the output of all other model. Select the model with the lowest number of samples that has an error below ϵ

The results of the application of this protocol are shown figure 12. For the values of $|\Omega|$ between 10^4 and 10^6, one can see a significant reduction to the number of samples needed to approximate the whole model. For $\epsilon = .05$, one can expect to need only half of the spectrum, whereas for $\epsilon = .5$, one only need about 10% of the spectrum. The trend does not continue above $|\Omega| = 10^7$.

There are several limitations to this experiment. The main one is the limited training of the models. For the biggest values of $|\Omega|$ we limit ourselves to 50 epochs, which may be not enough to reach the optimal parameters, and thus blur the interpretation. Furthermore although the theorem is valid for every number of data points, the overparameterized regime where there are much more parameters than data points is known to exhibit unusual effects in linear regression (Hastie et al., 2022).

Given the choices of λ and ϵ, the theoretical bounds are very high for the regimes we experimentally tested, so they are not relevant. The effect that is quantified by the theory appears from $|| = 10^30$, e.g one need approximately $D = 10^{30}$ samples to approximate a 10^{40} frequency. That is still unfeasible on classical computer, so only standard benchmarks will state on the usefulness of the RFF methods.