nature portfolio

https://doi.org/10.1038/s43588-021-00084-1

Supplementary information

The power of quantum neural networks

In the format provided by the authors and unedited

Supplementary Information: The power of quantum neural networks

Amira Abbas[1,2], David Sutter[1], Christa Zoufal[1,3], Aurelien Lucchi[3], Alessio Figalli[3], and Stefan Woerner[1]

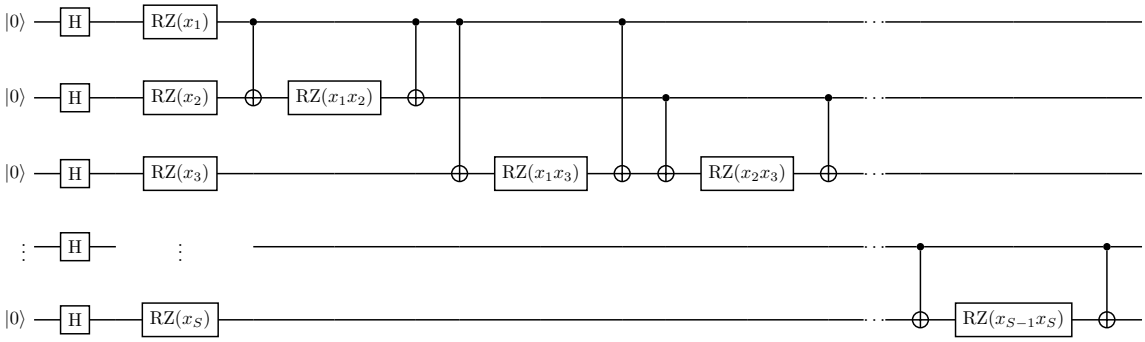
[1] *IBM Quantum, IBM Research – Zurich*
[2] *University of KwaZulu-Natal, Durban*
[3] *ETH Zurich*

1 Details of the quantum models

The quantum models considered in this study are of the form given in Figure 1 of the main manuscript. In the following, we depict the circuits used for the feature map and variational form of the quantum neural network.

1.1 The feature map

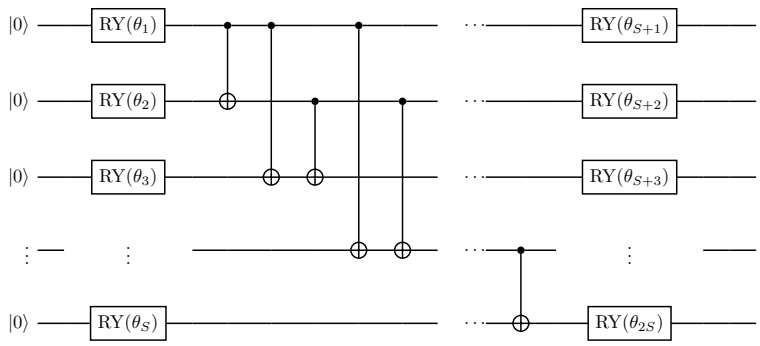
Supplementary Figure 1 contains a circuit representation of the hard feature map used in the quantum neural network. The Z-rotations are about angles which depend on the feature values of the data, x_i. A more detailed explanation is contained in the Methods section of the main manuscript.



Supplementary Figure 1: **Feature map** from [1], used in the quantum neural network. First, Hadamard gates are applied to each qubit. Then, normalized feature values of the data are encoded using RZ-gates. This is followed by CNOT-gates and higher order data encoding between every pair of qubits, and every pair of features in the data. The feature map is repeated to create a depth of 2. The easy quantum model, introduced in the results section, applies only the first sets of Hadamard and RZ-gates.

1.2 The variational form

Supplementary Figure 2 contains a circuit diagram of the variational form used in the quantum neural network, as well as the easy quantum model. There are layers of parameterized Y-rotations and all-to-all qubit entanglement. More details can be found in the Methods section of the main manuscript.



Supplementary Figure 2: **Variational circuit** used in both quantum models is plotted in this figure. The circuit contains parameterized RY-gates, followed by CNOT-gates and another set of parameterized RY-gates.

2 Properties of the effective dimension

2.1 Effective dimension converges to maximal rank of Fisher information matrix

The effective dimension converges to the maximal rank of the Fisher information matrix denoted by $\bar{r} := \max_{\theta\in\Theta} r_\theta$ in the limit $n \to \infty$. Since the Fisher information matrix is positive semidefinite, it can be unitarily diagonalized. By definition of the effective dimension, we see that, without loss of generality, $F(\theta)$ can be diagonal, i.e. $F(\theta) = \mathrm{diag}(\lambda_1(\theta), \ldots, \lambda_{r_\theta}(\theta), 0 \ldots, 0)$. Furthermore we define the normalization constant

$$\beta := d\frac{V_\Theta}{\int_\Theta \mathrm{tr}\, F(\theta)\mathrm{d}\theta}\,,$$

such that $\hat{F}(\theta) = \beta F(\theta)$. Let $\kappa_n := \frac{\gamma n}{2\pi \log n}$ and consider n to be sufficiently large such that $\kappa_n \geq 1$. By definition of the effective dimension we find

$$\begin{aligned}
d_{\gamma,n} &= 2\log\left(\frac{1}{V_\Theta}\int_\Theta \sqrt{\det(\mathrm{id}_d + \kappa_n \hat{F}(\theta))}\mathrm{d}\theta\right) / \log(\kappa_n) \\
&= 2\log\left(\frac{1}{V_\Theta}\int_\Theta \sqrt{\big(1+\kappa_n\beta\lambda_1(\theta)\big)\ldots\big(1+\kappa_n\beta\lambda_{r_\theta}(\theta)\big)}\mathrm{d}\theta\right) / \log(\kappa_n) \\
&\leq 2\log\left(\frac{1}{V_\Theta}\int_\Theta \sqrt{\kappa_n^{r_\theta}\big(1+\beta\lambda_1(\theta)\big)\ldots\big(1+\beta\lambda_{r_\theta}(\theta)\big)}\mathrm{d}\theta\right) / \log(\kappa_n) \\
&\leq 2\log\left(\frac{\kappa_n^{\bar{r}/2}}{V_\Theta}\int_\Theta \sqrt{\big(1+\beta\lambda_1(\theta)\big)\ldots\big(1+\beta\lambda_{\bar{r}}(\theta)\big)}\mathrm{d}\theta\right) / \log(\kappa_n)\,,
\end{aligned}$$

where the final step uses that the Fisher information matrix is positive definite. Taking the limit $n \to \infty$ gives

$$\lim_{n\to\infty} d_{\gamma,n} \leq \bar{r} + \lim_{n\to\infty} 2\log\left(\frac{1}{V_\Theta}\int_\Theta \sqrt{\big(1+\beta\lambda_1(\theta)\big)\ldots\big(1+\beta\lambda_{\bar{r}}(\theta)\big)}\mathrm{d}\theta\right) / \log(\kappa_n) = \bar{r}\,.$$

To see the other direction, let $\mathcal{A} := \{\theta \in \Theta : r_\theta = \bar{r}\}$ and denote its volume by $|\mathcal{A}|$. By definition of the effective dimension we obtain

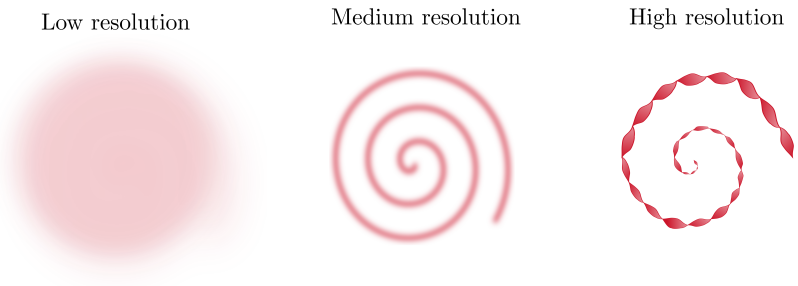
$$\begin{aligned}
\lim_{n\to\infty} d_{\gamma,n} &\geq \lim_{n\to\infty} 2\log\left(\frac{1}{V_\Theta}\int_\mathcal{A} \sqrt{\det(\mathrm{id}_d + \kappa_n \hat{F}(\theta))}\mathrm{d}\theta\right) / \log(\kappa_n) \\
&= \lim_{n\to\infty} 2\log(|\mathcal{A}|/V_\Theta) / \log(\kappa_n) + \lim_{n\to\infty} 2\log\left(\frac{1}{|\mathcal{A}|}\int_\mathcal{A} \sqrt{\det(\mathrm{id}_d + \kappa_n \hat{F}(\theta))}\mathrm{d}\theta\right) / \log(\kappa_n) \\
&\geq \lim_{n\to\infty} 2\log\left(\frac{1}{|\mathcal{A}|}\int_\mathcal{A} \sqrt{\det(\kappa_n \hat{F}(\theta))}\mathrm{d}\theta\right) / \log(\kappa_n)
\end{aligned}$$

$$= \bar{r} + \lim_{n\to\infty} 2\log\left(\frac{1}{|\mathcal{A}|}\int_{\mathcal{A}}\sqrt{\det(\hat{F}(\theta))}\mathrm{d}\theta\right)/\log(\kappa_n)$$
$$= \bar{r}\,.$$

This proves the other direction and concludes the argument.

2.2 A geometric depiction of the effective dimension

The effective dimension defined in the main document does not necessarily increase monotonically with the number of data, n. Recall that the effective dimension attempts to capture the size of a model, whilst n determines the resolution at which the model can be observed. Supplementary Figure 3 contains an intuitive example of a case where the effective dimension is not monotone in n. We can interpret a model as a geometric object. When n is small, the resolution at which we are able to see this object is very low. In this unclear, low resolution setting, the model can appear to be a 2-dimensional disk as depicted in Supplementary Figure 3. Increasing n, increases the resolution and the model can then look 1-dimensional, as seen by the spiralling line in the medium resolution regime. Going to very high resolution, and thus, very high n, reveals that the model is a 2-dimensional structure. In this example, the effective dimension will be high for small n, where the model is considered 2-dimensional, lower for slightly higher n where the model seems 1-dimensional, and high again as the number of data becomes sufficient to accurately quantify the true model size. Similar examples can be constructed in higher dimensions by taking the same object and allowing it to spiral inside the unit ball of the ambient space $\mathbb{R}^d$. Then, the effective dimension will be d for small n, it will go down to a value close to 1, and finally converge to 2 as $n \to \infty$. In all experiments conducted in this study, we examine the effective dimension over a wide range of n, to ensure it is sufficient in accurately estimating the size of a model.



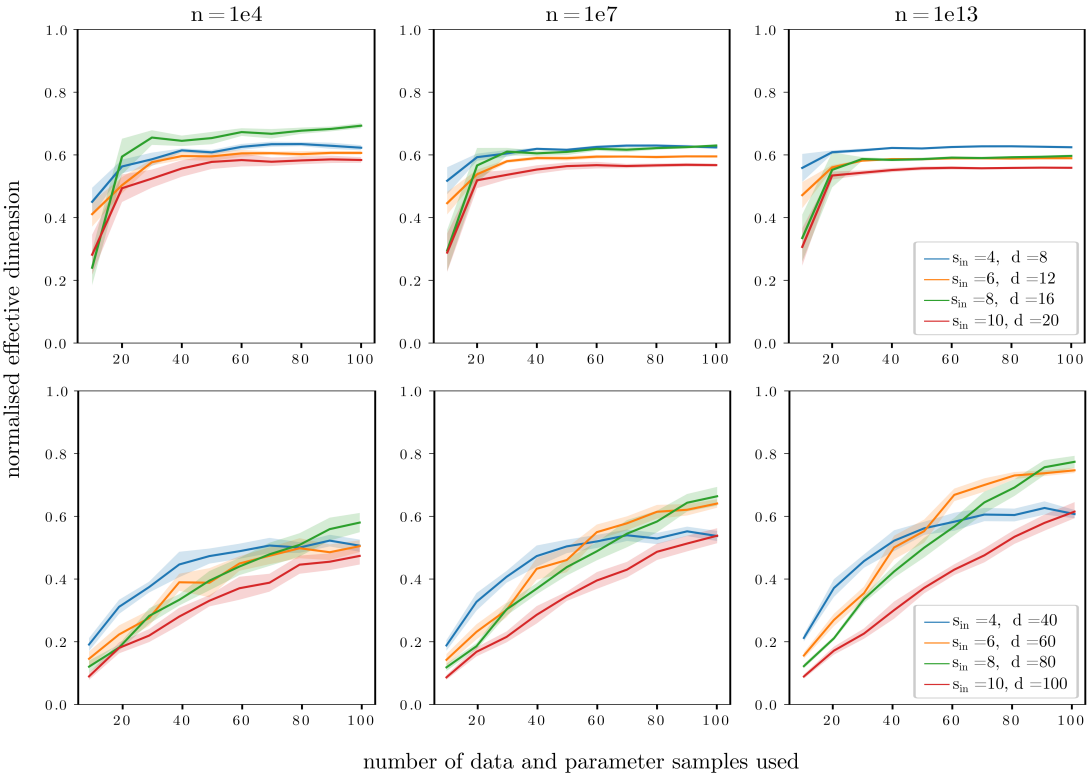
Supplementary Figure 3: **Geometric picture of a model at different resolution scales.** In the low resolution scale, the model can appear as a 2-dimensional disk and the effective dimension attempts to quantify the size of this disk. As we enhance the resolution by increasing the number of data used in the effective dimension, the medium scale reveals a 1-dimensional line, spiralling. Adding sufficient data and moving to high resolution allows the effective dimension to accurately capture the model's true size, which in this case is actually a 2-dimensional object. Thus, the effective dimension does not necessarily increase monotonically with the number of data used.

3 Numerical experiments

3.1 Sensitivity analysis for the effective dimension

We use Monte Carlo sampling to estimate the effective dimension. The capacity results will, thus, be sensitive to the number of data samples used in estimating the Fisher information matrix for a given θ, and to the number of θ samples then used to calculate the effective dimension. We plot the normalized effective dimension with n fixed, in Supplementary Figure 4 over an increasing number of data and parameter samples using the classical feedforward model. For networks with less trainable parameters, d, the results stabilize with as little as 40 data and parameter samples. When higher dimensions are considered, the standard deviation around the results increases, but

100 data and parameter samples are still reasonable given that we consider a maximum of $d = 100$. For higher d, it is likely that more samples will be needed.



Supplementary Figure 4: **Sensitivity analysis** of the normalized effective dimension to different numbers of data and parameter samples, used in calculating the empirical Fisher information matrix, and subsequently, the effective dimension.

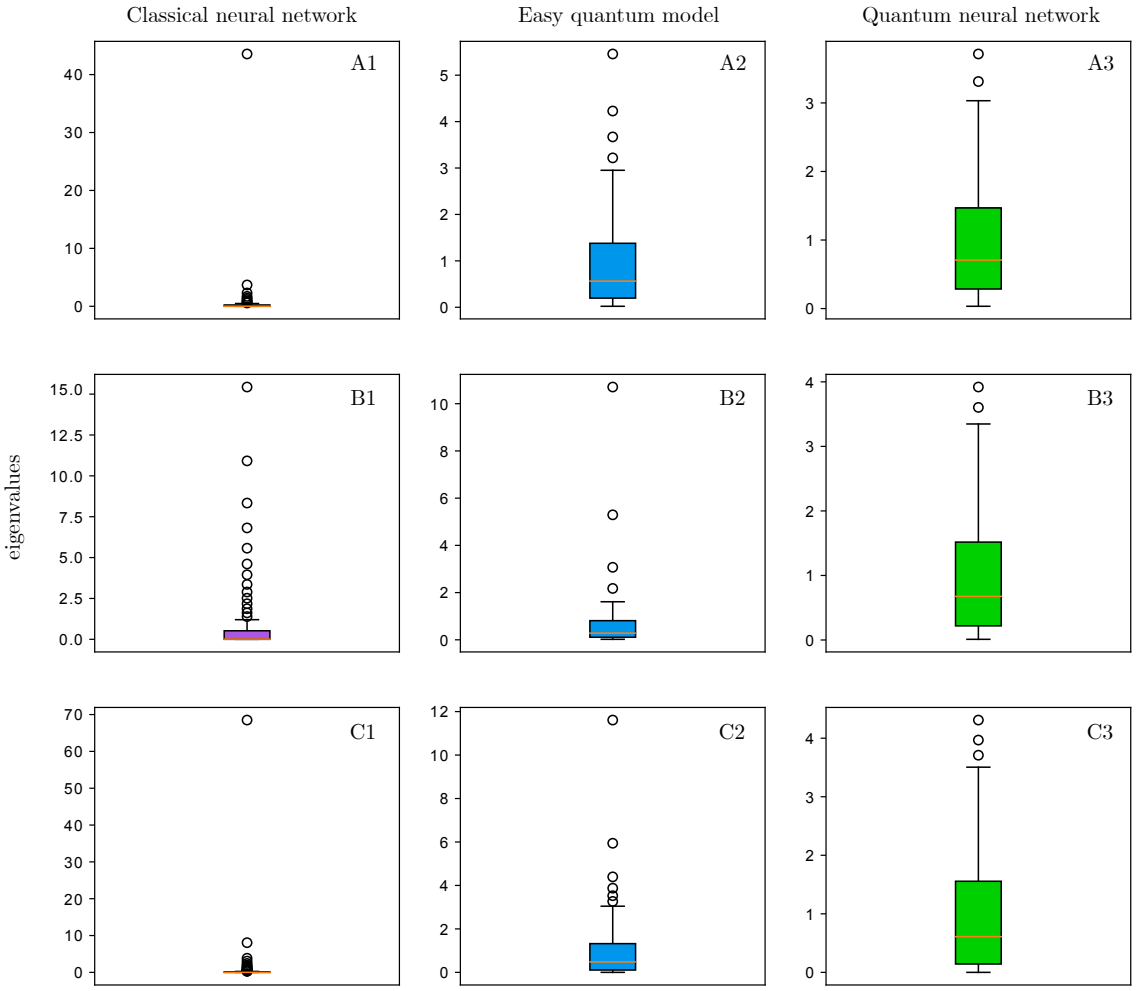
3.2 The Fisher information spectra for varying model size

Supplementary Figure 5 plots the average distribution of the Fisher information eigenvalues for all model types in box plots, over increasing input size, s_{in}, and hence, increasing number of parameters, d. The dots in the box plots represent outlier values relative to the length of the whiskers. The lower whisker is at the lowest datum above Q1 - 1.5*(Q3-Q1), and the upper whisker at the highest datum below Q3 + 1.5*(Q3-Q1), where Q1 and Q3 are the first and third quartiles. This is a standard method to compute these plots.

These average distributions are generated using 100 Fisher information matrices with parameters, θ, drawn uniformly at random on $\Theta = [-1, 1]^d$. Row A contains the distribution of eigenvalues for models with $s_{\text{in}} = 6$, row B for $s_{\text{in}} = 8$ and row C for $s_{\text{in}} = 10$. In all scenarios, the classical model has a majority of its eigenvalues near or equal to zero, with a few very large eigenvalues. The easy quantum model has a somewhat uniform spectrum for a smaller input size, but this deteriorates as the input size (also equal to the number of qubits in this particular model) increases. The quantum neural network, however, maintains a more uniform spectrum over increasing s_{in} and d, showing promise in avoiding unfavorable qualities, such as barren plateaus.

3.2.1 The Fisher information spectra with hardware noise

Similar to Supplementary Figure 5, Supplementary Figure 6 contains the average distribution of the Fisher information eigenvalues where the outputs of both quantum models are attained from a a shot-based simulation of the `ibmq_montreal 27-qubit` chip which includes a model for the underlying physical noise. Row A contains models with $s_{\text{in}} = 4$, row B with $s_{\text{in}} = 6$, row C

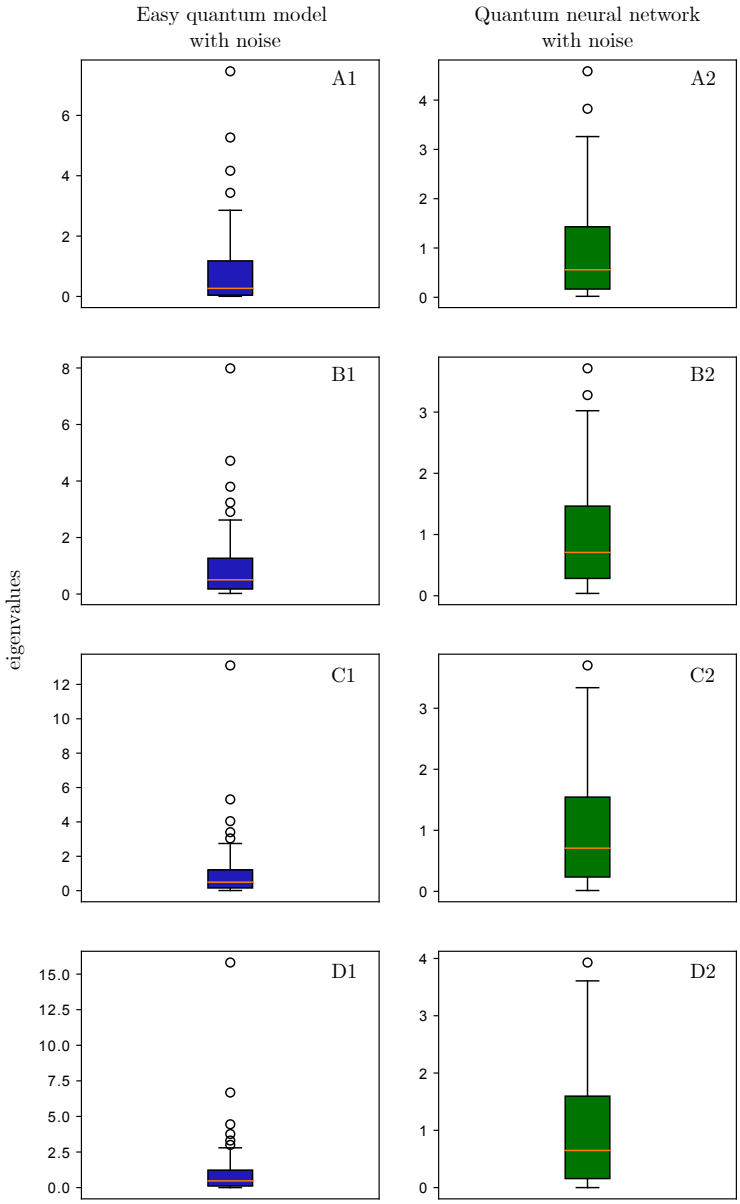


Supplementary Figure 5: **Average Fisher information spectrum** depicted as a boxplot plot for all three model types, over increasing input size, s_{in}. Row A contains models with $s_{\text{in}} = 6$ and $d = 60$, row B has $s_{\text{in}} = 8$ and $d = 80$ and row C has $s_{\text{in}} = 10$ and $d = 100$. In all cases, $s_{\text{out}} = 2$.

with $s_{\text{in}} = 8$ and row D with $s_{\text{in}} = 10$. The effect of hardware noise does not seem to change the eigenvalue distributions significantly. The easy quantum model continues to display uneven spectra as the model's size increases, whereas the quantum neural network's spectra remains stable.

3.3 Training the models using a simulator

To test the trainability of all three model types, we conduct a simple experiment using the `Iris` dataset. In each model, we use an input size of $s_{\text{in}} = 4$, output size $s_{\text{out}} = 2$ and $d = 8$ trainable parameters. We train the models for 100 training iterations, using 100 data points from the first two classes of the dataset. Standard hyperparameter choices are made, using an initial learning rate = 0.1 and the `ADAM` optimizer. Each model is trained 100 times, with initial parameters θ sampled uniformly on $\Theta = [-1, 1]^d$ each trial. We choose $\Theta = [-1, 1]^d$ as the sample space for the initial parameters, as well as for the parameter sample space in the effective dimension. Another convention is to use $[-2\pi, 2\pi]^d$ as the parameter space for initialization of the quantum model, however, we stick with $[-1, 1]^d$ to be consistent and align with classical neural network literature. We note that for the effective dimension, using either parameter space does affect the observed results. The average training loss and average Fisher-Rao norm after 100 training iterations, is captured in Supplementary Table 1. The quantum neural network notably has the highest Fisher-Rao norm and lowest training loss on average.



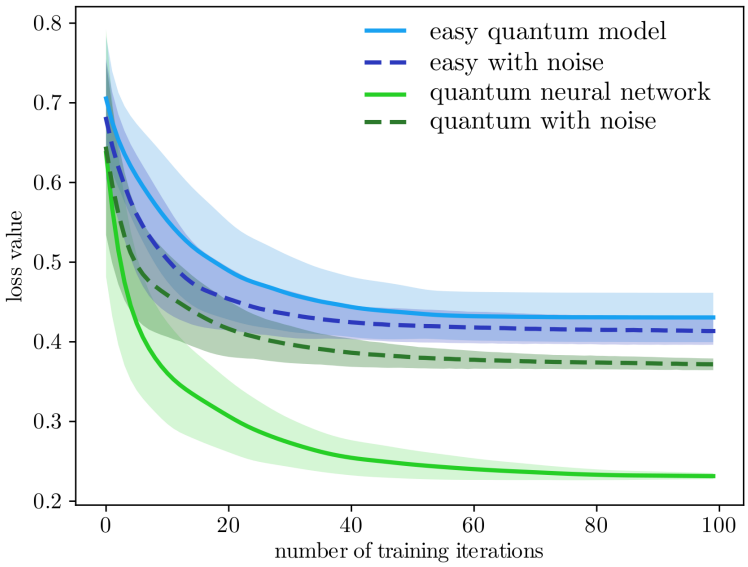
Supplementary Figure 6: **Average Fisher information spectrum** for the easy quantum model and the quantum neural network with hardware noise. Row A contains models with $s_{\text{in}} = 4$ and $d = 40$, row B with $s_{\text{in}} = 6$ and $d = 60$, row C with $s_{\text{in}} = 8$ and $d = 80$ and row D with $s_{\text{in}} = 10$ and $d = 100$.

Model	Training loss	Fisher-Rao norm
Classical neural network	37.90%	46.45
Easy quantum model	43.05%	104.89
Quantum neural network	23.14%	117.84

Supplementary Table 1: **Average training loss** and **average Fisher-Rao norm** for all three models, using 100 different trials with 100 training iterations.

3.3.1 Training with hardware noise

Training quantum models can be drastically effected by the impact of hardware noise. In order to explore this further, we investigate the training performance under noisy hardware conditions for the quantum neural network and the easy quantum model using a shot-based simulation of the `ibmq_montreal 27-qubit` device and plot the results in Supplementary Figure 7. As one would expect, the introduction of noise slows down the loss convergence for the quantum neural network. Interestingly, the easy quantum model's training performance improves when noise is added. The quantum neural network with noise still outperforms the easy quantum model with/without noise. But in both noisy situations, the models struggle to train using the `ADAM` optimizer.



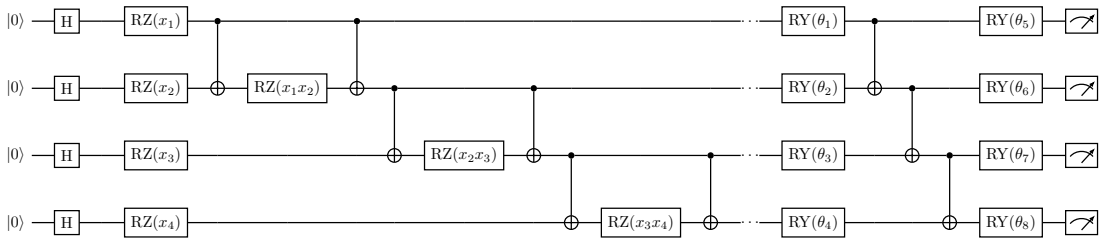
Supplementary Figure 7: **Training with hardware noise** using a cross entropy loss function and the `ADAM` optimizer. As expected, the quantum neural network's performance is negatively impacted by hardware noise. On the other hand, noise seems to improve the training performance of the easy quantum model, but overall, the quantum neural network (even with noise) performs better than the easy quantum model with and without noise.

3.4 Training the quantum neural network on real hardware

The hardware experiment is conducted on the `ibmq_montreal 27-qubit` device. We use 4 qubits with linear connectivity to train the quantum neural network on the first two classes of the `Iris` dataset and plot the circuit in Supplementary Figure 8.

4 The Fisher spectrum and the barren plateau phenomenon

The Fisher information spectrum for fully connected feedforward neural networks reveals that the parameter space is flat in most dimensions, and strongly distorted in a few others [2]. These distortions are captured by a few very large eigenvalues, whilst the flatness corresponds to eigenvalues being close to zero. This behavior has also been reported for the Hessian matrix, which coincides with the Fisher information matrix under certain conditions, e.g., under the use of certain loss functions [3–5]. These types of spectra are known to slow down a model's training and may render optimization suboptimal [6]. In the quantum realm, the negative effect of barren plateaus on training quantum neural networks has been linked to the Hessian matrix [7]. It was found that the entries of the Hessian vanish exponentially with the size of the system in models that are in



Supplementary Figure 8: **Circuit implemented on quantum hardware.** First, Hadamard gates are applied. Then the data is encoded using RZ-gates applied to each qubit whereby the Z-rotations depend on the feature values of the data. Thereafter, CNOT entangling layers with RZ-gates encoding products of feature values are applied. The data encoding gates, along with the CNOT-gates are repeated to create a depth 2 feature map. Lastly, parameterized RY-gates are applied to each qubit followed by linear entanglement and a final layer of parameterized RY-gates. The circuit has a total of 8 trainable parameters.

a barren plateau. This implies that the loss landscape becomes increasingly flat as the size of the model increases, making optimization more difficult.

The Fisher information can also be connected to barren plateaus. Assuming a log-likelihood loss function, without loss of generality, we can formulate the empirical risk over the full training set as

$$R_n(\theta) = -\frac{1}{n}\log\Big(\prod_{i=1}^{n} p(y_i|x_i;\theta)\Big) = -\frac{1}{n}\sum_{i=1}^{n}\log p(y_i|x_i;\theta),$$

where $p(y_i|x_i;\theta)$ is the conditional distribution for a data pair (x_i, y_i). From Bayes rule, note that the derivative of the empirical risk function is then equal to the derivative of the log of the joint distribution summed over all data pairs, i.e.,

$$\frac{\partial}{\partial\theta}R_n(\theta) = -\frac{\partial}{\partial\theta}\frac{1}{n}\sum_{i=1}^{n}\log p(y_i|x_i;\theta) = -\frac{\partial}{\partial\theta}\frac{1}{n}\sum_{i=1}^{n}\log p(x_i,y_i;\theta) = -\frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial\theta}\log p(x_i,y_i;\theta)\,,$$

since the prior distribution $p(\cdot)$ does not depend on θ. From [8], we know that we are in a barren plateau if, for parameters θ uniformly sampled from Θ, each element of the gradient of the loss function with respect to θ vanishes exponentially in the number of qubits, S. In mathematical terms this means

$$\left|\mathbb{E}_\theta\Big[\frac{\partial}{\partial\theta_j}R_n(\theta)\Big]\right| = \left|\mathbb{E}_\theta\Big[\frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial\theta_j}\log p(x_i,y_i;\theta)\Big]\right| = \left|\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_\theta\Big[\frac{\partial}{\partial\theta_j}\log p(x_i,y_i;\theta)\Big]\right| \leq \omega_S\,,$$

for all $j = 1, \dots d$ and for some nonnegative constant ω_S that goes to zero exponentially fast with increasing S. The barren plateau result also tells us that $\mathrm{Var}_\theta[\frac{\partial}{\partial\theta_j}R_n(\theta)] \leq \omega_S$ for models in a barren plateau. By definition of the empirical Fisher information (see main document), the entries of the Fisher matrix can be written as

$$F(\theta)_{jk} = \frac{\partial}{\partial\theta_j}R_n(\theta)\frac{\partial}{\partial\theta_k}R_n(\theta)\,,$$

for $j, k = 1, \dots, d$. Hence we can write

$$\mathbb{E}_\theta[F(\theta)_{jj}] = \mathbb{E}_\theta\left[\left(\frac{\partial}{\partial\theta_j}R_n(\theta)\right)^2\right] = \mathrm{Var}_\theta\left[\frac{\partial}{\partial\theta_j}R_n(\theta)\right] + \left(\mathbb{E}_\theta\left[\frac{\partial}{\partial\theta_j}R_n(\theta)\right]\right)^2 \leq \omega_S + \omega_S^2\,,$$

which implies $\mathrm{tr}(\mathbb{E}_\theta[F(\theta)]) \leq d(\omega_S + \omega_S^2)$. Due to the positive semidefinite nature of the Fisher information matrix and by definition of the Hilbert-Schimdt norm, all matrix entries will approach zero if a model is in a barren plateau, and natural gradient optimization techniques become unfeasible. We can conclude that a model suffering from a barren plateau will have a Fisher information

spectrum with an increasing concentration of eigenvalues approaching zero as the number of qubits in the model increase. Conversely, a model with a Fisher information spectrum that is not concentrated around zero is unlikely to experience a barren plateau.

5 Generalization properties of the effective dimension

5.1 Proof of generalisation bound

Here we provide a proof for the generalisation bound stated as a theorem in the main document. Given a positive definite matrix $A \geq 0$, and a function $g : \mathbb{R}^+ \to \mathbb{R}^+$, we define $g(A)$ as the matrix obtained by taking the image of the eigenvalues of A under the map g. In other words, $A = U^\dagger \mathrm{diag}(\mu_1, \dots, \mu_d)U$ implies $g(A) = U^\dagger \mathrm{diag}(g(\mu_1), \dots, g(\mu_d))U$. To prove the assertion of the theorem, we start with a lemma that relates the effective dimension to the covering number.

Lemma 1. *Let $\Theta = [-1, 1]^d$, and let $\mathcal{N}(\varepsilon)$ denote the number of boxes of side length ε required to cover the parameter set Θ, the length being measured with respect to the metric $\hat{F}_{ij}(\theta)$. Under the assumption of the theorem (see main document), there exists a dimensional constant $c_d < \infty$ such that for $\gamma \in (0, 1]$ and for all $n \in \mathbb{N}$, we have*

$$\mathcal{N}\left(\sqrt{\frac{2\pi \log n}{\gamma n}}\right) \leq c_d \left(\frac{\gamma n}{2\pi \log n}\right)^{d_{\gamma,n}/2} .$$

Proof. The result follows from the arguments presented in [9]. More precisely, thanks to the bound $\|\nabla_\theta \log \hat{F}(\theta)\| \leq \Lambda$, which holds by assumption, it follows that

$$\|\hat{F}(\theta) - \hat{F}(0)\| \leq c_d \Lambda \|\hat{F}(0)\| \qquad \text{and} \qquad \|\hat{F}(\theta) - \hat{F}(0)\| \leq c_d \Lambda \|\hat{F}(\theta)\| \qquad \forall\, \theta \in \Theta\,. \tag{1}$$

In the following, we set $\varepsilon := \sqrt{2\pi \log n/(\gamma n)}$. Note that, if $\mathcal{B}_\varepsilon(\bar{\theta}_k)$ is a box centered at $\bar{\theta}_k \in \Theta$ and of length ε (the length being measured with respect to the metric $\hat{F}_{ij}$), then this box contains $(1 + c_d\Lambda)^{-1/2}\mathcal{B}_\varepsilon(\bar{\theta}_k)$, where

$$\mathcal{B}_\varepsilon(\bar{\theta}_k) := \{\theta \in \Theta \,:\, \langle \hat{F}(0) \cdot (\theta - \bar{\theta}_k), \theta - \bar{\theta}_k\rangle \leq \varepsilon^2\}\,.$$

Up to a rotation, we can diagonalize the Fisher information matrix as $\hat{F}(0) = \mathrm{diag}(s_1^2, \dots, s_d^2)$. Then, we see that n the number of boxes of the form $(1 + c_d\Lambda)^{-1/2}\mathcal{B}_\varepsilon(\bar{\theta}_k)$ needed to cover Θ is given by

$$\begin{aligned}
\hat{c}_d(1 + c_d\Lambda)^{d/2} \prod_{i=1}^{d} \lceil \varepsilon^{-1} s_i \rceil &\leq \hat{c}_d(1 + c_d\Lambda)^{d/2} \sqrt{\prod_{i=1}^{d} \left(1 + \varepsilon^{-2} s_i^2\right)} \\
&= \hat{c}_d(1 + c_d\Lambda)^{d/2} \sqrt{\det\left(\mathrm{id}_d + \frac{\gamma n}{2\pi \log n} \hat{F}(0)\right)} \\
&\leq \hat{c}_d(1 + c_d\Lambda)^{d/2} \sqrt{\det\left(\mathrm{id}_d + \frac{\gamma n}{2\pi \log n} (1 + c_d\Lambda)\hat{F}(\theta)\right)} \\
&\leq \hat{c}_d(1 + c_d\Lambda)^{d} \sqrt{\det\left(\mathrm{id}_d + \frac{\gamma n}{2\pi \log n} \hat{F}(\theta)\right)},
\end{aligned}$$

where the second inequality follows from (1) and the fact that the determinant is operator monotone on the set of positive definite matrices, i.e., $0 \leq A \leq B$ implies $\det(A) \leq \det(B)$ [10, Exercise 12 in Section 82]. Here $\hat{c}_d$ depends on the orientation of $[-1, 1]^d$ with respect to the boxes $\mathcal{B}_\varepsilon(\bar{\theta}_k)$. In particular $\hat{c}_d \leq 2\sqrt{d}$ (the length of the diagonal of $[-1, 1]^d$), and if the boxes $\mathcal{B}_\varepsilon(\bar{\theta}_k)$ are aligned along the canonical axes, then $\hat{c}_d = 2$.

Since the number of boxes of size ε (with respect to the metric $\hat{F}_{ij}$) needed to cover Θ is bounded by the number of boxes of the form $(1 + c_d\Lambda)^{-1/2}\mathcal{B}_\varepsilon(\bar{\theta}_k)$, averaging the bound above with

respect to $\theta \in \Theta$ we proved that

$$\mathcal{N}(\varepsilon) \leq \hat{c}_d(1 + c_d\Lambda)^d \frac{1}{V_\Theta} \int_\Theta \sqrt{\det\left(\mathrm{id}_d + \frac{\gamma n}{2\pi \log n} \hat{F}(\theta)\right)} \mathrm{d}\theta \,,$$

which implies the inequality in the statement of Lemma 1 by recalling the definition of the effective dimension and $\varepsilon := \sqrt{2\pi \log n/(\gamma n)}$. □

Lemma 2. *Let $\varepsilon \in (0, 1)$. Under the assumption of the theorem (see main document), we have*

$$\mathbb{P}\left(\sup_{\theta\in\Theta} |R(\theta) - R_n(\theta)| \geq \varepsilon\right) \leq 2\mathcal{N}\left(\left(\frac{\varepsilon}{4M}\right)^{1/\alpha}\right) \exp\left(-\frac{n\varepsilon^2}{2B^2}\right) ,$$

where $\mathcal{N}(\varepsilon)$ denotes the number of balls of side length ε, with respect to $\hat{F}$, required to cover the parameter set Θ.

Proof. The proof is a slight generalization of a result found in [11, Chapter 3]. Let $S(\theta) := R(\theta) - R_n(\theta)$. Then

$$|S(\theta_1) - S(\theta_2)| \leq |R(\theta_1) - R(\theta_2)| + |R_n(\theta_1) - R_n(\theta_2)| \leq 2M\|\theta_1 - \theta_2\|_\infty^\alpha \,, \tag{2}$$

where the final step uses the fact that $R(\cdot)$ as well as $R_n(\cdot)$ are α-Hölder continuous with constant M for $M = M_1^\alpha M_2$. To see this recall that by definition of the risk, we find for the observed input and output distributions $r \in \mathrm{P}(\mathcal{X})$ and $q \in \mathrm{P}(\mathcal{Y})$, respectively,

$$\begin{aligned}
|R(\theta_1) - R(\theta_2)| &= \left|\mathbb{E}_{r,q}\Big[L\big(p(y|x;\theta_1)r(x), q(y)\big)\Big] - \mathbb{E}_{r,q}\Big[L\big(p(y|x;\theta_2)r(x), q(y)\big)\Big]\right| \\
&\leq \mathbb{E}_{r,q}\Big[\big|L\big(p(y|x;\theta_1)r(x), q(y)\big) - L\big(p(y|x;\theta_2)r(x), q(y)\big)\big|\Big] \\
&\leq M_2\mathbb{E}_r\big[\,\|p(y|x;\theta_1)r(x) - p(y|x;\theta_2)r(x)\|_1^\alpha\,\big] \\
&\leq M_2\,\|p(y|x;\theta_1) - p(y|x;\theta_2)\|_\infty^\alpha\,\mathbb{E}_r\big[\,\|r(x)\|_1^\alpha\,\big] \\
&= M_2\,\|p(y|x;\theta_1) - p(y|x;\theta_2)\|_\infty^\alpha \\
&\leq M_2 M_1^\alpha\,\|\theta_1 - \theta_2\|_\infty^\alpha \,,
\end{aligned}$$

where the third step uses the continuity assumption of the loss function and the fourth step follows from Hölder's inequality. The final step uses the Lipschitz continuity assumption of the model. Equivalently we see that

$$|R_n(\theta_1) - R_n(\theta_2)| \leq M_2 M_1^\alpha\,\|\theta_1 - \theta_2\|_\infty^\alpha \,.$$

Assume that Θ can be covered by k subsets $B_1, \ldots, B_k$, i.e. $\Theta = B_1 \cup \ldots \cup B_k$. Then, for any $\varepsilon > 0$,

$$\mathbb{P}\left(\sup_{\theta\in\Theta} |S(\theta)| \geq \varepsilon\right) = \mathbb{P}\left(\bigcup_{i=1}^k \sup_{\theta\in B_i} |S(\theta)| \geq \varepsilon\right) \leq \sum_{i=1}^k \mathbb{P}\left(\sup_{\theta\in B_i} |S(\theta)| \geq \varepsilon\right) , \tag{3}$$

where the inequality is due to the union bound. Finally, let $k = \mathcal{N}((\frac{\varepsilon}{4M})^{1/\alpha})$ and let $B_1, \ldots, B_k$ be balls of radius $(\frac{\varepsilon}{4M})^{1/\alpha}$ centered at $\theta_1, \ldots, \theta_k$ covering Θ. Then the following inequality holds for all $i = 1, \ldots, k$,

$$\mathbb{P}\left(\sup_{\theta\in B_i} |S(\theta)| \geq \varepsilon\right) \leq \mathbb{P}\left(|S(\theta_i)| \geq \frac{\varepsilon}{2}\right) . \tag{4}$$

To prove (4), observe that by using (2) we have for any $\theta \in B_i$,

$$|S(\theta) - S(\theta_i)| \leq 2M\|\theta - \theta_i\|_\infty^\alpha \leq \frac{\varepsilon}{2} \,.$$

The last inequality implies that, if $|S(\theta)| \geq \varepsilon$, it must be that $|S(\theta_i)| \geq \frac{\varepsilon}{2}$. This in turns implies (4).

To conclude, we apply Hoeffding's inequality, which yields

$$\mathbb{P}\left(|S(\theta_i)| \geq \frac{\varepsilon}{2}\right) = \mathbb{P}\left(|R(\theta_i) - R_n(\theta_i)| \geq \frac{\varepsilon}{2}\right) \leq 2\exp\left(\frac{-n\varepsilon^2}{2B^2}\right). \tag{5}$$

Combined with (3), we obtain

$$\begin{aligned}\mathbb{P}\left(\sup_{\theta\in\Theta}|S(\theta)| \geq \varepsilon\right) &\leq \sum_{i=1}^{k}\mathbb{P}\left(\sup_{\theta\in B_i}|S(\theta)| \geq \varepsilon\right)\\ &\leq \sum_{i=1}^{k}\mathbb{P}\left(|S(\theta_i)| \geq \frac{\varepsilon}{2}\right)\\ &\leq 2\mathcal{N}\left(\left(\frac{\varepsilon}{4M}\right)^{1/\alpha}\right)\exp\left(\frac{-n\varepsilon^2}{2B^2}\right),\end{aligned}$$

where the second step uses (4). The final step follows from (5) and by recalling that $k = \mathcal{N}((\frac{\varepsilon}{4M})^{1/\alpha})$. □

Having Lemma 1 and Lemma 2 at hand we are ready to prove the assertion of the theorem stated in the main document. Lemma 2 implies for $\varepsilon = 4M\sqrt{2\pi\log n/(\gamma n)}$

$$\begin{aligned}&\mathbb{P}\left(\sup_{\theta\in\Theta}|R(\theta) - R_n(\theta)| \geq 4M\sqrt{2\pi\log n/(\gamma n)}\right)\\ &\qquad\leq 2\,\mathcal{N}\left(\left(\frac{2\pi\log n}{\gamma n}\right)^{\frac{1}{2\alpha}}\right)\exp\left(-\frac{16M^2\pi\log n}{B^2\gamma}\right)\\ &\qquad\leq 2\,\mathcal{N}\left(\left(\frac{2\pi\log n^{1/\alpha}}{\gamma n^{1/\alpha}}\right)^{\frac{1}{2}}\right)\exp\left(-\frac{16M^2\pi\log n}{B^2\gamma}\right)\\ &\qquad\leq 4c_d\left(\frac{\gamma n^{1/\alpha}}{2\pi\log n^{1/\alpha}}\right)^{\frac{d_{\gamma,n^{1/\alpha}}}{2}}\exp\left(-\frac{16M^2\pi\log n}{B^2\gamma}\right),\end{aligned} \tag{6}$$

where the penultimate step uses

$$\left(\frac{2\pi\log n}{\gamma n}\right)^{\frac{1}{2\alpha}} \geq \left(\frac{2\pi\log n^{1/\alpha}}{\gamma n^{1/\alpha}}\right)^{\frac{1}{2}},$$

for all $\lambda \in (0,1]$ and $\alpha \in (0,1]$. The final step in (6) uses Lemma 1. □

Remark 3 (Choice of γ parameter)**.** As mentioned in the main text the parameter $\gamma \in (0,1]$ needs to be chosen sufficiently small to ensure that the right-hand side of the generalisation bound vanishes in the limit $n \to \infty$. A simple calculation reveals that this occurs if γ scales at most as $\gamma \sim 32\pi\alpha M^2/(dB^2)$. To see this, we use the fact that $d_{\gamma,n} \leq d + \tau/|\log n|$ for some constant $\tau > 0$.

Remark 4 (Improved scaling for relative entropy loss function)**.** The relative entropy is commonly used as a loss function. Note that the relative entropy is log-Lipschitz in the first argument which is better than Hölder continuous. Recall that the function $f(t) = t\log(t)$ is log-Lipschitz with constant 1, i.e., $|f(t) - f(s)| \leq |t-s|\log(|t-s|)$ for $|t-s| \leq 1/\mathrm{e}$. As a result we can improve the bound from Lemma 2 to

$$\mathbb{P}\left(\sup_{\theta\in\Theta}|R(\theta) - R_n(\theta)| \geq \varepsilon\right) \leq 2\,\mathcal{N}\left(\frac{\varepsilon/(4M)}{|\log(\varepsilon/4)|}\right)\exp\left(-\frac{n\varepsilon^2}{2B^2}\right),$$

by following the proof given above and utilizing the log-Lipschitz property of the relative entropy in its first argument and the fact that the inverse of $t|\log(t)|$ behaves like $s/|\log(s)|$ near the origin. More precisely we can choose $k = \mathcal{N}(\frac{\varepsilon/(4M)}{|\log(\varepsilon/4)|})$ in the proof above.

Remark 5 (Boundedness assumption of loss function)**.** By utilizing a stronger concentration bound than Hoeffding's inequality in (5), one may be able to relax the assumption that the loss function in the generalisation bound has to be bounded.

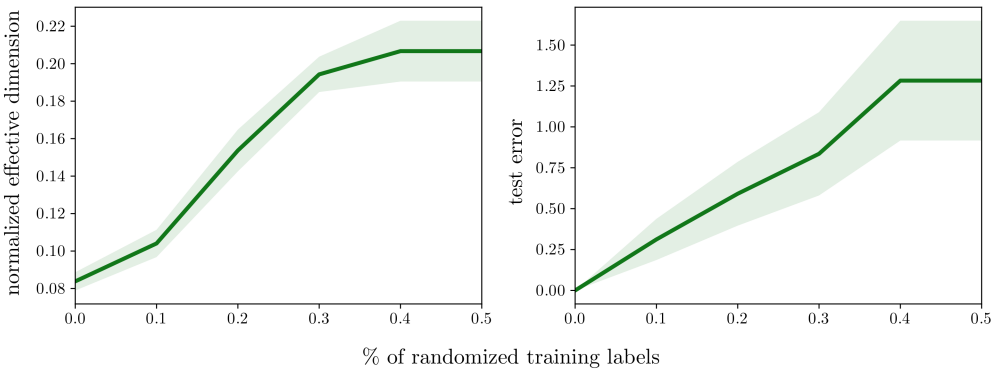
5.2 Generalization ability of the effective dimension

In order to assess the effective dimension's ability to capture generalization behavior, we conduct a numerical experiment originally introduced in [12]. The authors argue that a capacity measure is able to capture generalization behavior across local minima, if it is positively correlated with the test error on a dataset. To demonstrate this, we fix the architecture of a classical neural network model where we use a feedforward network with a single hidden layer, an input size of $s_{\text{in}} = 6$, output size $s_{\text{out}} = 2$ and number of trainable weights $d = 880$, and train it on multiple confusion sets constructed from scikit-learn's `make blobs` dataset [13]. A confusion set is a dataset where the training labels are randomized to some degree.

We first generate 1000 data points and separate them into training and test datasets using an $80\% - 20\%$ split. Then, we create confusion sets by gradually increasing the entropy in the training data through randomizing the labels in increments of 10%. Using a cross entropy loss function, we train the neural network to zero error on the original training dataset (i.e. with no randomization). Then we calculate the effective dimension using the trained parameters, as well as the test error on the test dataset. Next, we repeat this process by training the same model to zero error on each of the confusion sets and calculate the resulting effective dimension and test error.

We plot the normalized effective dimension and test error over the increasing degree of randomization in Supplementary Figure 9. As we would expect, the test error (also referred to as the generalization error) increases as the randomization in the training set increases. This is because the model is being trained on data that is increasingly noisy and eventually starts to overfit the data (i.e. starts to fit the noise). Thus, the model's ability to accurately classify the test data diminishes and it's generalization error increases.

Similarly in that regard, if a proposed capacity measure accurately captures generalization ability, we would expect to see an increasing capacity as the percentage of randomized labels in the confusion set increases. Intuitively, this is because the model requires more expressive power to fit these random labels which leads to overfitting and a higher generalization error. The effective dimension displays exactly this behavior, increasing as the entropy in the training data increases and in line with the generalization error. This experiment nicely motivates that the effective dimension can be used to capture a model's generalization ability.



Supplementary Figure 9: **Generalization behavior** of the effective dimension. On the left, we plot the normalized effective dimension for the same network trained on confusion sets with increasing randomization, averaged over 10 different training runs, with one standard deviation above and below the mean. The effective dimension correctly increases as the data becomes "more random" and is thus, able to accurately capture a model's generalization behavior. The test error, also referred to as generalization error, is plotted on the right. As the noise in the data increases, we see the generalization error increasing as the model starts to overfit (i.e. fit the noise).

5.3 Removing the rank constraint via discretization

The aim of this section is to find a suitable generalization of the results in Section 5.1 when the Fisher information matrix does not satisfy the bound $||\nabla_\theta \log \hat{F}|| \leq \Lambda$. Indeed, this is a rather strong bound as it forces $\hat{F}$ to have constant rank, so it is desirable to find a variant of Lemmas 1 and 2 that do not require such an assumption.

Our approach to this general problem is based on the idea that, in practical applications, the Fisher matrix is evaluated at finitely many points, so it makes sense to approximate a statistical model with a discretized one where the corresponding Fisher information matrix is piecewise constant.

Let $\Theta = [-1,1]^d$ and consider a statistical model $\mathcal{M}_\Theta := \{p(\cdot,\cdot;\theta) : \theta \in \Theta\}$ with a Fisher information matrix denoted by $F(\theta)$ for $\theta \in \Theta$. Given an integer $\kappa > 1$, we consider a discretized version of the statistical model. More precisely, we split Θ into κ^d disjoint cubes $\{G_i\}_{i=1}^{\kappa^d}$ of size $2/\kappa$. Then, given one of these small cubes G_i, we consider its center x_i and we split G_i into 2^d disjoint simplices, where each simplex is generated by x_i and one of the faces of ∂G_i. We denote the set of all these simplices by $\{\Theta_\ell\}_{\ell=1}^m$, where $m = 2^d\kappa^d$. Note that $\{\Theta_\ell\}_{\ell=1}^m$ is a regular triangulation of Θ.

Now, let $\mathcal{M}_\Theta^{(\kappa)} := \{p^{(\kappa)}(\cdot,\cdot;\theta) : \theta \in \Theta\}$ be a discretized version of $\mathcal{M}_\Theta$ such that $p^{(\kappa)}$ is affine on each simplex Θ_ℓ. For this, it suffices to define $p^{(\kappa)}(\cdot,\cdot;\theta) = p(\cdot,\cdot;\theta)$ whenever θ coincides with one of the vertices of Θ_ℓ for some ℓ, and then one extends $p^{(\kappa)}$ inside each simplex Θ_ℓ as an affine function. Note that, with this definition, the Fisher information matrix of the discretized model $F^{(\kappa)}(\theta)$ is constant inside each simplex Θ_ℓ. We note that, by construction, $\theta \mapsto p^{(\kappa)}(\cdot,\cdot;\theta)$ is still M_1-Lipschitz continuous. Indeed, recall that we defined $p^{(\kappa)} = p$ on the vertices of the simplices and then we extended $p^{(\kappa)}$ as an affine function inside each simplex. With this construction, the Lipschitz constant of $p^{(\kappa)}$ is bounded by the Lipschitz constant of p (since the affine extension does not increase the Lipschitz constant).

The risk function with respect to the discretized model is denoted by $R^{(\kappa)}$.

Theorem 6 (Generalization bound for effective dimension without rank constraint)**.** *Let $\Theta = [-1,1]^d$ and consider a statistical model $\mathcal{M}_\Theta := \{p(\cdot,\cdot;\theta) : \theta \in \Theta\}$ satisfying Equation (4) from the main document. For $\kappa \in \mathbb{N}$, let $\mathcal{M}_\Theta^{(\kappa)} := \{p^{(\kappa)}(\cdot,\cdot;\theta) : \theta \in \Theta\}$ be the discretized form as described above. Let $d_{\gamma,n}^{(\kappa)}$ denote the effective dimension of $\mathcal{M}_\Theta^{(\kappa)}$ as defined in the main document. Furthermore, let $L : \mathrm{P}(\mathcal{Y}) \times \mathrm{P}(\mathcal{Y}) \to [-B/2, B/2]$ for $B > 0$ be a loss function that is α-Hölder continuous with constant M_2 in the first argument w.r.t. the total variation distance for some $\alpha \in (0,1]$. Then, there exists a dimensional constant c_d such that for $\gamma \in (0,1]$ and for all $n \in \mathbb{N}$, we have*

$$\mathbb{P}\left(\sup_{\theta\in\Theta} |R^{(\kappa)}(\theta) - R_n^{(\kappa)}(\theta)| \geq 4M\sqrt{\frac{2\pi\log n}{\gamma n}}\right) \leq c_d \left(\frac{\gamma n^{1/\alpha}}{2\pi \log n^{1/\alpha}}\right)^{\frac{d_{\gamma,n}^{(\kappa)}}{2}} \exp\left(-\frac{16M^2\pi\log n}{B^2\gamma}\right), \quad (7)$$

where $M = M_1^\alpha M_2$.

To prove the statement of the theorem we need a preparatory lemma that is the discretized version of Lemma 1.

Lemma 7. *Let $\Theta = [-1,1]^d$, and let $\mathcal{N}^{(\kappa)}(\varepsilon)$ denote the number of boxes of side length ε required to cover the parameter set Θ, the length being measured with respect to the metric $\hat{F}_{ij}^{(\kappa)}(\theta)$. Under the assumption of Theorem 6, there exists a dimensional constant $c_d < \infty$ such that for $\gamma \in (0,1]$ and for all $n \in \mathbb{N}$, we have*

$$\mathcal{N}^{(\kappa)}\left(\sqrt{\frac{2\pi\log n}{\gamma n}}\right) \leq c_d \left(\frac{\gamma n}{2\pi\log n}\right)^{d_{\gamma,n}^{(\kappa)}/2}.$$

Proof. Recall that we work in the discretized model $\mathcal{M}_\Theta^{(\kappa)}$, so our metric $\hat{F}_{ij}^{(\kappa)}(\theta)$ is constant on each element Θ_ℓ of the partition. So, we fix ℓ, and we count first the number of boxes of side length ε required to cover Θ_ℓ.

Up to a rotation, we can diagonalize the Fisher information matrix $\hat{F}^{(\kappa)}|_{\Theta_\ell}$ as $\mathrm{diag}(s_1^2, \dots, s_d^2)$. Note that Θ_ℓ has Euclidean diameter bounded by $2\kappa^{-1}$ and volume κ^{-d}. Also, if $\mathcal{B}_\varepsilon(\bar{\theta}_\ell)$ is a ball centered at $\bar{\theta}_\ell \in \Theta_\ell$ and of length ε, then

$$\mathcal{B}_\varepsilon(\bar{\theta}_\ell) \cap \Theta_\ell := \left\{ \theta \in \Theta_\ell \,:\, \sum_{i=1}^d s_i^2 [(\theta - \bar{\theta}_\ell) \cdot e_i]^2 \le \varepsilon^2 \right\}.$$

Then, the number of balls of size ε needed to cover Θ_ℓ is bounded by

$$\begin{aligned}
\hat{c}_d \prod_{i=1}^d \left\lceil 2\kappa^{-1}\varepsilon^{-1} s_i \right\rceil &\le \hat{c}_d 2^d \kappa^{-d} \prod_{i=1}^d \left\lceil \varepsilon^{-1} s_i \right\rceil \\
&\le \hat{c}_d 2^d \kappa^{-d} \sqrt{\prod_{i=1}^d \left(1 + \varepsilon^{-2} s_i^2\right)} \\
&= \hat{c}_d 2^d \int_{\Theta_\ell} \sqrt{\det\left(\mathrm{id}_d + \varepsilon^{-2} \hat{F}^\kappa(\theta)\right)} \mathrm{d}\theta \,,
\end{aligned}$$

where $\hat{c}_d$ is a positive dimensional constant, and the last equality follows from the fact that the volume of Θ_ℓ is equal to κ^{-d} and that $\hat{F}^{(\kappa)}$ is constant on Θ_ℓ.

Summing this bound over $\ell = 1, \dots, m$, we conclude that (note that $V_\Theta = 2^d$)

$$\mathcal{N}^{(\kappa)}(\varepsilon) \le \hat{c}_d 2^d \sum_{\ell=1}^m \int_{\Theta_\ell} \sqrt{\det\left(\mathrm{id}_d + \varepsilon^{-2} \hat{F}^{(\kappa)}(\theta)\right)} \mathrm{d}\theta = \hat{c}_d 4^d \frac{1}{V_\Theta} \int_\Theta \sqrt{\det\left(\mathrm{id}_d + \varepsilon^{-2} \hat{F}^{(\kappa)}(\theta)\right)} \mathrm{d}\theta \,.$$

Applying this bound with $\varepsilon = \sqrt{2\pi \log n/(\gamma n)}$ and recalling the definition of the effective dimension, the result follows. □

Proof of Theorem 6. We start be noting that Lemma 2 remains valid for the discretized setting and under the assumption of Theorem 6, where Lemma 2 does not require the full rank assumption of the Fisher information matrix, i.e.,

$$\mathbb{P}\left(\sup_{\theta \in \Theta} |R^{(\kappa)}(\theta) - R_n^{(\kappa)}(\theta)| \ge \varepsilon \right) \le 2\mathcal{N}^{(\kappa)} \left(\left(\frac{\varepsilon}{4M} \right)^{1/\alpha} \right) \exp\left(-\frac{n\varepsilon^2}{4B^2} \right), \tag{8}$$

where $\mathcal{N}^{(\kappa)}(\varepsilon)$ denotes the number of balls of side length ε, with respect to $\hat{F}^{(\kappa)}$, required to cover the parameter set Θ. This can be seen by going through the proof of Lemma 2. Hence, by Lemma 7, we find for $\varepsilon = 4M\sqrt{2\pi \log n/(\gamma n)}$

$$\begin{aligned}
&\mathbb{P}\left(\sup_{\theta \in \Theta} |R^{(\kappa)}(\theta) - R_n^{(\kappa)}(\theta)| \ge 4M\sqrt{2\pi \log n/(\gamma n)} \right) \\
&\qquad \le 4c_d \left(\frac{\gamma n^{1/\alpha}}{2\pi \log n^{1/\alpha}} \right)^{\frac{d^{(\kappa)}_{\gamma, n^{1/\alpha}}}{2}} \exp\left(-\frac{16M^2 \pi \log n}{B^2 \gamma} \right).
\end{aligned} \tag{9}$$

□

Remark 8 (How to choose the discretization parameter κ)**.** In this remark we discuss conditions such that the generalization bound of Theorem 6 for the discretized model $\mathcal{M}_\Theta^{(k)}$ is a good approximation to a generalization bound of the original model $\mathcal{M}_\Theta$. Assume that the model $\mathcal{M}_\Theta$ satisfies an additional regularity assumption of the form $\|\nabla_\theta \hat{F}(\theta)\| \le \Lambda$ for some $\Lambda \ge 0$ and for all $\theta \in \Theta$, then choosing the discretization parameter $\kappa \gg \Lambda$ ensures that that $\mathcal{M}_\Theta \approx \mathcal{M}_\Theta^{(\kappa)}$ and $F(\theta) \approx F^{(\kappa)}(\theta)$. Furthermore, $\sqrt{n} \gg \kappa$ is required to ensure that the balls used to cover each simplex of the triangulation are smaller than the size of each simplex.

References

[1] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212, 2019. DOI: 10.1038/s41586-019-0980-2.

[2] R. Karakida, S. Akaho, and S.-I. Amari. Universal statistics of Fisher information in deep neural networks: Mean field approach. volume 89 of *Proceedings of Machine Learning Research*, pages 1032–1041. PMLR, 2019. Available online: http://proceedings.mlr.press/v89/karakida19a.html.

[3] J. Pennington and P. Worah. The spectrum of the Fisher information matrix of a single-hidden-layer neural network. In *Advances in Neural Information Processing Systems 31*, pages 5410–5419. Curran Associates, Inc., 2018. http://papers.nips.cc/paper/7786-the-spectrum-of-the-fisher.

[4] F. Kunstner, P. Hennig, and L. Balles. Limitations of the empirical Fisher approximation for natural gradient descent. In *Advances in Neural Information Processing Systems 32*, pages 4156–4167. 2019. http://papers.nips.cc/paper/limitations-of-fisher-approximation.

[5] Z. Liao, T. Drummond, I. Reid, and G. Carneiro. Approximate fisher information matrix to characterise the training of deep neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 2018. DOI: 10.1109/TPAMI.2018.2876413.

[6] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller. *Efficient BackProp*, pages 9–48. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. DOI: 10.1007/978-3-642-35289-8_3.

[7] M. Cerezo and P. J. Coles. Impact of barren plateaus on the Hessian and higher order derivatives, 2020. Available online: https://arxiv.org/abs/2008.07454.

[8] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9(1):1–6, 2018. DOI: 10.1038/s41467-018-07090-4.

[9] O. Berezniuk, A. Figalli, R. Ghigliazza, and K. Musaelian. A scale-dependent notion of effective dimension, 2020. Available online: https://arxiv.org/abs/2001.10872.

[10] P. Halmos. *Finite-Dimensional Vector Spaces*. Springer-Verlag New York, 1958. DOI: 10.1007/978-1-4612-6387-6.

[11] M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. MIT press, 2018. Available online: https://cs.nyu.edu/~mohri/mlbook/.

[12] Z. Jia and H. Su. Information-theoretic local minima characterization and regularization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 4773–4783. PMLR, 2020. Available online: http://proceedings.mlr.press/v119/jia20a.html.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. DOI: 10.5555/1953048.2078195.