Quantum-Enhanced Cluster Detection in Physical Images

Jason L. Pereira,[1,2,*] Leonardo Banchi,[2,3] and Stefano Pirandola[1]

[1]*Department of Computer Science, University of York, York YO10 5GH, United Kingdom*

[2]*Department of Physics and Astronomy, University of Florence, via G. Sansone 1, Sesto Fiorentino, FI I-50019, Italy*

[3]*INFN Sezione di Firenze, via G. Sansone 1, Sesto Fiorentino, FI I-50019, Italy*

Identifying clusters in data is an important task in many fields. In this paper, we consider situations in which data live in a physical world, so we first have to collect the images using sensors before clustering them. Using sensors enhanced by quantum entanglement, we can image surfaces more accurately than using purely classical strategies. However, it is not immediately obvious whether the advantage we gain is robust enough to survive data-processing steps such as clustering. It has previously been found that using quantum-enhanced sensors for imaging and pattern recognition can give an advantage for supervised learning tasks, and here we demonstrate that this advantage also holds for an unsupervised learning task, namely clustering.

I. INTRODUCTION

Pattern recognition is an important task in the field of data processing [1]. The goal is to recognize the presence of a pattern—or to identify which of a number of possible patterns are present—in data. Automated pattern recognition can be accomplished via machine learning. This could be either supervised or unsupervised learning. In supervised machine learning, we have a set of data points that fit the patterns that we are trying to recognize (labeled by which of the patterns they fit), and the algorithm must learn from this set how to classify previously unseen data points (i.e., how to find the pattern most closely matched by the unseen data). In unsupervised learning, the goal is to identify patterns in data without the use of sample data.

Clustering is one example of an unsupervised learning problem. The goal of clustering is to identify clusters of similar data points in a data set. Part of the problem is defining what constitutes a cluster in the first place [2]. Intuitively, a cluster should be a region of the parameter space in which there is a high density of data points with similar values, and any two clusters should be distinct from each other. Often, there will need to be a tradeoff between these two intuitive ideas. How this intuition is formalized

depends on the type of algorithm used, and so different types of algorithm can find different clusterings.

In some settings, the data will first need to be collected before processing [3]. For instance, suppose we have a surface that we wish to image and identify clusters on. In a microscopy setting, we may wish to find clusters of similar particles on a surface [4]. Kerr microscopy is often used to identify domains in thin magnetic films [5], which can be regarded as a clustering problem. In a biological setting, we may want to identify structures in cells [6,7]. In quantum reading [8], we have a data set encoded in a set of pixels with different values, and we may want to identify clustering or structure in the underlying data.

The improved accuracy offered by using quantum measurements has previously been shown to provide an advantage for supervised learning tasks [9,10]. We might therefore expect a similar advantage for unsupervised learning tasks, such as clustering. In this paper we compare protocols that use classical measurements for data collection with protocols that use quantum measurements, in order to establish whether such an advantage exists.

We note that this is distinct from using quantum computers to implement classical machine learning algorithms [11], and the improvement that we are looking to prove here is in accuracy, as in quantum metrology [12], rather than a speedup in computation time [13]. If we start with classical information, we cannot gain an accuracy advantage, because any operation that can be done by a quantum computer can be performed by a classical computer (albeit more slowly). The aim of gaining accuracy rather than speed is similar to that in Refs. [14,15], which consider

---

*jason.pereira@fi.infn.it

clustering of quantum states, rather than using quantum computers to speed up the clustering of classical states.

The accuracy advantage that we can gain is due to the improved imaging possible due to using quantum probes. Although one might intuitively expect that an improvement in the data used for clustering will lead to improved cluster detection, it is not obvious that the quantum advantage in imaging is significant enough to affect the results of a clustering algorithm, since it could be the case that a data-processing step could reduce the advantage due to the better data to the point of any advantage becoming negligible. In this paper, we show that a quantum advantage in imaging is robust enough to survive a clustering algorithm.

In Sec. II we outline the imaging and clustering task that we are considering. In Sec. III we give a brief overview of classical clustering algorithms and how clusterings can be validated and compared. In Sec. IV we carry out a numerical comparison of quantum and classical protocols for specific clustering tasks by simulating them using MATLAB. Section V concludes.

Some details of the numerical comparison in Sec. IV are found in the appendices. In Appendix A we calculate the receiver operating characteristic used in our numerical studies for both the classical and the quantum cases. In Appendix B we explain how we calculate the mutual information values in the numerical studies. The code used to carry out the calculations and generate the plots is available as Supplemental Material [16].

II. OVERVIEW OF THE TASK

Let us formalize our setting by considering a surface that has been divided into a grid of pixels. We believe that the surface contains some form of underlying structure (i.e., a global property of the surface), and we are interested in identifying it. In Fig. 1, this ground truth is represented by (a). The ground truth could be the positions, numbers, or shapes of clusters of particles (in the setting of microscopy) or could be how the surface is divided into domains (in the setting of microscopic films), etc. It consists of everything (and only those things) that we want to find out. In the example shown in Fig. 1, this is the number of 2×3 blocks of dark pixels.

We mathematically represent the surface as a multichannel, with each pixel being represented by a quantum channel, $\mathcal{C}_i$, where the label denotes the position of the pixel [(b) in Fig. 1]. These channels represent the interactions of the pixels with any probes sent at them. Each channel is drawn from a set of possible channels (such as a set of lossy channels with different reflectivities), and which channel we have is determined by some underlying classical parameter, ϕ_i. The identity of the channel enacted by a given pixel is determined by ϕ_i, so that identifying the channel $\mathcal{C}_i$ is equivalent to measuring ϕ_i. The data set $\{\phi_i\}$ is what we call the channel pattern and is what we are
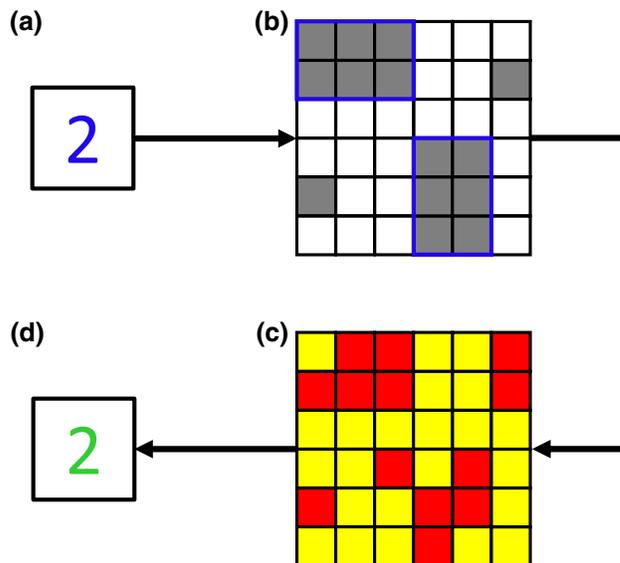
FIG. 1.   (a) The ground truth, namely the number of clusters (two, in the example shown), (b) the true channel pattern, (c) the measurement result, and (d) the result of the classical clustering (two). In the example shown, a cluster is a block of 2×3 dark pixels (which could represent collections of particles, structures, etc). The number of clusters (a) is the only piece of information that we are interested in, rather than the position of the clusters on the surface (if we were interested in the positions, that information would also be part of the ground truth). (b) is how this ground truth corresponds to the physical multichannel that we probe. Here, each pixel is one of two types of channel: gray, representing the presence of a particle, or white, representing the absence of particles. The two clusters are outlined in blue, but note that there are two gray pixels that are not part of either of the clusters. These could represent, for instance, particles that do not form part of one of the structures that we are looking for. If we had perfect knowledge of the state of (b), we would then carry out classical clustering algorithms in order to find the two outlined clusters and recover the value of (a). Instead, we carry out measurements on the multichannel in order to estimate (b), with the result being (c), where red squares represent pixels we believe to be gray in (b) and yellow squares represent pixels we believe to be white. Our measurement process cannot perfectly reproduce (b), and so some squares that are white in (b) are red in (c) and some squares that are gray in (b) are yellow in (c). These are the two types of misdetection that can occur in this scenario. Finally, we carry out some classical clustering algorithm on (c) in order to estimate the number of clusters, and hence (a). This estimate is (d). Note that in this figure, the algorithm has only found two clusters, and hence has estimated (a) correctly, despite the extra red pixels. A different, potentially less appropriate, clustering algorithm might mistakenly decide that the two red pixels in the top-right corner of (c) are part of a cluster, and so overestimate (a).

trying to find out by imaging the surface. For instance, the presence or absence of a particle could each correspond to one of two possible quantum channels. In this binary case of each pixel being one of only two possible channels,

each ϕ_i will be drawn from the set $\{0, 1\}$. If we have more possible channels, these could each correspond to different types of particles being present (and we would have more possible values of ϕ_i). Alternatively, the channel could be continuously parameterized, corresponding to continuous data (e.g., the magnetization vector of each pixel in a thin magnetic film).

This data set, $\{\phi_i\}$, will stochastically depend on the ground truth, but may not be entirely determined by it. There could, for instance, be unclustered particles randomly distributed over the surface. In Fig. 1, (a) gives the number of clusters, but not their positions or orientations. In other words, the mapping from (a) to (b) may not be deterministic. Any information present in (b) that is not present in (a) is information that a classical clustering algorithm would aim to discard.

Suppose our goal is to carry out pattern recognition on the data set, $\{\phi_i\}$, which corresponds to the channel pattern. A protocol to do so would consist of two stages: carrying out measurements to estimate the parameters ϕ_i and then clustering the resulting data set.

We can image the surface by sending probes to interact with each of the pixels. These probes can be regarded as quantum states, and could consist of photons or of some other type of particles, such as electrons (e.g., in electron microscopy). Some measurement is then carried out on the return states, in order to determine the quantum channels corresponding to each pixel. In other words, we carry out quantum channel tomography (or metrology if the parameterization of the possible channels is continuous) in order to reconstruct the true channel pattern [(b) in Fig. 1] from the measurement result [(c) in Fig. 1]. Note that if we are able to use unlimited energy to probe the pixels, we will be able to perfectly reconstruct the channel pattern, so there will be no difference between (b) and (c). However, if we limit the energy of the probes sent through each pixel, we will (in general) have an imperfect reconstruction of (b). This constraint could be due to our sample being sensitive, so that an energetic measurement would be destructive. We can then carry out classical clustering algorithms on the measurement result to get (d), an estimate of the ground truth (a).

We can categorize possible protocols based on the type of probe used: specifically whether or not the probes have a positive P-representation. We call states with a positive P-representation (such as coherent states) classical, and all other states, such as two-mode squeezed vacuums (TMSVs) and number states, quantum. Other quantum states, with multipartite entanglement, were considered for imaging purposes in Ref. [17].

We will therefore call protocols that probe the pixels with classical states and then carry out clustering on the measurement results "classical protocols" and will call protocols that probe the pixels with quantum states and then carry out clustering on the results "quantum-classical

protocols." This latter choice of name is to allow for a potential third type of fully quantum protocol (not considered in this paper) that probes the surface with some collective quantum state—potentially with entanglement between pixels—and then carries out a collective measurement on the return state in order to extract a global property (the ground truth), rather than probing the surface pixelwise and carrying out classical clustering on the result. However, the quantum-classical class of protocols is more relevant from a near-term perspective, since TMSVs, which could be used as signal-idler pairs to individually probe the pixels, can be generated using current technology. This may not be the case for more general states used in fully quantum protocols.

In this paper, we will consider classical and quantum-classical protocols that send a single probe through each pixel, with the same energy constraint for each pixel, and then carry out a classical clustering algorithm on the results. Although the optimal quantum-classical protocol cannot be worse than the optimal classical protocol at reconstructing the channel pattern [recovering (b) from (c)], due to the fact that classical protocols are a special case of quantum-classical protocols, there is no guarantee that any advantage gained at this stage will be retained through the data processing. It is therefore worth investigating whether the improved imaging capabilities afforded by quantum states can lead to a non-negligible advantage in clustering accuracy.

III. CLASSICAL CLUSTERING

A. Types of clustering algorithms

Two of the main, basic types of classical clustering algorithms are centroid-based algorithms (such as k-means [18]) and density-based algorithms (such as DBSCAN [19]). With centroid-based algorithms, the goal is to minimize the sum of the distances (via some metric) of the points to the nearest centroid, for a fixed number of centroids. For density-based algorithms, we grow clusters as areas with a high density of points (and some density cutoff). Other possible types of algorithms include distribution-based methods, in which we fit our points to some distribution (e.g., a sum of Gaussians) where parameters of the distributions are unknown. Each method may correspond to a different scenario we may be interested in, and the "best" type depends on the specific setting.

In k-means, we must choose the number of clusters in advance (as k) and then we find the k centroids (points) such that the total distance from each point to the nearest centroid is minimized. Distance may be Euclidean, squared Euclidean, Bures, etc. By design, the clusters will always be roughly spherical (in terms of the distance metric). k-means assigns every point to a cluster. One common variant of k-means is k-medoids [20], which is similar to k-means, but limits the possible centroids to the data

points themselves (rather than allowing them to be located anywhere in the coordinate space).

In DBSCAN, we calculate the density of points in a region around each point (the size of the region, ϵ, is a parameter for the algorithm). If this is above a cutoff, the point is a core point of a cluster. If not, it is either a noncore point or not in a cluster. Such algorithms find all clusters rather than specifying the number in advance. DBSCAN may decide that some points are not part of a cluster at all.

While both k-means and DBSCAN assign each data point to at most a single cluster, variants exist that carry out "fuzzy" clustering, in which data points can be assigned to multiple clusters, with different degrees of affiliation.

Which type of algorithm is appropriate depends on our setting and what we want to discover. This relates to the previously mentioned question of what constitutes a cluster. For supervised learning, we have a correct answer and can calculate the error. For unsupervised learning, we only have intuitive ideas of what a good clustering "should" look like.

Consider Fig. 2, which compares k-means clustering and DBSCAN for two different data sets. The top data set was generated by picking three pairs of coordinates and then choosing coordinates randomly from normal distributions centered on those three points, while the bottom data set was generated by picking a pair of coordinates and picking points from both a normal distribution centered on it and

from a ring around it. k-means clustering with three clusters is able to identify three distinct, roughly circular sets of points in the top data set, while DBSCAN groups two of these sets together, due to their having some overlap. In the bottom data set, DBSCAN is able to separate out the ring and the circle in the middle and identify both as being clusters while k-means clustering is inherently unable to identify one cluster inside another, since both clusters have the same center, and so simply splits the ring in two. In each case, one of the clustering techniques correctly identifies the ground truth—the underlying distribution used to generate the data set—while the other does not. The appropriate clustering algorithm depends on the data set itself and what information we want to extract from it.

B. Cluster validation

How can we assess how well clustering has been done? There are several metrics that we can use, each of which weights the properties that we want for our clusters differently. Methods for assessing a clustering can be broadly divided into two categories: internal and external.

An internal metric compares a clustering to our intuitive idea of what a cluster should be. Clusters should be compact (small intracluster distances) and well separated (large intercluster distances). One possible metric is the sum of the distances between points and the center of their cluster. By definition, this is minimized by k-means (so long as it finds a global, rather than a local, minimum), but there is no guarantee that clusters will be well separated. It also performs poorly (compared to our intuitive idea of what clusters look like) when the clusters in the data are nonspherical. Other options include the Dunn index, which takes into account both the intra- and the intercluster distances, and the silhouette coefficient.

An external assessment assumes the existence of a ground truth. In other words, there exists some objectively correct clustering that we need to find. In a physical imaging context, this will often be the case: we carry out imaging in order to determine some real property of interest about the surface we are imaging. To assess our closeness to the true clustering, however, we would need to know the ground truth beforehand. If our goal is to find the ground truth, this can present a problem, since we would not be able to use it to assess the validity of our clustering. However, external validation can be of use in comparing different imaging and clustering protocols, to see which is better at extracting the information we want about the underlying structure.

FIG. 2. k-means clustering (left) and DBSCAN (right) on two different data sets. The top data set was generated as the sum of three normal distributions with different mean values, while the bottom data set was generated as a normal distribution with a surrounding ring. k-means clustering (with three clusters) is better than DBSCAN at identifying the three (roughly circular) clusters corresponding to each of the normal distributions in the top image, while DBSCAN is better (than k-means clustering with two clusters) at identifying the structure of a ring with a circle inside in the bottom image. Note that only the DBSCAN images have unclustered points (in black).

C. Comparing clusterings

If we want to assess how different measurement techniques affect the clustering found, how can we go about doing this? If we were to use an internal validation method, we might find that a less good measurement method (one

that less faithfully reconstructs $\{\phi_i\}$) results in better clustering performance than a better measurement method. However, this would simply be telling us that the measurement data produced by the worse measurement is more clustered than the data produced by the better measurement, and would not give any information about which method is closer to the "true" clustering.

Instead, we can use an external validation method. Specifically, we can compare the clustering resulting from a given protocol to the ground truth clustering. We can do this in terms of the mutual information between the ground truth and the estimation of the ground truth that we get from our measurement result. The mutual information is a classical quantity that tells us how much information the measurement result [(c) in Fig. 1] holds about the ground truth that we are trying to discover [(a) in Fig. 1]. The mutual information is routinely used for machine learning purposes, forms the basis of the information bottleneck method [21], and was employed to study the generalization and classification errors in supervised quantum classifiers [22]. For a given protocol, we can calculate the reduction in uncertainty about the position of the clusters due to the use of the protocol—quantified by the mutual information—and compare this to the reduction in uncertainty achieved by a different protocol.

Note that the best possible measurement will not necessarily be one that maximizes the mutual information between the measurement result and the channel pattern [(b) in Fig. 1], since it could be the case that a protocol that discards some of the information that is not relevant to the ground truth (e.g., has a low probability of detecting unclustered points or outliers) is better at identifying the ground truth than one that more faithfully reproduces the channel pattern.

IV. CLASSICAL VERSUS QUANTUM-CLASSICAL PATTERN DETECTION

In both discrete variable and continuous variable settings, we can gain a quantum advantage for channel discrimination and metrology for a large variety of sets of possible channels. Since quantum probes can give a quantum advantage in discriminating between possible channels, we expect them to be able to better determine the ground truth.

The extent of the quantum advantage depends on the set of possible channels that we are considering. For instance, if we are discriminating between a Pauli Z channel and the identity channel, an entangled state can perfectly discriminate between the two options, whereas a classical state cannot distinguish between them at all. This means that it is not possible to make general statements about the advantage offered by using quantum states, beyond the fact that they offer an improvement.

Instead, we will consider two specific examples of problems: one involving centroid-based clustering and the other involving density-based clustering. We will compare classical protocols and quantum-classical protocols, restricting ourselves to one-shot, nonadaptive protocols.

The types of problem we will consider will involve imaging surfaces divided into grids of pixels and clustering the measurement results. In the scenarios we will consider, imaging the pixels is a quantum reading type task—each of the pixels has a binary value (0 or 1) and each possible value corresponds to a different lossy channel ($\mathcal{C}_0$ or $\mathcal{C}_1$) [8]. In each case, we will consider the same pair of possible channels, but with different types of underlying channel pattern and consequently different clustering strategies after carrying out the measurements.

Since we have only two possible channels, we have two types of error. These are the Type I error—the probability of detecting a particle when no particle is present—and the Type II error—the probability of not detecting a particle that is present. The plot of the achievable Type I error for a given Type II error, for a given detector, is called the receiver operating characteristic (ROC).

For certain pairs of channels and energy constraints, we can prove a quantum advantage in discriminating between them, meaning that we can achieve a lower Type II error for a fixed Type I error. Our aim is to show, via numerical simulations (in MATLAB), that the lower Type II errors obtained by using quantum imaging protocols can lead to an increase in clustering accuracy, and hence that using quantum, rather than classical, imaging protocols can allow us to gain more information about the ground truth.

In order to numerically compare quantum and classical protocols, we must choose some concrete parameter values. We let both channels be pure loss channels, setting the transmissivity of $\mathcal{C}_0$ to 0.95 and the transmissivity of $\mathcal{C}_1$ to 0.4. We constrain the per-pixel probe energy to a mean photon number of 8. This gives us the ROC in Fig. 3. See Appendix A for details of how the ROC was found for each case.

For each type of protocol, we choose pairs of Type I and II errors from the ROC curve in Fig. 3, for Type I error values between 0 and 0.05, and carry out simulations using these pairs of errors. The reason for using this range is that for large values of the Type I error, the results become almost entirely random, since the number of false positives becomes comparable to the number of true positives. This means that, for a classical protocol, the Type II error varies between approximately 0.1899 (for a Type I error of 0.05) and 0.3918 (for a Type I error of 0), while for a quantum protocol, the Type II error varies between approximately 0.1107 (for a Type I error of 0.05) and 0.1424 (for a Type I error of 0).

Note that the classical Type II errors are lower bounds (the best possible Type II errors for fixed Type I errors), while the quantum Type II errors are upper bounds on the best possible Type II errors for fixed Type I errors. Consequently, any calculated mutual information values
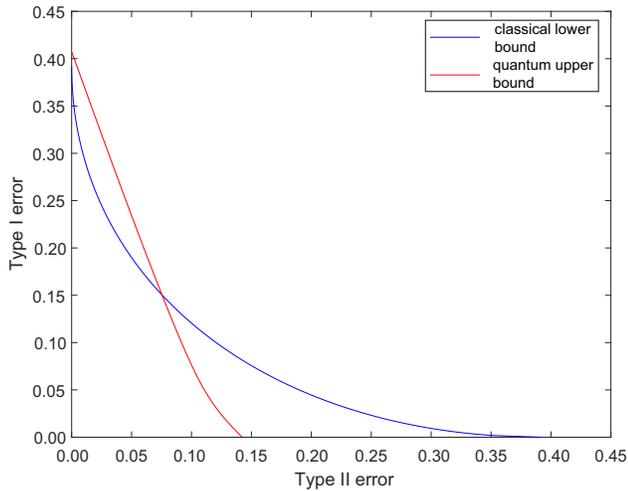
FIG. 3. Receiver operating characteristic for classical and quantum protocols, each with an average photon number of 8, discriminating between two pure loss channels, one with a transmissivity of 0.95 and one with a transmissivity of 0.4. The blue line is the lower bound (LB) for classical protocols (protocols using probes with positive P-representations), while the red line is an achievable curve for quantum protocols [and therefore constitutes an upper bound (UB) on the optimal quantum protocol]. For low Type I errors, we can prove a quantum advantage in terms of measurement errors.

(between the ground truth and our estimate of it) that use the classical error values will be upper bounds for classical protocols, while mutual information values that use the quantum error values will be lower bounds on the achievable values for optimal quantum protocols.

We will refer to the variable encoding our ground truth as A, the variable encoding the channel pattern as B, the measurement result as C, and the resulting estimate of the ground truth as D (see also Fig. 1). A is some global property of the surface (i.e., not a property of any particular single pixel) that we want to find out. B is the actual pattern of pixels on the surface (a matrix of 0s and 1s), which is correlated to A. Each protocol images each pixel, using either classical or quantum states, to get a different matrix of 0s and 1s, and then carries out clustering on the result. The matrix of 0s and 1s constitutes the variable C (our estimation of B), while the result of the clustering algorithm gives us the variable D (our estimation of A). We can quantify how good a protocol is at obtaining the ground truth by finding the mutual information between variables A and D. This can be expressed as

$$I(A:D) = H(D) - H(D|A). \qquad (1)$$

The full simulation process for a given pair of Type I and II errors is as follows. First, randomly choose a value of the ground truth variable, A. Next, generate a true channel pattern (matrix of 0s and 1s) based on the value of
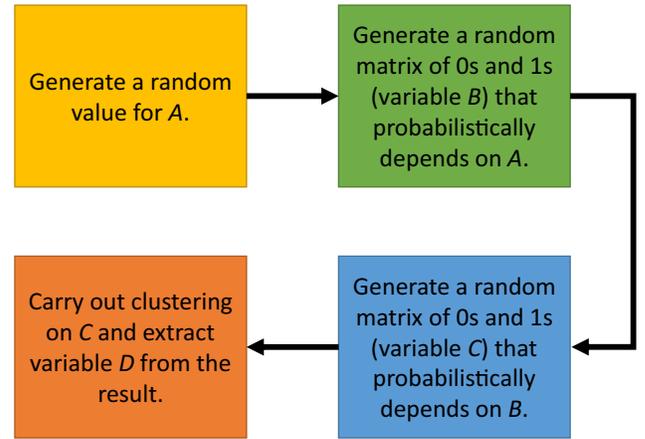
FIG. 4. Flow chart outlining the numerical simulation process used to compare classical and quantum-classical protocols.

A. This channel pattern is the variable B, and the probabilistic mapping between A and B depends on the specific scenario. Then probabilistically generate a measurement result, C, based on B. This is again a matrix of 0s and 1s. If an entry in matrix B is 0, the corresponding entry in C is 0 with probability $1 - \xi_1$ and is 1 with probability ξ_1, where ξ_1 is the Type I error. If an entry in matrix B is 1, the corresponding entry in C is 0 with probability ξ_2 and is 1 with probability $1 - \xi_2$, where ξ_2 is the Type II error. A clustering algorithm is then carried out on variable C (the choice of algorithm and the settings depend on the scenario) to generate variable D. Finally, variables A and D are recorded.

These steps are repeated a large number of times so that the mutual information between A and D can be estimated. The process for generating each sample is outlined in Fig. 4. Note that the process is a Markov chain: the estimation of the ground truth, D, depends only on the measurement result, C, which depends only on the channel pattern, B, which depends only on the ground truth, A.

A. Classical versus quantum-classical k-medoids clustering

In our first scenario, we have a surface on which there are a number of noninteracting particles. We divide this surface into a $d \times d$ grid of pixels and image each pixel to determine the presence of absence of a particle in that pixel. We assume that a pixel can be occupied by at most one particle, so that we have only two choices of possible channels, which we can label as $\mathcal{C}_0$ (no particle) and $\mathcal{C}_1$ (particle present).

Suppose we know that the surface contains m attractors, to which the particles are attracted, so that the probability function for finding a particle near to an attractor is Gaussian. Specifically, suppose the m attractors are located at coordinates $\{x_m, y_m\}$ and the probability of a pixel with its
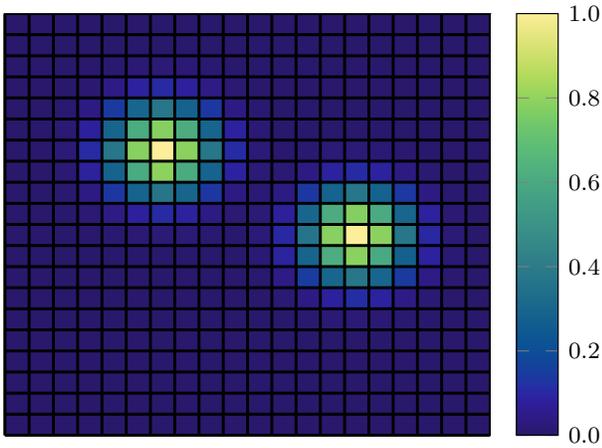
FIG. 5. Proportion of samples in which each pixel contains a particle, for two attractors, for the scenario in Sec. IV A. The probability of each pixel containing a particle for any given sample is given by Eq. (2) (assessed at the center of each pixel). As can be seen, the pixels on which the attractors are centered always contain particles, while those further out contain particles less frequently, and those more than 4 pixels from an attractor have a probability of containing a particle that is close to 0. The aim of imaging the surface is to locate the two attractors.

center at position (x, y) containing a particle is given by

$$f(x,y) = \phi \sum_{i=1}^{m} e^{-[(x-x_m)^2 + (y-y_m)^2]/2\sigma^2}, \qquad (2)$$

where ϕ is some positive constant and σ^2 is some variance. For our simulations, we will set $m = 2$, so that there are always exactly two attractors. The probability of finding a particle on each pixel of the surface, for two attractors and a specific choice of attractor locations, is shown in Fig. 5. The probability is high close to the attractors and decays with distance from them. More than three standard deviations from either attractor, the probability of a pixel containing a particle is close to 0. If our task is to locate the attractors, k-medoids clustering (with m clusters) is a natural choice.

We emphasize that the task is to find the locations of the attractors, via their effect on the distribution of particles, rather than the locations of the particles themselves. The grid may contain any number of particles, but will always contain exactly two attractors. Furthermore, we are not able to directly detect the attractors, we can only detect the presence or absence of particles, the probability of which (for each pixel) is affected by the positions of the attractors. The purpose of this is to relate the task to a physical scenario in which clustering of the results is necessary. For instance, the attractors could be electrical charges on the surface to which the particles are attracted.

The simulation is done as per Fig. 4. The coordinates of the attractors on the grid of pixels for each sample constitute the variable A for that sample, and the positions of all of the particles constitute the variable B. The coordinates of the centroids found by the k-medoids algorithm constitute variable D, our estimate of A.

We choose random pairs of well-separated coordinates for each sample of A and generate the corresponding probabilities for each pixel to have a value of 1 in the channel pattern (variable B). Each value of B is a matrix of 0s and 1s, with a probability of each that is different for every pixel and depends on A [given by Eq. (2)]. If we were to draw a large number of sample values of B for a fixed value of A, we would get a distribution similar to Fig. 5.

As can be seen, some pixels have a value of 0 in almost all of the samples (corresponding to the absence of a particle in that pixel), while some almost always have a value of 1 (corresponding to the presence of a particle). Note that if we were to draw different values of A for each sample, we would have a roughly flat distribution (although not entirely uniform, due to our constraints on the possible values of A, which introduce some structure).

Each value of C is also a matrix of 0s and 1s. The probability of a pixel having each value depends on its value in the channel pattern and the two error values. Carrying out k-medoids clustering, with two clusters, on C results in a pair of pair of coordinates, similar to A.

In Fig. 6, we fixed A (to be the same as for Fig. 5), then drew B randomly 10 000 times, then drew C once for each value of B (using a Type I error of 0.05 and a Type II error of 0.2), and then calculated D for each value of C. This gave us 10 000 samples of the calculated cluster centers for a specific, fixed value of A. We then plotted the proportions of the samples in which each pixel contained one of the calculated cluster centers.

If the protocol were ideal, these calculated cluster centers would always be at the coordinates given by A, and so the positions of the cluster centers would be a perfect predictor of the positions of the attractors (the value of A). Figure 6 shows that the actual cluster centers found are spread over a small region around the true positions of the attractors, meaning that D has some entropy, even for fixed A, although the fact that most of the locations of the cluster centers are close to the true positions means that the protocol does gain some information about them. For further details about the calculation of the mutual information between variables A and D, see Appendix B.

To simplify our comparison of the protocols, we make the following assumptions: the attractors are well separated from each other (i.e., the distance between each attractor and its nearest neighbor is much greater than σ), none of the attractors are close to the edge of the surface (i.e., the distance between each attractor and the nearest edge is much greater than σ), and the attractors are each located at the center of a pixel. The first two conditions simplify the
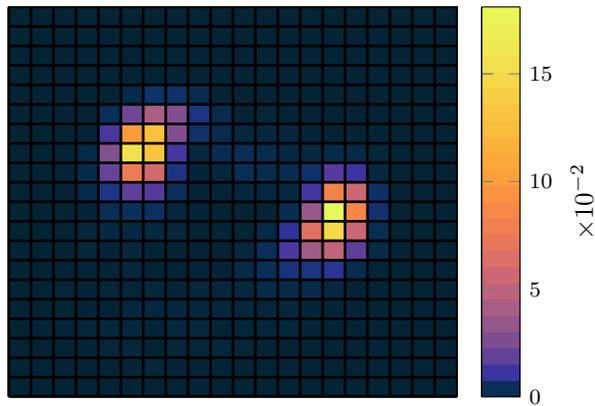
FIG. 6. Proportion of (10 000) samples in which each pixel is predicted to contain an attractor (i.e., to be the center of a cluster), for the scenario in Sec. IV A. The predicted locations follow a distribution that has a high probability close to the true locations and a low probability elsewhere. The less spread out this distribution, the less uncertainty we have about the locations of the attractors, and hence the better the imaging protocol. This measurement outcome corresponds to a Type I error of 0.05 and a Type II error of 0.2. Note that this plot does not show the same thing as Fig. 5; Fig. 5 shows the proportion of samples in which each pixel contains a particle (and multiple pixels will contain a particle in each sample), while this plot shows the proportion of samples in which each pixel is found to be a cluster center (and each sample contains only, and exactly, two cluster centers).

probability function, while the third allows us to use the discrete Shannon entropy, rather than the Shannon entropy for continuous variables. The first two conditions are likely to hold if $d \gg \sigma$. We set $d = 20$, $\phi = 1$, and $\sigma^2 = 2$. We then simulate measurement results for pairs of Type I and Type II errors and hence calculate the mutual information for each pair of errors.

For any given Type I error, a higher Type II error means fewer of the particles will be detected. Conversely, a higher Type I error means pixels will be found to contain particles when they actually do not. Both types of error are expected to reduce the accuracy with which we can estimate the cluster centers, since in one case we have less information and in the other case we have misleading information. We would therefore expect that a lower value of either error would result in Fig. 6 having a sharper (less spread-out) distribution.

Figure 7 shows, as expected, that by measuring the pixels with quantum, rather than classical states, we can achieve a higher mutual information with the ground truth, and therefore can gain more information about the positions of the cluster centers. Recall that, for each Type I error, the Type II error for each type of protocol is given by the curves in Fig. 3, and that the classical Type II error varies between approximately 0.1899 and 0.3918,
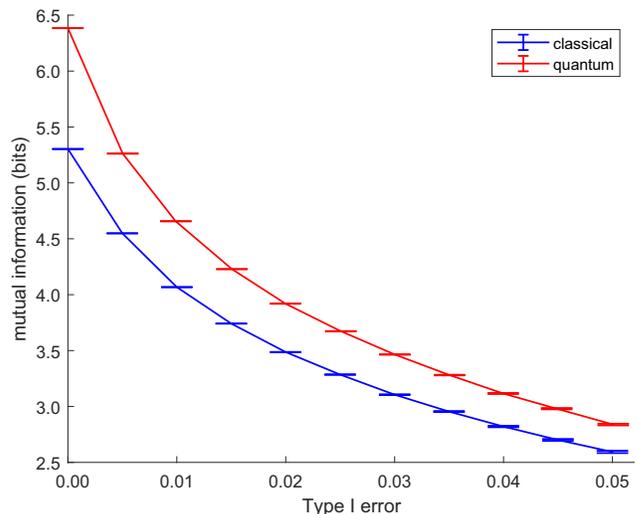
FIG. 7. Mutual information between the ground truth and the estimated result for a pattern detection task involving k-medoids clustering, for both classical and quantum measurements of pixels. For each value of the Type I error, measurement results are simulated for the corresponding Type II error, for each type of measurement, and k-medoids clustering is carried out on the results. The error bars show the variance of our estimator of the mutual information.

while the quantum Type II error varies between approximately 0.1107 and 0.1424. Note that the mutual information decreases as the Type I error increases, for both types of protocol, despite the fact that the Type II error decreases as the Type I error increases.

B. Classical versus quantum-classical DBSCAN clustering

In the second scenario, we are again imaging a surface that can be represented as a grid of pixels. Suppose this surface has long, thin (noncircular) particles on it, that cover multiple pixels. For simplicity, we will assume that the particles are rectangular, with integer dimensions (in terms of pixels covered) d_1 and d_2, and that they are oriented in one of two ways (vertically or horizontally). The particles are distributed randomly over the surface, with the only constraints being that their corners lie at the corners of pixels (so that every pixel is either completely covered or not covered) and that they do not overlap.

Our task is to determine the number of such particles, which is randomly chosen from a uniform distribution between 0 and a maximum number, m. DBSCAN may be more suitable than k-means clustering for identifying clusters corresponding to the particles, due to the noncircular shape of the particles and the fact that the number is not fixed or known beforehand.

We again carry out the numerical simulations as per Fig. 4. The number of particles present is variable A, the

ground truth that we want to discover. For each sample, this is drawn randomly from the uniform distribution between 0 and m. For a given value of A, we then randomly place the clusters on the grid of pixels to generate variable B, the channel pattern (with the constraint that no two particles overlap). After generating a measurement result (variable C) based on the channel pattern and the Type I and II error values, we carry out DBSCAN on the measurement result to try and identify the particles. We use the number of clusters identified by DBSCAN as variable D, our estimate of variable A. For further details about the calculation of the mutual information between variables A and D, see Appendix B.

We set the number of pixels to 50×50, m to 10, and the dimensions of the particles to 2×5 pixels. For the DBSCAN algorithm, we set the minimum number of

points in a region surrounding a point to identify the central point as a core point to 4, and we set the radius of the region to $\sqrt{2}$, so that it includes all neighboring pixels (including diagonal neighbors).

Figure 8 shows an example measurement result for a Type I error of 0.05 and a Type II error of 0.2. The colored pixels are those identified as being part of a cluster, and hence as containing a particle, while the black pixels are those that have a value of 1 in the measurement result but are not part of a cluster (and hence are assumed to be false positives).

In this example, we identify six clusters, and hence this is our estimate of the number of particles. However, there are actually only five particles; all of the pixels in the magenta cluster at the bottom right of the image are false positives. This shows how a high Type I error can lead to the protocol identifying clusters that do not exist, and hence overestimating the number of particles present. Similarly, a high Type II error could lead to a cluster not being identified at all, and hence to the protocol underestimating the number of clusters. Type II errors can also lead to overestimations, if the pixels falsely detected to have a value of 0 lie in the middle of a particle. In this case, the two ends of the particle may be incorrectly found to be separate clusters. Both types of error can therefore lead to a misestimation of the number of particles, and hence will reduce the mutual information between A and D.

Figure 9 again shows that measurements with entangled states achieve a higher mutual information, and are therefore better at determining the number of particles,

FIG. 8. Example of a measurement result for the scenario in Sec. IV B. Every pixel has been individually imaged and found to have a value of either 0 or 1 (each corresponding to a different possible channel). The particles we are looking for are five pixels long and two pixels wide. All of the pixels containing a particle should have the value 1 and all other pixels should have a value of 0. However, sometimes, we will determine a pixel to have value 0 when its actual value is 1 (Type II error). Similarly, sometimes we will get a value of 1 for a pixel that does not contain a particle (Type I error). In order to find which pixels that we determine to have value 1 actually contain particles and which do not, we carry out DBSCAN on the measurement results to find clusters of 1s and only consider 1s that are part of clusters to be real particles rather than measurement errors. All pixels found to have a value of 0 are colored white. Pixels (found to have a value of 1) that are determined to belong to the same cluster are given the same (nonblack) color, while the remaining, unclustered pixels are colored black.

FIG. 9. Mutual information between the ground truth and the estimated result for a pattern detection task involving DBSCAN, for both classical and quantum measurements of pixels. For each value of the Type I error, measurement results are obtained for the corresponding Type II error, for each type of measurement, and DBSCAN is carried out on the results. The error bars show the variance of our estimator of the mutual information.

than classical measurements. For both types of protocols, the mutual information initially rises as the Type I error increases before falling again. This is not unexpected because the Type II error decreases as the Type I error increases (see Fig. 3), so the increase in mutual information is not because of the higher Type I error but because of the lower Type II error.

In particular, for low values of the Type I error, we have a clear quantum advantage (for a Type I error of 0, the advantage is more than one bit). The large gap between the protocols, relative to their absolute values, shows that the advantage in the misdetection probability due to using quantum probes can lead to improvements in clustering accuracy for certain unsupervised learning tasks. Note that low values of the Type I error correspond to larger differences in the Type II error between the two types of protocols (see Fig. 3), although there was no guarantee that this would translate to a higher quantum advantage for lower values of the Type I error.

V. CONCLUSION

Quantum states can often give an advantage over classical states when it comes to quantum imaging tasks. In this paper we have shown that this advantage can translate to an improvement in clustering accuracy when classical clustering algorithms are performed on the results of a measurement. We demonstrated this advantage numerically for both a scenario involving the k-medoids algorithm (a centroid-based algorithm) and one involving DBSCAN (a density-based method). In both cases, using a quantum measurement, and therefore achieving a lower Type II error for the same Type I error, resulted in a higher mutual information between the ground truth that we wanted to know and our estimation of the ground truth based on the measurement protocol.

While this result is intuitive, a small advantage in misdetection probability could easily have been lost or made negligible during the data-processing stage. It is therefore encouraging that the quantum advantage for measurements is robust enough to survive the clustering algorithms. This complements existing results showing that imaging using quantum states can give improved results for supervised learning.

Moreover, we found that different clustering algorithms and different scenarios result in different levels of quantum advantage, even for the same pairs of possible channels. Comparing Figs. 7 and 9 shows that the same quantum advantage in measurements can lead to different quantum advantages in clustering accuracy when applied to different problems involving different clustering techniques. Indeed, the two graphs are qualitatively different, as well as quantitatively: in Fig. 7, the mutual information decreases as the Type I error increases for both the classical and the quantum-classical cases, while in Fig. 9, both mutual

informations initially peak before decreasing again. This demonstrates that the question raised in this paper is nontrivial.

Since ground truths such as the number of clusters on a surface are global properties of the entire surface, future research could consider the possibility that fully quantum protocols that collectively probe the pixel pattern as a whole, rather than individually probing each pixel, could give a further advantage over quantum-classical protocols.

Another possible consideration is protocols that involve multiple rounds of sending probes through the pixels to image them before the clustering stage. Such protocols could potentially also be adaptive, meaning that the probes used in subsequent rounds could depend on the results of measuring the return states from previous rounds. Adaptivity is known to provide an advantage for certain channel discrimination problems.

Further research could also consider situations in which there is a global constraint on the energy used to probe the entire surface, but no firm constraint on the per-pixel energy used (i.e., the sum of the energies of all of the probes is fixed, but not the distribution over the pixels).

ACKNOWLEDGMENTS

APPENDIX A: CALCULATING THE RECEIVER OPERATING CHARACTERISTIC

Suppose we want to discriminate between a pair of pure loss channels, $\mathcal{L}_1$ and $\mathcal{L}_2$, with transmissions of τ_1 and τ_2 respectively, by probing the target channel with either a classical probe or a quantum probe. In both cases, we require that the average photon number of the part of the state that passes through the channel is at most m. For the classical case, we will optimize over all classical mixtures of coherent states, and therefore find the exact optimum ROC. For the quantum case, we will consider a TMSV probe, and will therefore find an achievable upper bound on the optimum.

1. Classical ROC

First, let us consider the output of a single coherent state sent through a pure loss channel with a transmission of τ. Let $\left| \sqrt{n} \right\rangle$ be a coherent state with an average photon number of n. To fully specify such a state, we should also

give a phase, but the phase is not important for our calculations. Sending such a state through the channel results in the pure output state $\left|\sqrt{\tau n}\right\rangle$. The fidelity between the outputs of channels $\mathcal{L}_1$ and $\mathcal{L}_2$ is [23]

$$F_{\text{coh}} = \chi^n, \quad \chi = \exp\left[-\frac{1}{2}|\sqrt{\tau_1} - \sqrt{\tau_2}|^2\right]. \quad \text{(A1)}$$

Classical states can be expressed as classical mixtures of coherent states. From the linearity of quantum channels and the joint concavity of the fidelity, we can write

$$F_{\text{mix}} \geq f\left(p\left(n\right)\right) = \int_{n=0}^{\infty} p(n)^2 \chi^n dn, \quad \text{(A2)}$$

where $p(n)^2$ is the probability density function for input states with an average photon number of n [and we have squared $p(n)$ to enforce non-negativity]. We would like to minimize the functional $f(p)$ subject to the normalization condition

$$\int_{n=0}^{\infty} p(n)^2 dn = 1, \quad \text{(A3)}$$

and the energy constraint

$$\int_{n=0}^{\infty} p(n)^2 n dn = m. \quad \text{(A4)}$$

We define the Lagrangian functional

$$\mathcal{L}(p(n), \mu, \lambda) = \int_{n=0}^{\infty} L(p, \mu, \lambda) dn - (\mu + \lambda m), \quad \text{(A5)}$$

$$L(p(n), \mu, \lambda) = p(n)^2 (\chi^n + \mu + \lambda n), \quad \text{(A6)}$$

where μ and λ are Lagrange multipliers. For a function to be a minimum of the constrained optimization, it must be a stationary point of the Lagrangian functional [a point is here defined as a specific choice of $p(n)$, μ, and λ].

The stationary points of a functional are those functions for which the functional derivative is 0. The functional derivative of L is

$$\frac{\partial L}{\partial p(n)} = 2p(n)(\chi^n + \mu + \lambda n). \quad \text{(A7)}$$

Since μ and λ are numbers, rather than functions of n, we can only set $\chi^n + \mu + \lambda n = 0$ for at most two different values of n (since it is a continuous function of n with at most one turning point). To set $dL/dp(n) = 0$, we must have $p(n) = 0$ for all but two (or fewer) values of n (except in the trivial case of $\tau_1 = \tau_2$). Applying the continuity condition and the energy constraint, the candidate solutions take

the form

$$p_{n_0,p_0}(n)^2 = p_0 \delta_{n,n_0} + (1 - p_0)\delta_{n,n_0+\Delta(n_0,p_0)},$$

$$\Delta(n_0, p_0) = \frac{m - n_0}{1 - p_0}, \quad 0 \leq p_0 < 1, \ 0 \leq n_0 \leq m, \quad \text{(A8)}$$

where δ is the Kronecker delta function.

We now minimize $f(p)$ over our candidate solutions. We calculate

$$f(p_{n_0,p_0}) = \chi^{n_0}\left(p_0 + (1 - p_0)\chi^{\Delta(n_0,p_0)}\right), \quad \text{(A9)}$$

and partially differentiating with respect to n_0, we get

$$\frac{\partial f}{\partial n_0} = p_0 \ln[\chi]\chi^{n_0}\left(1 - \chi^{\Delta(n_0,p_0)}\right). \quad \text{(A10)}$$

This is always negative if $p_0 > 0$ and $n_0 < m$, so to minimize we can set $n_0 = m$. The minimum fidelity can therefore be achieved by using a pure coherent state with an average photon number of m. The minimum fidelity between output states for a classical probe is

$$F_{\text{class}} = \chi^m. \quad \text{(A11)}$$

Finally, for discriminating between pure states with a fidelity of F, the minimum Type I error, α, for a given Type II error, β, is given by [24]

$$\alpha = \beta - 2\beta F^2 + F\left(F - 2\sqrt{(1 - \beta)\beta(1 - F^2)}\right). \quad \text{(A12)}$$

2. Quantum ROC

For a TMSV probe, the initial covariance matrix of the probe (before one mode is sent through the channel) is

$$V^{\text{in}} = \begin{pmatrix} (2m+1)\mathbf{I} & 2\sqrt{m(m+1)}\mathbf{Z} \\ 2\sqrt{m(m+1)}\mathbf{Z} & (2m+1)\mathbf{I} \end{pmatrix}, \quad \text{(A13)}$$

where m is the average photon number of each of the modes. The covariance matrix of the return state is

$$V_{(i)}^{\text{out}} = \begin{pmatrix} (2m+1)\mathbf{I} & 2\sqrt{\tau_i m(m+1)}\mathbf{Z} \\ 2\sqrt{\tau_i m(m+1)}\mathbf{Z} & (2m\tau + 1)\mathbf{I} \end{pmatrix}, \quad \text{(A14)}$$

where i depends on the identity of the channel. This covariance matrix can be diagonalized by a two-mode squeezing (unitary) operation, with the squeezing parameter depending on τ_i. The resulting state is the tensor product of a vacuum state and a thermal state, with average photon number $\bar{n}_i$ (which again depends on τ_i). Let us call the

channel output ρ'_i, and let us call the diagonalizing unitary for that output state U_i. Then

$$U_i \rho'_i U_i^\dagger = |0\rangle\langle 0| \otimes \sigma_i,$$

$$\sigma_i = \sum_{n=0}^{\infty} \frac{\bar{n}_i^n}{(\bar{n}_i + 1)^{n+1}} |n\rangle\langle n|. \tag{A15}$$

Consequently, we can apply U_1 to our output state, defining $\rho_i = U_1 \rho'_i U_1^\dagger$, such that

$$\rho_1 = |0\rangle\langle 0| \otimes \sigma_1,$$

$$\rho_2 = U^\dagger(|0\rangle\langle 0| \otimes \sigma_2)U, \quad U = U_2 U_1^\dagger. \tag{A16}$$

Note that, as a composition of two-mode squeezing operations, U is also a two-mode squeezing operation, and so can be written as [25]

$$U(r) = \exp\left[r(\hat{a}\hat{b} - \hat{a}^\dagger \hat{b}^\dagger)\right]. \tag{A17}$$

Now, suppose we carried out a photon counting measurement on the first mode of state ρ_i. If $i = 1$, we always get a result of 0 (a Type I error of 0). On the other hand, if $i = 2$, we get a result of 0 with some nonzero probability (a nonzero Type II error). This is one possible measurement scheme.

Suppose, on the other hand, we carried out a photon-counting measurement on the first mode of state $U\rho_i U^\dagger$ (i.e., apply the two-mode squeezing unitary, U, prior to the measurement). Now, if $i = 1$, we get a result of 0 some of the time, but if $i = 2$, we always get a result of 0. In this case, therefore, we get a nonzero Type I error and a Type II error of 0. This defines another possible measurement scheme. Note that in both schemes, we only care about whether the result is 0 or not 0 (not the actual number), so this is more akin to a click detector.

On a plot of Type I error against Type II error (ROC), we can draw a straight line between these two points, and achieve any pair of errors along this line. This could be achieved by carrying out the first measurement with probability a and the second measurement with probability $1 - a$. Suppose we want a better than linear interpolation between the two points. How might we go about designing a measurement to achieve this?

One possibility, instead of choosing one measurement or the other with some classical probability, might be to control which measurement is carried out using a quantum state. Suppose we apply a controlled unitary to the state ρ_i, so that if the control qubit is $|0\rangle$, we apply the identity to ρ_i, and if the control qubit is $|1\rangle$, we apply U. Let us denote the resulting channel as $\mathcal{U}$, and write

$$
\begin{aligned}
\mathcal{U} : |a\rangle\langle a| \otimes \rho_i \to{} & a\, |0\rangle\langle 0| \otimes \rho + \sqrt{a(1-a)} \\
& \times \left(|0\rangle\langle 1| \otimes \rho_i U^\dagger + |1\rangle\langle 0| \otimes U\rho_i\right) \\
& + (1-a)\, |1\rangle\langle 1| \otimes U\rho U^\dagger, \tag{A18}
\end{aligned}
$$

where $|a\rangle = \sqrt{a}\,|0\rangle + \sqrt{1-a}\,|1\rangle$ is a control qubit.

If we now measured the control qubit and then the first mode, this would reduce to the classical combination of measurements, but by retaining the off-diagonal components, we may be able to do better. To reduce the complexity of the problem, let us apply the following channel to the return state (and the identity to the control qubit), mapping it from a continuous variable (CV) state to a discrete variable (DV) state, while retaining the superposition of measurements:

$$\mathcal{C}_{\mathrm{CV}\to\mathrm{DV}}[\rho] = \sum_k K_k \rho K_k^\dagger, \tag{A19}$$

$$\{K_k\} = \{|00\rangle\langle 00|, |01\rangle\langle 0i|, |10\rangle\langle i0|, |11\rangle\langle ij|\}, \tag{A20}$$

where i and j both run from 1 to ∞. In other words, this channel, when applied to a two-mode Gaussian state, maps it to a two-qubit state by mapping all of the nonvacuum components to the single outcome $|1\rangle$. Note that, by doing this, we lose some information, but not all of it, and that we have not traced over the second mode, even though the original two protocols trace over it (by only carrying out photon counting on the first mode). The reason we do not simply trace over the second mode is to retain as much information as possible, while still having a state that is easier to work with.

Let us calculate the resulting three-qubit states, $\rho_i^{\mathrm{out}} = (\mathbf{I} \otimes \mathcal{C}_{\mathrm{CV}\to\mathrm{DV}}) \cdot \mathcal{U}[|a\rangle\langle a| \otimes \rho_i]$. We will do this component by component and state by state. Let us start with

$$
\begin{aligned}
\mathcal{C}_{\mathrm{CV}\to\mathrm{DV}}[\rho_1] ={} & \langle 0|\sigma_1|0\rangle\,|00\rangle\langle 00| \\
& + (1 - \langle 0|\sigma_1|0\rangle)\,|01\rangle\langle 01| \\
={} & x_1\,|00\rangle\langle 00| + (1 - x_1)\,|01\rangle\langle 01|, \tag{A21}
\end{aligned}
$$

where we define $x_i = (\bar{n}_i + 1)^{-1}$.

Next, we will calculate $\mathcal{C}_{\mathrm{CV}\to\mathrm{DV}}[U\rho_1]$, using the Fock basis expansion of σ_1 in Eq. (A15):

$$\mathcal{C}_{\mathrm{CV}\to\mathrm{DV}}[U\rho_1] = \mathcal{C}_{\mathrm{CV}\to\mathrm{DV}}\left[\sum_{m=0}^{\infty} \frac{\bar{n}_1^m\, U\,|0m\rangle\langle 0m|}{(\bar{n}_1 + 1)^{m+1}}\right]. \tag{A22}$$

The key quantity we need to calculate here is

$$\langle i,j\,|U|0,m\rangle\langle 0,m|i,j\rangle = \delta_{i0}\delta_{jm}\,\langle 0,m|U|0,m\rangle. \tag{A23}$$

We can decompose U as [26]

$$U(r) = \cosh(r)^{-1} \exp\left[-\tanh(r)\hat{a}^\dagger\hat{b}^\dagger\right]$$
$$\times \exp\left[-\ln(\cosh(r))(\hat{a}^\dagger\hat{a} + \hat{b}^\dagger\hat{b})\right]$$
$$\times \exp\left[\tanh(r)\hat{a}\hat{b}\right], \quad (A24)$$

which is much simpler to use, since we apply the lowering and raising operators in a fixed order, rather than in every possible sequence. The resultant state is [27]

$$U(r)\,|0, m\rangle = \cosh(r)^{-(m+1)}$$
$$\times \sum_{k=0}^\infty \left[\binom{k+m}{m}^{1/2}(-\tanh(r))^k\,|k\rangle\,|k+m\rangle\right]. \quad (A25)$$

Consequently,

$$\langle i, j\,|U|0, m\rangle\,\langle 0, m|i, j\rangle = \delta_{i0}\delta_{jm}\cosh(r)^{-(m+1)}, \quad (A26)$$

and so

$$\mathcal{C}_{\text{CV}\to\text{DV}}[U\rho_1] = x_1 y\,|00\rangle\langle 00| + \frac{(1-x_1)x_1 y^2}{1-(1-x_1)y}\,|01\rangle\langle 01|, \quad (A27)$$

where $y = \cosh(r)^{-1}$.

We now calculate $\mathcal{C}_{\text{CV}\to\text{DV}}[U\rho_1 U^\dagger]$, again making use of Eq. (A25). We can write

$$\langle ij\,|U|0m\rangle\,\langle 0m|U^\dagger|ij\rangle$$
$$= \delta_{j-i,m}\cosh(r)^{-2(m+1)}\tanh(r)^{2i}\binom{i+m}{m}. \quad (A28)$$

Then, setting $i = 0$ and $j = 0$, we get

$$\langle 00|U\rho U^\dagger|00\rangle = x_1 y^2. \quad (A29)$$

Setting $i = 0$ and instead summing over j, we get

$$\sum_{j=1}^\infty \langle 0j\,|U\rho U^\dagger|0j\rangle = \frac{(1-x_1)x_1 y^4}{1-(1-x_1)y^2}. \quad (A30)$$

Finally, noting that, if $i \neq 0$ and $j = 0$, the right-hand side of Eq. (A28) goes to 0, we can write

$$\sum_{i,j=1}^\infty \langle ij\,|U\rho U^\dagger|ij\rangle$$
$$= 1 - \langle 00|U\rho U^\dagger|00\rangle - \sum_{j=1}^\infty \langle 0j\,|U\rho U^\dagger|0j\rangle$$
$$= \frac{1-y^2}{1-(1-x_1)y^2}. \quad (A31)$$

Therefore, our expression for $\mathcal{C}_{\text{CV}\to\text{DV}}[U\rho_1 U^\dagger]$ is

$$\mathcal{C}_{\text{CV}\to\text{DV}}[U\rho_1 U^\dagger] = x_1 y^2\,|00\rangle\langle 00|$$
$$+ \frac{(1-x_1)x_1 y^4}{1-(1-x_1)y^2}\,|01\rangle\langle 01|$$
$$+ \frac{1-y^2}{1-(1-x_1)y^2}\,|11\rangle\langle 11|. \quad (A32)$$

Putting all of these elements together, we can write the three-qubit state

$$\rho_{1,a}^{\text{out}} = \text{diag}\left[ax_1, a(1-x_1), 0, 0, (1-a)x_1 y^2,\right.$$
$$\left.\frac{(1-a)(1-x_1)x_1 y^4}{1-(1-x_1)y^2}, 0, \frac{(1-a)(1-y^2)}{1-(1-x_1)y^2}\right]$$
$$+ \begin{pmatrix} \mathbf{0}_4 & \boldsymbol{\omega}_1 \\ \boldsymbol{\omega}_1 & \mathbf{0}_4 \end{pmatrix}, \quad (A33)$$

where $\mathbf{0}_4$ is the 4×4 zero matrix and we define

$$\boldsymbol{\omega}_i = \begin{pmatrix} \sqrt{a(1-a)}x_i y & 0 & 0 & 0 \\ 0 & \frac{\sqrt{a(1-a)}(1-x_i)x_i y^2}{1-(1-x_i)y} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (A34)$$

Similarly, we can write

$$\rho_{2,a}^{\text{out}} = \text{diag}\left[ax_2 y^2, \frac{a(1-x_2)x_2 y^4}{1-(1-x_2)y^2}, 0, \frac{a(1-y^2)}{1-(1-x_2)y^2},\right.$$
$$\left.(1-a)x_2, (1-a)(1-x_2), 0, 0\right]$$
$$+ \begin{pmatrix} \mathbf{0}_4 & \boldsymbol{\omega}_2 \\ \boldsymbol{\omega}_2 & \mathbf{0}_4 \end{pmatrix}. \quad (A35)$$

Thus, we have reduced a two-mode continuous variable system to a sparse (nine-element), three-qubit state, for which the optimal errors can be more easily calculated. For a given value of the parameter a, we can calculate optimal values of the Type I and II errors using

$$\alpha(a, b) = \text{Tr}[(\mathbf{1} - \Pi_{a,b})\rho_{1,a}^{\text{out}}], \quad (A36)$$
$$\beta(a, b) = \text{Tr}[\Pi_{a,b}\rho_{2,a}^{\text{out}}], \quad (A37)$$
$$\Pi_{a,b} = \{(1-b)\rho_{2,a}^{\text{out}} - b\rho_{1,a}^{\text{out}}\}_-, \quad (A38)$$

where $\{X\}_-$ is the projector onto the negative eigenspace of X.

Even for our reduced system, it is still difficult to analytically find the ROC by optimizing over parameter a for

each value of b. Instead, we numerically sample a large number of a and b values to get a large number of pairs $\{\alpha(a,b), \beta(a,b)\}$, then join up the bottom of this set of values to approximate the ROC curve.

APPENDIX B: CALCULATION OF THE MUTUAL INFORMATION

In both scenarios, our aim is to calculate the mutual information between the ground truth (variable A) and the estimate of the ground truth (variable D). For both scenarios, variables A and D are the same size: in Sec. IV A, they are pairs of coordinates (A is composed of the true locations of the two attractors and D is composed of the locations of the two cluster centers as found by the clustering algorithm), and in Sec. IV B, they are integers between 0 and 10 (A is the number of particles on the surface and D is the number of clusters found by DBSCAN). Note that DBSCAN may find more than 10 clusters; in this case we replace the actual number of clusters found with 10, since we know this is the maximum number of particles.

Let us first discuss how we can determine the entropy of a probability distribution over a finite number of outcomes by sampling the distribution a large number of times. If we (independently) sample the distribution a large number of times, we can approximate the true probability distribution using the number of occurrences of each outcome (i.e., the probability of an outcome is approximately the number of occurrences divided by the total number of samples). We call this approximated probability distribution the sample distribution. We can then calculate the entropy of the sample distribution (the sample entropy) in order to approximate the entropy of the true probability distribution. This method is called the plugin estimator.

In the asymptotic limit of $N \gg P$ samples, where P is the number of possible outcomes, the value we calculate will be normally distributed around an expected value. This expected value is not the same as the true value of the entropy, since our estimator is biased: for any finite N, the expected value of the plugin estimator is less than the true entropy. We have the following conditions for the variance and bias of the estimator [28,29]:

$$\text{var}(\hat{H}_N) \leq \frac{\log_2^2 N}{N}, \tag{B1}$$

$$\mathbf{E}[\hat{H}_N] = H - \frac{P-1}{2N} \log_2 e + \mathcal{O}[N^{-2}], \tag{B2}$$

where H is the true entropy and $\hat{H}_N$ is the sample entropy for N samples.

We first consider the scenario in Sec. IV B. Here, we generate N samples according to the flow chart in Fig. 4, recording variables A and D for each. We then calculate the sample entropy, $\hat{H}_N(D)$, and the sample conditional entropy, $\hat{H}_N(D|A)$. The conditional entropy is given by

$$H(D|A) = \sum_a p(A = a)H(D|A = a), \tag{B3}$$

and so, recalling that A has a uniform probability over all outcomes, we can write

$$\hat{H}_N(D|A) = \frac{1}{m+1} \sum_{a=0}^{m} \hat{H}_{N/(m+1)}(D|A = a), \tag{B4}$$

where m is the maximum number of clusters. Note that $H(D|A = a)$ is the entropy conditioned on a particular outcome of A, and therefore is calculated using only $1/(m+1)$ of the total number of samples.

The plugin estimator of the conditional entropy is a sum of independent normal distributions and therefore has the following conditions on its variance and bias:

$$\text{var}(\hat{H}_N(D|A)) \leq \frac{(m+1)\log_2^2[N/(m+1)]}{N}, \tag{B5}$$

$$\mathbf{E}[\hat{H}_N(D|A)] = H(D|A) - \frac{m(m+1)}{2N} \log_2 e + \mathcal{O}[N^{-2}]. \tag{B6}$$

The plugin estimator of the mutual information is given by

$$\hat{I}_N(A:D) = \hat{H}_N(D) - \hat{H}_N(D|A), \tag{B7}$$

and so has the following conditions on its variance and bias:

$$\text{var}(\hat{I}_N(A:D)) \leq \frac{\log_2^2 N + (m+1)\log_2^2[N/(m+1)]}{N}, \tag{B8}$$

$$\mathbf{E}[\hat{I}_N(A:D)] = I(A:D) + \frac{m^2}{2N} \log_2 e + \mathcal{O}[N^{-2}]. \tag{B9}$$

Recalling that $m = 10$, and setting the number of samples, N, to 20 000, we find that the variance of our estimator is less than or equal to approximately 0.0126 bits and the bias is approximately -0.0004 bits.

Next, let us consider how to calculate the entropy for the scenario in Sec. IV A. The difficulty in this scenario is that both A and D are sets of m coordinates in a $d \times d$ grid. This means that the dimension of each is the number of unique choices of m positions out of d^2, which is the binomial coefficient $\binom{d^2}{m}$. For a 20×20 grid and two clusters, this means that A and D each have 79 800 possible outcomes. This is a lot more than the 11 possible outcomes for A and D for the scenario in Sec. IV B, and so we need a lot more samples for the plugin estimator, $\hat{H}_N(D)$, to be in the asymptotic regime ($N \gg P$). In order to estimate the conditional entropy in the same way, we would need roughly 80 000 times more samples than even this, because

we would be sampling the entropy of D for each possible outcome of A.

Instead, we make an assumption about the conditional entropy in order to simplify the calculations. We assume that the entropy of D conditioned on a single value of A, $H(D|A = a)$, is approximately the same for all permitted values of A. This assumption is justified by the conditions we imposed, which state that the attractors are well separated from each other and are not close to the edge of the grid, so the distributions due to each of the attractors should not interfere with each other.

As a result, we estimate the entropy of D for a small number (5) of randomly chosen fixed values of A, and use the mean value of these to approximate the conditional entropy, $H(D|A)$. We then assume that the variance of the values we get is approximately the variance of $H(D|A = a)$, that is, the variance of the entropy of D conditioned on specific values of A. Note that the variance of our estimator of $H(D|A = a)$ follows Eq. (B1), and so is negligible in the asymptotic regime (we take 800 000 samples, so that $P/N < 1/10$).

We estimate $H(D)$ in the normal way, by randomly choosing values of A for each sample and calculating corresponding values for D. Since we use the same number of samples for our estimation of each $H(D|A = a)$ as we do for our estimation of $H(D)$, the biases cancel out, so we have an unbiased estimator of the mutual information, whose variance is approximately the variance of our estimated values of $H(D|A = a)$.

---

[1] C. M. Bishop *Pattern Recognition and Machine Learning*, Information Science and Statistics (Springer-Verlag, Berlin, 2006).

[2] V. Estivill-Castro, Why so many clustering algorithms: A position paper, ACM SIGKDD Explor. Newsl. **4**, 65 (2002).

[3] M. Genovese, Real applications of quantum imaging, J. Opt. **18**, 073002 (2016).

[4] M. Abobeih, J. Randall, C. Bradley, H. Bartling, M. Bakker, M. Degen, M. Markham, D. Twitchen, and T. Taminiau, Atomic-scale imaging of a 27-nuclear-spin cluster using a quantum sensor, Nature **576**, 411 (2019).

[5] I. V. Soldatov and R. Schäfer, Advances in quantitative Kerr microscopy, Phys. Rev. B **95**, 014426 (2017).

[6] C. Ounkomol, S. Seshamani, M. M. Maleckar, F. Collman, and G. R. Johnson, Label-free prediction of three-dimensional fluorescence images from transmitted-light microscopy, Nat. Methods **15**, 917 (2018).

[7] M. E. Kandel, Y. R. He, Y. J. Lee, T. H.-Y. Chen, K. M. Sullivan, O. Aydin, M. T. A. Saif, H. Kong, N. Sobh, and G. Popescu, Phase imaging with computational specificity (PICS) for measuring dry mass changes in sub-cellular compartments, Nat. Commun. **11**, 6256 (2020).

[8] S. Pirandola, Quantum Reading of a Classical Digital Memory, Phys. Rev. Lett. **106**, 090504 (2011).

[9] L. Banchi, Q. Zhuang, and S. Pirandola, Quantum-Enhanced Barcode Decoding and Pattern Recognition, Phys. Rev. Appl. **14**, 064026 (2020).

[10] C. Harney, L. Banchi, and S. Pirandola, Ultimate limits of thermal pattern recognition, Phys. Rev. A **103**, 052406 (2021).

[11] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, Quantum machine learning, Nature **549**, 195 (2017).

[12] V. Giovannetti, S. Lloyd, and L. Maccone, Advances in quantum metrology, Nat. Photonics **5**, 222 (2011).

[13] S. Lloyd, M. Mohseni, and P. Rebentrost, Quantum algorithms for supervised and unsupervised machine learning, Preprint ArXiv:1307.0411 (2013).

[14] S. Gambs, Quantum classification, Preprint ArXiv:0809.0444 (2008).

[15] G. Sentís, A. Monràs, R. Muñoz-Tapia, J. Calsamiglia, and E. Bagan, Unsupervised Classification of Quantum Data, Phys. Rev. X **9**, 041029 (2019).

[16] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevApplied.19.054031 for the code used to generate the plots.

[17] J. L. Pereira, L. Banchi, Q. Zhuang, and S. Pirandola, Idler-free channel position finding, Phys. Rev. A **103**, 042614 (2021).

[18] S. Lloyd, Least squares quantization in PCM, IEEE Trans. Inf. Theory **28**, 129 (1982).

[19] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD'96* (AAAI Press, Portland, Oregon, 1996), p. 226.

[20] L. Kaufman and P. J. Rousseeuw, in *Finding Groups in Data* (John Wiley & Sons, New York, 1990), Chap. 2, p. 68.

[21] N. Tishby, in *Proc. 37th Annual Allerton Conference on Communications, Control and Computing, 1999* (University of Illinois at Urbana-Champaign, Monticello, Illinois, 1999), p. 368.

[22] L. Banchi, J. Pereira, and S. Pirandola, Generalization in Quantum Machine Learning: A Quantum Information Standpoint, PRX Quantum **2**, 040321 (2021).

[23] C. Weedbrook, S. Pirandola, R. García-Patrón, N. J. Cerf, T. C. Ralph, J. H. Shapiro, and S. Lloyd, Gaussian quantum information, Rev. Mod. Phys. **84**, 621 (2012).

[24] J. L. Pereira, L. Banchi, and S. Pirandola, Analytical Bounds for Nonasymptotic Asymmetric State Discrimination, Phys. Rev. Appl. **19**, 054030 (2023).

[25] C. M. Caves and B. L. Schumaker, New formalism for two-photon quantum optics. I. Quadrature phases and squeezed states, Phys. Rev. A **31**, 3068 (1985).

[26] B. L. Schumaker and C. M. Caves, New formalism for two-photon quantum optics. II. Mathematical foundation and compact notation, Phys. Rev. A **31**, 3093 (1985).

[27] A. V. Chizhov and B. K. Murzakhmetov, Photon statistics and phase properties of two-mode squeezed number states, Phys. Lett. A **176**, 33 (1993).

[28] A. Antos and I. Kontoyiannis, Convergence properties of functional estimates for discrete distributions, Random Struct. Algorithms **19**, 163 (2001).

[29] G. P. Basharin, On a statistical estimate for the entropy of a sequence of independent random variables, Theory Probab. Appl. **4**, 333 (1959).