# Summer 2023 Writeup

Oscar Scholin*, Alec Roberson†, Richard Zheng†
*Department of Physics, Pomona College, Claremont, CA 91711 and
† Department of Physics, Harvey Mudd College, Claremont, CA 91711

We summarize theoretical and experimental work related to entanglement witnessing. An entanglement witness is a positive semidefinite operator constructed such that all non-entangled (separable) states yield a non-negative expectation value, and there exists at least 1 entangled state that has a negative expectation value. Riccardi et al. [1] define a group of 6 of these witnesses $\{W\}$, and last summer's group expand this list by 9 $\{W'\}$. We first give an overview of the notation and key ideas like entanglement and the witnessing process. Then we describe our efforts to improve a neural network using Python's Keras to optimize the selection of witnesses based on 9 input projective probabilities of the state in question, which demonstrates an increase of 4% in comparison to last year for states generated in the style of [2] satisying $\min\{W\} \geq 0$ and at least one of $\{W'\} < 0$. Despite a variety of model architectures and algorithms, we are unable to improve past this point, which illustrates maximal use of the input statistical features. On the experimental side, we attempt to implement an algorithm we have developed called InstaQ [3] which would allow for the automatic realization of states in the lab based on determining the optimal settings of optical devices via gradient descent. We are unable to satisfactorily produce experimental data of entanglement witnesses matching theory or adjusted theory calculations, despite high fidelity to the adjusted theory states—this would be a potentially worthwhile path for future research, and we include an analysis of all attempts and possible further directions. All code is available on `https://github.com/Lynn-Quantum-Optics/Summer-2023`.

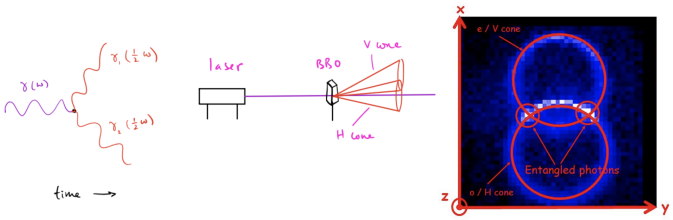


FIG. 1: Illustration of SPDC. *(a–left)* Feynmann Diagram for the creation of two 810nm photons from an incident 405nm. *(b–middle)* Sketch showing the emergence of H and V polarized cones after passing through the BBO. *(c–right)* Vertical slice experimentally showing the two cones and entangled regions (due to [4]). Interestingly, SPDC is actually quite an inefficient process: only about 1 in 1 billion photons is down converted.

## I. INTRODUCTION

The work this summer focused on entanglement witnessing and had two primary components:

1. theoretical work in generating random quantum states and training a variety of machine learning algorithms to predict the optimal $W'$ triplet, a new set of witness values defined during last summer's research with Yang, Verghese, and Hartley, as well as more broadly

2. experimental work creating and measuring states of the form

$$|\psi\rangle = \cos\eta|\Psi^+\rangle + \sin\eta e^{i\chi}|\Psi^-\rangle$$

where $\Psi^\pm$ are Bell states.

This writeup is organized as follows: in the introduction, we give an overview of the theory, in particular various mathematical definitions, that are imperative to understanding the work. In section II, we describe the method of randomization [5] and the machine learning problems we tried to solve. In section III, we present data taken on states, for which experimental waveplate settings were determined via an algorithm we have developed called InstaQ, which uses the Jones Calculus representation of optical implements like half and quarter waveplates.

The single most central idea of this work, and arguably one of the most defining features of quantum mechanics, is *entanglement*. Entanglement is correlation with respect to certain physical properties of a state, like spin or angular momentum or polarization, that persists even under a local [6] change of basis [7]. In other words, these correlations are stronger than classical correlations—the physicist John Bell constructed a mathematical argument which, when tested experimentally [8], confirms this unique behavior [9]. Surprisingly, it is even spatially independent: Bell tests have been conducted with particles beamed to satellites and their partners on the ground, over 120 kilometers away [10].

Entanglement can be created experimentally in optical systems, as in our lab, through a process called *spontaneous parametric down conversion* (SPDC) illustrated in figure 1, by which an incident high energy photon hits two orthogonally-oriented nonlinear crystals and creates two lower energy photons, i.e. $|0\rangle \to |11\rangle$ and $|1\rangle \to |00\rangle$ [4]. Because of this "meiosis" event, the resultant pairs of photons are entangled with respect to their polarization and angular momentum. We can take advantage of the information stored in these persistent relationships and

perform computations. There are many quantum algorithms, such as super dense coding, teleportation, Grover search, that utilize this information stored in the persistent relationships between particles [7]. Therefore, it is central to understand entanglement and how to verify it in both a theoretical and experimental setting.

Before we can define and quantify entanglement, we must discuss the notation of quantum states. Typical in quantum mechanics classes is the Dirac "bra-ket" notation $|\phi\rangle^\dagger = \langle\phi|$, where $^\dagger$ is the complex conjugate, but this only allows for a narrow class of possible states: *pure states*, that is ones that have definite basis representation. What if instead we have a classical ensemble of many different quantum states, or *mixed states*: say, $\{\psi_i\}$ with probabilities $\{p_i\}$ yielding the total state $|\psi\rangle = \sum_i p_i|\psi_i\rangle$? We can represent both with one formalism that is computation friendly: the *density matrix* $\rho = |\psi\rangle\langle\psi|$.[11] By definition, $\rho$ must satisfy:

1. $\mathrm{Tr}\,(\rho) = 1$ (Trace 1)
2. $\rho = \rho^\dagger$ (Hermitian, i.e. all real eigenvalues).
3. $\lambda_i \geq 0$ for all eigenvalues of $\rho$ (positive semi-definiteness)

There are several different measures for entanglement (e.g., the Peres–Horodeck criterion), but the one we selected due to its applicability in any $2^n$ dimensional state is called *concurrence*, which is defined as:

$$C(\rho) = \max\{0, \lambda_1 - \lambda_2 - \lambda_3 - \lambda_4\},$$

where $\lambda_i$ are the eigenvalues sorted in decreasing order of the Hermitian matrix

$$R = \sqrt{\sqrt{\rho}\tilde{\rho}\sqrt{\rho}},$$

where $\rho$ is the density matrix of the state and $\tilde{\rho} = (\sigma_i \otimes \sigma_i)\rho^*(\sigma_i \otimes \sigma_i)$, where $\sigma_i$ is any of the 3 Pauli matrices with $i > 0$ (see section II, $\rho^*$ is the *complex conjugate* (not to be confused with $\rho^\dagger$, which is the *adjoint* or complex conjugate transpose)); we call $\tilde{\rho}$ the *spin-flipped* state [12]. Note that $C(\rho) \in [0, 1]$, which $C = 1$ implies maximal entanglement and $C = 0$ implies no entanglement. The 2-qubit Bell states are a special class of maximally entangled states defined as:

$$|\Phi^\pm\rangle = |00\rangle \pm |11\rangle$$
$$|\Psi^\pm\rangle = |01\rangle \pm |10\rangle.$$

There are two more useful quantities we need to specify. A key question is measure distance in a $2^n$ dimensional system between two quantum states. A common and useful metric is called *fidelity*, defined as:

$$\mathcal{F}(\rho, \sigma) = \mathrm{Tr}\left(\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}}\right).$$

We are also interested in how "mixed" a quantum state is, which we can quantify with a metric called *purity*:

$$\gamma = \mathrm{Tr}\left(\rho^2\right).$$

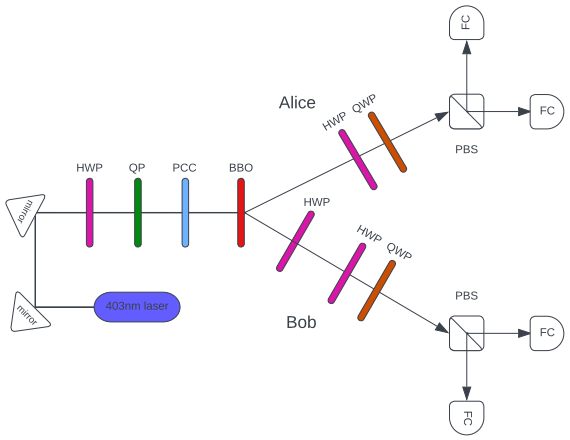


FIG. 2: Experimental setup configuration. HWP is half waveplate, QWP is quarter waveplate, QP is quartz plate, BBO is the BBO crystal. The square with a line on its diagonal represents a polarizing beam splitter."FC" are the fiber optics cables that catch the photons and send them to a single-photon counter. Note actual angle of separate paths, labeled "Alice" and "Bob", from the horizontal is 3°. The pair of HWP and QP before the PBS allow for measurement basis configuration.

We must discuss how to compute measurements in different bases. Since our setup is optical, we use the standard polarization vectors

$$|H\rangle = \begin{bmatrix}1\\0\end{bmatrix}, |D\rangle = \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix}, |R\rangle = \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{i}{\sqrt{2}}\end{bmatrix},$$
$$|V\rangle = \begin{bmatrix}0\\1\end{bmatrix}, |A\rangle = \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{-1}{\sqrt{2}}\end{bmatrix}, |L\rangle = \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{-i}{\sqrt{2}}\end{bmatrix},$$

which we can combine via the tensor product to act on both particles in the pair., e.g.:

$$|HD\rangle = |H\rangle \otimes |D\rangle = \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\0\\0\end{bmatrix}.$$

We define projection via an operator. Let $s_1$ be the first state and $s_2$ be the second for our measurement. Consider $s_1$: for a single qubit, measurement in this basis is

$$m_1 = \mathrm{Tr}\left(\mathrm{s_1 s_1^\dagger \rho}\right),$$

where $s_1 s_1^\dagger$ is the projection operator, and multiplying by the density matrix and taking the trace allows us to

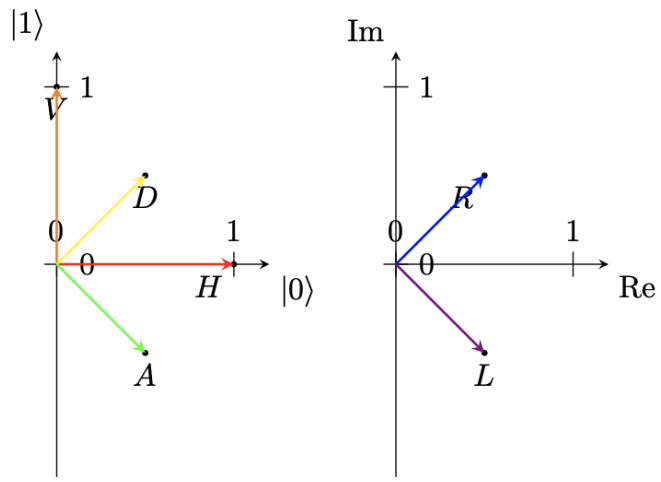


FIG. 3: Illustration of $|H\rangle$, $|V\rangle$, $|D\rangle$, $|A\rangle$ states in $\mathbb{R}^2$ and $|R\rangle$ and $|L\rangle$ in $\mathbb{C}$.

compute the expectation value. We can extend this to 2-qubits by taking the tensor product:

$$m_{1,2} = \mathrm{Tr}\left((\mathrm{s}_1\mathrm{s}_1^\dagger \otimes \mathrm{s}_2\mathrm{s}_2^\dagger)\rho\right).$$

Since our experimental setup, shown in figure 2, can only measure orthogonal states, we can perform a change of basis on the Hilbert space in which $\rho$ lives and essentially choose which two states we want to be orthogonal to.

Essential to this summer's work is the theory of *entanglement witnesses*. [1] describes an entanglement witness as an operator $W$ satisfying:

1. $\mathrm{Tr}\left(\mathrm{W}\rho_{\mathrm{Sep}}^2\right) \geq 0$ for all separable density matrices.
2. $\exists$ at least one entangled state $\rho_{\mathrm{Ent}}$ for which $\mathrm{Tr}\left(\mathrm{W}\rho_{\mathrm{Ent}}^2\right) < 0$.

These $W$ are commonly defined both in terms of a *partial transpose* and directly in terms of *Stokes parameters*. We discuss each in turn. Consider a $4 \times 4$ density matrix for a 2-qubit in terms of $2 \times 2$ block matrices:

$$\rho = \begin{bmatrix} A & B \\ C & D \end{bmatrix}.$$

Applying the partial transpose $^\Gamma$ yields:

$$\rho^\Gamma = \begin{bmatrix} A^\intercal & B^\intercal \\ C^\intercal & D^\intercal \end{bmatrix},$$

where $^\intercal$ is the standard transpose. The partial transpose formulation of entanglement witnesses then takes the form

$$W = |\phi_k\rangle\langle\phi_k|^\Gamma.$$

The Stokes parameters are a set of 16 expectation values for a given density matrix $\rho$ in terms of the Pauli matrices

$$\sigma_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \sigma_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \sigma_2 = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \sigma_3 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

in this form:

$$\langle\sigma_{ij}\rangle = \mathrm{Tr}\left((\sigma_\mathrm{i} \otimes \sigma_\mathrm{j})\rho\right).$$

An example of equivalent definitions of a valid $W$ applied to a state $\rho$ is given by [1]:

$$|\phi_2\rangle = a|\Psi^+\rangle + b|\Psi^-\rangle \Rightarrow W_2 = |\phi_2\rangle\langle\phi_2|^\Gamma$$
$$W_2 = \frac{1}{4}(1 + \sigma_3 \otimes \sigma_3 + (a^2 - b^2)\sigma_1 \otimes \sigma_1 + (a^2 - b^2)\sigma_2 \otimes \sigma_2 + 2ab(\sigma_3 \otimes \sigma_0 + \sigma_0 \otimes \sigma_3).$$

[1] defines a set of 6 witnesses requiring measurements of $\sigma_1 \otimes \sigma_1$, $\sigma_2 \otimes \sigma_2$, $\sigma_3 \otimes \sigma_3$, $\sigma_1 \otimes \sigma_0$, $\sigma_2 \otimes \sigma_0$, $\sigma_3 \otimes \sigma_0$. The conversion from Stokes parameter expectation value to counts is given in Beili Nora's thesis (with a slight correction):

$$\sigma_0 \otimes \sigma_0 = 1$$
$$\sigma_0 \otimes \sigma_1 = \frac{DD - DA + AD - AA}{DD + DA + AD + AA}$$
$$\sigma_0 \otimes \sigma_2 = \frac{RR + LR - RL - LL}{RR + LR + RL + LL}$$
$$\sigma_0 \otimes \sigma_3 = \frac{HH - HV + VH - VV}{HH + HV + VH + VV}$$

$$\sigma_1 \otimes \sigma_0 = \frac{DD + DA - AD - AA}{DD + DA + AD + AA}$$
$$\sigma_1 \otimes \sigma_1 = \frac{DD - DA - AD + AA}{DD + DA + AD + AA}$$
$$\sigma_1 \otimes \sigma_2 = \frac{DR - DL - AR + AL}{DR + DL + AR + AL}$$
$$\sigma_1 \otimes \sigma_3 = \frac{DH - DV - AH + AV}{DH + DV + AH + AV}$$

$$\sigma_2 \otimes \sigma_0 = \frac{RR - LR + RL - LL}{RR + LR + RL + LL}$$
$$\sigma_2 \otimes \sigma_1 = \frac{RD - RA - LD + LA}{RD + RA + LD + LA}$$
$$\sigma_2 \otimes \sigma_2 = \frac{RR - RL - LR + LL}{RR + RL + LR + LL}$$
$$\sigma_2 \otimes \sigma_3 = \frac{RH - RV - LH + LV}{RH + RV + LH + LV}$$

$$\sigma_3 \otimes \sigma_0 = \frac{HH + HV - VH - VV}{HH + HV + VH + VV}$$
$$\sigma_3 \otimes \sigma_1 = \frac{HD - HA - VD + VA}{HD + HA + VD + VA}$$
$$\sigma_3 \otimes \sigma_2 = \frac{HR - HL - VR + VL}{HR + HL + VR + VL}$$
$$\sigma_3 \otimes \sigma_3 = \frac{HH - HV - VH + VV}{HH + HV + VH + VV}$$

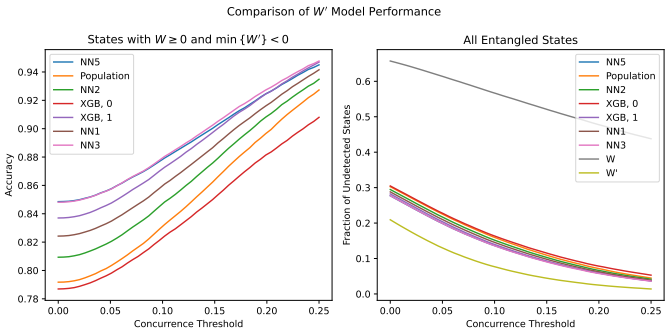


FIG. 4: Performance results of all models, previous and new, investigating during the summer. Strength of entanglement required for witness selection increases in the positive $x$ direction. The figure on the left compares the accuracy on states satisfying $W \geq 0$ and $W'_{\min} < 0$, and the right the fraction of states undetected both by $W$ and all $W'$ in addition to the models on entangled states with no prior $W$ and $W'$ restriction.

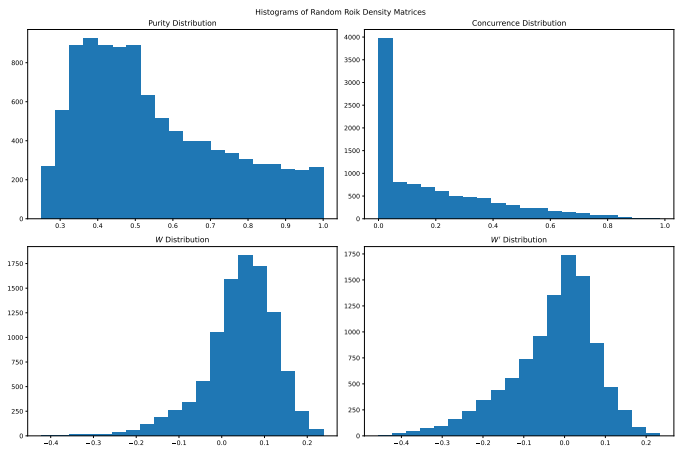


FIG. 5: Histogram for states generated via the Roik method.

[13] extend this selection to 9 more witnesses $\{W'\}$ by including the cross terms $\sigma_1 \otimes \sigma_2$, $\sigma_2 \otimes \sigma_3$, $\sigma_1 \otimes \sigma_3$. Thus they divide their $\{W'\}$ into three triplets, $W'_{t_1}, W'_{t_2}, W'_{t_3}$, based on these sets of local measurements.

To calculate one of these witness values, we can use either the partial transpose method, which is easiest if we have the complete density matrix beforehand, say if we generated the state randomly, or the Stokes parameter method if we have raw counts, e.g., if we are taking data experimentally. We must then find the values of $a, b$ that minimize $W$.

## II. THEORETICAL WORK

One primary goal for this summer was to build on last summer's work, [13], to train a supervised neural network with input projective probabilities corresponding to the measurements necessary for the 6 $\{W\}$ by [1], specifically:

$$HH, HV, VV, DD, DA, DD, RR, RL, LL.$$

The target output is a *multi-hot encoded vector*, in which we calculate $W'_{t_1}, W'_{t_2}, W'_{t_3}$ by taking the minimum of the possible $W'$ within that triplet and then assign a 1 to that index if $W'_{t_x} < 0$ and a 0 otherwise. For example, say a state has the following $\{W\}$

$$0.337, 0.046, 0.079, 0.093, 0.1, 0.067$$

and $\{W'\}$:

$$0.321, 0.046, 0.062, 0.057, 0.092, 0.012, 0.092, 0.06, 0.046.$$

We assign a single $W$ value by taking $W = \min\{W\}$, which yields for this example 0.046—in other words, $W$ does not verify entanglement. Next, we break $W'$ into three triplets and assign corresponding minimums:

$$\{W'_{t_1}\} = 0.321, 0.046, 0.06 \Rightarrow W'_{t_1} = 0.046$$
$$\{W'_{t_2}\} = 0.057, 0.092, 0.012 \Rightarrow W'_{t_2} = 0.012$$
$$\{W'_{t_3}\} = 0.092, 0.06, 0.046 \Rightarrow W'_{t_3} = 0.046.$$

That is, no triplet verifies entanglement either. In fact, the concurrence for this random state is 0. This example segues perfectly into a brief discussion on the importance and complexity of random state generation, which Alec's writeup and poster from this summer dives into in more detail. The essential idea is to sample a 4 dimensional Hilbert Space evenly. However, what we mean by "evenly" is far from straightforward. In this work, we follow the procedure described in the appendix of [2]. The basic algorithm is as follows:

1. Pick 4 random diagonal elements via a random simplex; this sets the eigenvalues of the state. The random simplex that yeilded the best results was to take an initial random guess $M_1$ from 0 to 1. Then, $M_2 = r_1(1 - M_1)$, where $r_1$ is a random number from 0 to 1. We continue in this fashion, $M_n = r_{n-1}(1 - \sum_{i=0}^{n-1} M_i$. We randomize the order of $M_1$ to $M_n$ and then divide by the sum to normalize. (For the record, we call this simplex $S_0$.)

2. Generate 6 random unitaries which combine together to give an overall unitary transformation.

Note that [14] derives the random unitary generation strategy. [15] has a similar method called the "Standard method". See Figure 5 for example distributions with what we have termed the "Roik" method after Roik et al. Ensuring a truly random (or really as good as we can get) distribution of states ensures that when we train and test neural networks, we are using a representative sample of 2-qubits.

Now we describe pertinent machine learning terminology and notes the algorithms and architectures used. Most important to any machine learning project is both what data is used and *how* it used. We have already covered the basis of how we generated data, so we turn to how to use it. It is custom to divide up a dataset into 3 sections: train, validate, and test. The "train" portion is used to actually train the network, which will be explained shortly. "Validation" allows one to examine the performance of a network after altering its architecture, or specifically what are called *hyperparameters*. "Test" is a sacred, untouched portion during the training or tuning process—this will give an unbiased performance metric. It is central that "Test" is never involved in any way in the model development process, or overfitting can occur: this is when a network seems to perform well on the data given, but cannot generalize as well. We experienced overfitting in the beginning of the summer, as we were able to get up 97.1% accuracy on the witness problem with an algorithm called eXtreme Gradient Boosting (XGB) on the 102,000 states generated last spring, but when tested on a new set of data, it vastly underperformed this mark (see Figure 4, XGB-0). Thus we cannot stress the importance of a Testing dataset enough, lest one falls into this trap.

The two primary algorithms considered were neural networks and, as mentioned above, eXtreme Gradient Boosting [16]. Both utilize a loss function, which quantifies the inaccuracy of the prediction relative to the ground truth. Notationally, we define $y$ to be the output of the model at some step, either intermediary or final, and $\hat{y}$ as the ground truth, all for a particular input. In this case, $\hat{y}$ is a *multi-hot encoded vector*. All this means is that our output, which is which $W'_{t_X}$ to choose, is transformed under a map which sends the element at the index of each element to 1 if $W'_{t_X} < 0$ and 0 if not; this is how we, as programmers, can tell the computer which option is acceptable to choose. The model will output a probability vector $y$ based on which class it believes the output to most strongly correspond to, and the task of training, therefore, is to get $y$ as close to $\hat{y}$ as possible. The heart of machine learning is this one idea: we can accomplish this task by minimizing a loss function, that is performing a gradient descent optimization.

For regression problems, the most common loss function is root mean square error

$$L = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}.$$

For, as we are interested in here, classification problems, the function of choice is called cross-entropy [17] and is defined as:

$$L = -\sum_{i=1}^{n} \hat{y}_i \log y_i.$$

For simplicity's sake we can rewrite the sum as a dot product, which can be efficiently calculated using Einstein summation notation:

$$L = -\hat{y} \cdot \log y$$

For a neural network, the structure is represented in Figure 6. The model inputs are a vector of projections into different bases, which pass through a "hidden layer" with some preset number of neurons. Each neuron has an associated weight, and each hidden layer has a bias. Thus the output of the layer can be expressed as:

$$y' = Wy + b,$$

where $W$ is a matrix of weights, $y$ is the previous layer's output, and $b$ is a the bias vector. In order to regularize the probabilities, however, we then pass $y'$ through an *activation function*, most commonly either Softmax or ReLu or Sigmoid. Softmax is defined as

$$A(x_j) = \frac{e^{-x_j}}{\sum_{i=1}^{n} e^{-x_i}},$$

Sigmoid is defined as

$$A(x) = \frac{1}{1 + e^{-x}},$$

ReLu as

$$A(x) = \max(0, x).$$

Since sigmoid outputs are actually independent and thus do not sum to 1, this is the ideal activation function before the network gives its final prediction; that is, for the output layer. For speed we use ReLu on the hidden layers (since in lieu of calculating an exponential, it simply computes the maximum of two values). The process of training minimizes the total loss function, then, as a function of each of the weight and bias parameters. We perform this minimization operation on a set of the data a certain number of times, called *epochs*. We must set a priori the number of hidden layers, the number of neurons per layer, the learning rate (which adjusts how elastically the model responds to the gradient), the batch size (which allows us to train in groups of data as opposed to one at a time), and the optimizer.

We briefly mention a machine learning algorithm called eXtreme Gradient Boosting (XGB), which functions a bit differently from a neural network. The main idea of gradient boosting is the construction of decision trees [18]. We initialize a "base model" which is a constant

$$F_0(\vec{x}) = \min_{\hat{y}} \left( \sum_{i=1}^{n} L(y_i, \hat{y} \right).$$

We add new trees, or "weak learners", iteratively:

$$F_m(\vec{x}) = F_{m-1}(\hat{x}) - \eta \frac{\partial L}{\partial F_{m-1}(\vec{x})},$$

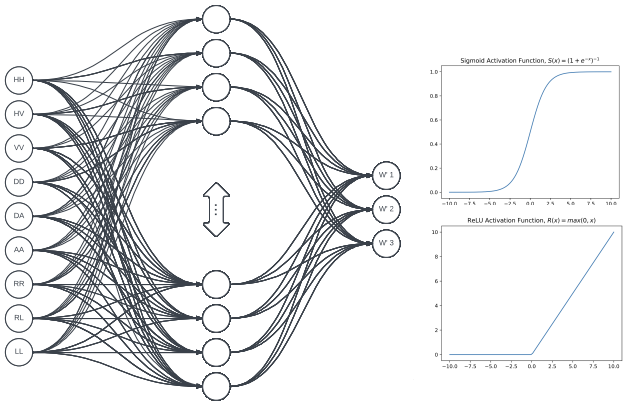


FIG. 6: Structure of neural network, with 9 probability inputs shown, mapping through a "hidden layer" to the final output, which is a probability vector for the three output classes, the witness triplets. Shown to the right are two common activation functions, which regularize the output of each layer.

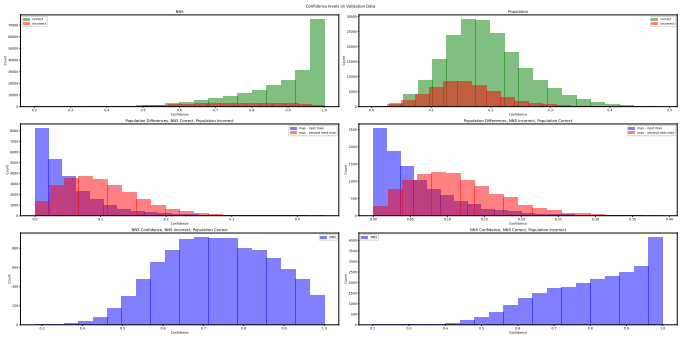

FIG. 7: Comparison of NN5 and the Population model.

where $\eta$ is the learning rate, which controls the level of response of the model to the dervative of the loss function $L$ in the direction of the previous level's output. The final model can be expressed as:

$$F(\vec{x}) = \sum_{m=0}^{M} F_m(\vec{x}).$$

The "eXtreme" bit is a little more complicated, but essentially allows for greater efficiency and control of the model, in particular the parameters *max depth* and *max stopping*, which limit the number of splits.

For the neural network, the process of optimizing these hyperparameters involves initializing and training multiple model configurations on around 1.2 million states and "sweeping" through the different options, in effect creating a grid search—a third party API can be used, such as WandB, which directly interfaces with most Python machine learning packages (including Keras/Tensorflow, which we used in the construction of this model). As aforementioned, the way we measure performance of these models is on a separate "validation" dataset (in this case, a set of 400,000 randomly generated states, matching the 400,000 used for testing), which is completely distinct from the data we use to report final results (although it is generated in the same manner as the training and test data). We considered models with 1, 3, 5, and 10 hidden layers, with between 5 to 500 neurons per layer, learning rates between $10^{-3}$ to 0.5, the optimizers SGD, RMSprop, Adam, Adadelta, Adagrad, and Adamax, and batch sizes between 60 and 2000.

The result of this process are three models called NN5, NN3, and NN1–for 1, 3, and 5 hidden layers–as shown in figure 4 (note: the 10 hidden layer NN is not shown since its performance was acrtually worse than that of NN5). The winning combination ultimately turned out to be 200 neurons per hidden layer and $10^{-3}$ learning rate, the Adam optimizer, a batch size of 2000, the ReLu activation function for all hidden layers, and Sigmoid for the final output layer. We note that NN5 has only a slight increase in performance compared to NN3, but since there is a difference at the 0 concurrence threshold on the validation data, this is the model we decided to present in the SQuInT poster. We used cross-entropy for a loss function as described above. We trained XGB models as well, but the performance did not surpass that of NN3.

We compared these against the results from last year: a 2 hidden layer model called NN2 By Becca Verghese and Laney Goldman [19] with 7 and 5 neurons respectively, with the ReLu and Softmax activation functions, a $10^{-2}$ learning rate, the Adam optimizer, and a custom loss function defined as $\hat{y} \cdot y$, and an analytical approach developed by Eritas Yang [20], which takes as input the combination of the probabilities $0.5 - (P_{HH} + P_{VV})$, $0.5 - (P_{DD} + P_{AA})$, and $0.5 - (P_{RR} + P_{LL})$, and the output is simply the argmax.

An interesting observation, as figure 7 illustrates, is that NN5 and the Population model have non-zero overlap; in fact, only about 88.3% of NN5's correct states the Population model also guesses correctly. More interesting is that if we combine all the correct predictions of the Population model with NN5, the overall accuracy increases to 89% from 85%. Naturally, we tried to come up with a way to determine, when the models disagree, which prediction to defer to: we compared their confidence levels at the extrema (do not use NN5 prediction if confidence, the largest probability output, was $< 0.55$, which would signify random guessing; use Population if NN5 tries to predict the least probable triplet from the Population perspective); we trained meta-models, both neural networks and XGBoost (described below) on the raw probability and difference outputs of NN5 and Population pre-label assignment; we trained neural networks on the six probability inputs $P_{HH}, P_{VV}, P_{DD}, P_{AA}, P_{RR}, P_{LL}$ and were able to achieve performance identical to the Population model and tried again to train meta-models and create a model ensemble that averaged the probability outputs of this neural-

network version of Population with NN5; we even added the inputs to the Population model and retrained NN5 from scratch—none of these attempts increased model performance beyond 85%. When trying to retrain NN5 based on the states it predicted incorrectly in the validation dataset, performance actually sharply decreased. Thus, it is likely that 85% is around the statistical limit of the usefulness of the input data, since any correlations present in the incorrect states actually corrupted the learned correlations in the other states; that is, while we might suspect the existence of correlations that should in theory allow for a 4% performance increase when combining NN5 with Population, it appears more likely that the difference is random.

On a final note, we were interested in assessing the performance of our own custom model on the problem investigated by [2]: namely, predicting whether a state is entangled or not using as inputs groups of conditional projective measurements defined by

$$P_{XY} = \frac{\mathrm{Tr}[(\hat{\rho}_I \otimes \hat{\rho}_{II})(\Pi_X \otimes \Pi_{\mathrm{Bell}} \otimes \Pi_Y)]}{\mathrm{Tr}[(\hat{\rho}_I \otimes \hat{\rho}_{II})(\Pi_X \otimes I_4 \otimes \Pi_Y)]},$$

where $\Pi_X, \Pi_Y$ are projections onto single qubit spaces and $\Pi_{\mathrm{Bell}}$ is a projection onto the singlet Bell state (i.e., $|\Psi^-\rangle$; moreover, $\hat{\rho}_{II}$ is derived from the original density matrix $\hat{\rho}_I$ by "exchanging subsystems"–that is, by swapping the middle two rows and columns.

In training a variety of different model configurations on 4 million states, on a 400,000 state test we are able to independently match Roik et al.'s results to within 1%, which lends strong evidence to the identification of their success metrics with the limit of the statistical usefulness of the input probability groups.

## III. EXPERIMENTAL WORK

On the experimental side, we wanted to demonstrate the efficacy of the $\{W'\}$ group of witnesses. [20] suggested states of the form

$$E_0(\eta, \chi) = \cos\eta|\Psi^+\rangle + \sin\eta e^{i\chi}|\Psi^-\rangle,$$

where $|\Psi^\pm\rangle = |HV\rangle \pm |VH\rangle$ are Bell states. Richard, in his writeup, describes an algorithm with adaptive feedback from the setup (see Figure 2) to create states of the form

$$\cos\eta|HV\rangle + \sin\eta e^{i\chi}|VH\rangle,$$

along with the $\{W\}$ and $\{W'\}$ values for the experimental states, fixing $\eta = 30°, 45°$, and $60°$ and sweeping $\chi \in [0, 90°]$.

I was also interested in taking experimental data, but with a generalized algorithm for any state our setup could create without having to rely on results from the setup in the moment, rather solely an initial calibration. For a more detailed explanation, see [21]. The basis of this

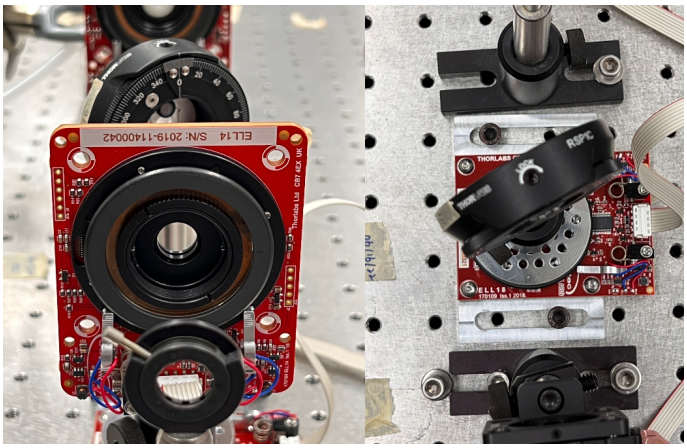

FIG. 8: Image from experimental setup. *(a–left)* A half waveplate front on, which rotates in the vertical plane (a quarter waveplate looks identical). *(b–right)* A quartz plate top down, which rotates in the horizontal plane. The red is the Thor Labs PCB that allows a computer to issue precise movement instructions for angle settings.

algorithm I developed, called InstaQ, is to use Jones calculus to describe the polarization state of light; we define our experimental setup mathematically and then determine the optimal parameters, which correspond to physical angle settings of the optical components, via gradient descent.

Figure 2 presents the main summary of these efforts. We used the Lynn Lab's quantum optics setup, illustrated in Figure 2, in which a laser produces horizontally polarized photons $|H\rangle$ at 403nm wavelength, and then a half waveplate, which shifts incoming polarization states by $\pi$ radians allows a superposition of $\cos(2\theta)^2|H\rangle + \sin(2\theta)|V\rangle$ for some angle $\theta$. A quartz plate allows a relative phase: $\cos(2\theta)^2|H\rangle + e^{i\phi}\sin(2\theta)|V\rangle$. A precompensation crystal improves the quality of the states by making them less mixed (since even though we are writing pure states, the operations have a non-unity fidelity due to noise. A pair of barium borate (BBO) crystals promote a process called spontaneous parametric down conversion (SPDC) [4] in which one high energy photon becomes two lower energy photons as shown in Figure 1; specifically, $|H\rangle \to |VV\rangle$ and $|V\rangle \to |HH\rangle$. Because of this "optical meiosis", the resultant pairs of photons are entangled with respect to their polarization (which is the property we exploit), momentum, and energy. Cones of these photons propagate down the optics table, and a further half waveplate along one of the paths, called "Bob" in contrast to "Alice", allows us to shift the polarization state of one of the photons in the pair independent from the other. We then have a half waveplate and a quarter waveplate, which shifts the polarization of incoming light by $\frac{\pi}{2}$ radians, along both Alice and Bob's paths, which enable us to rotate the created state into whatever different bases we wish to measure in, as ex-

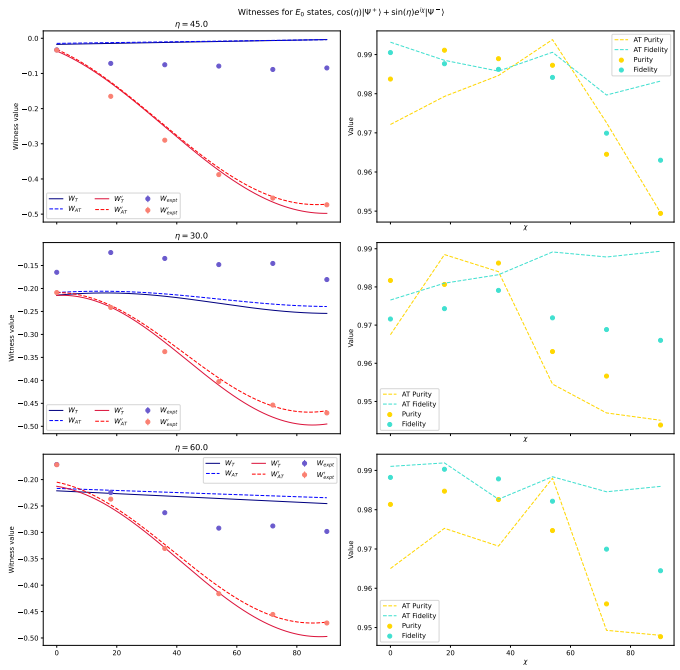


FIG. 9: Experimental results on states obtained with my algorithm. Data points represent actual measurements, solid curves represent theoretical predictions, and dashed curves adjusted theoretical predictions. Thus, ideally the curve showing the fidelity of the experimental state to the adjusted theoretical is as close to 1 as possible, and the curve for the purity of the adjusted theoretical state follows the experimental datapoints.

plained in the introduction and in Figure 3.

My algorithm works by creating a theoretical representation of the setup described in the language of Jordan calculus, which describes the effect of the optical components listed above as matrices [22], inspired by work that proved that a half waveplate and two quarter waveplates was sufficient to generate all of SU2 [23]. From Figure 9, we notice that for each of $\eta = 30°, 45°, 60°$, the $W$ value was quite patently off from theoretical and adjusted theoretical, although $W'$ was closer. We attempted two modes of corrections based on the following arguments: (1) there is phase smearing and non-zero $|HH\rangle$ and $|VV\rangle$ pairs leftover, and (2) assuming phase smearing, coherence, and extra $|HV\rangle$ and $|VH\rangle$ noise. Mathematically, that looks like the following. What we call "Model 3" is:

$$\begin{aligned}\rho_{AT} = (1-e_1-e_2)[p|\psi\rangle\langle\psi| + (1-p)(a|HV\rangle\langle HV| \\ + b|VH\rangle\langle VH|)] + e_1|HH\rangle\langle HH| \\ + e_2|VV\rangle\langle VV|,\end{aligned}$$

where $p$ is the purity of the experimental state, $|\psi\rangle$ is the theoretical target state,

$$a = \left|\frac{\cos\eta + e^{i\chi}\sin\eta}{\sqrt{2}}\right|^2 = \frac{1+\cos\chi\sin 2\eta}{2},$$

and

$$b = 1 - a.$$

We determine $e_1$ and $e_2$ via gradient descent optimization. The specific loss function we used was:

$$L = \frac{1}{\sqrt{\mathcal{F}_{AT}}} + |\gamma - \gamma_{AT}|,$$

where $\mathcal{F}_{AT}$ is the fidelity of the resulting adjusted theoretical state $\rho_{AT}$ to the experimental state $\rho_{Expt}$, $\gamma$ is the purity of the experimental state, and $\gamma_{AT}$ is the purity of the adjusted theoretical state. This loss function was chosen to not let the fidelity overdominate the purity. The results of this process are recorded in Table I. We also consider the model with coherence and extra $|HV\rangle$ and $|VH\rangle$, which we call "Model 1":

$$\begin{aligned}\rho_{AT} = (1-p)((1-e)(a|HV\rangle\langle HV| + b|VH\rangle\langle VH|) \\ + e(|HH\rangle\langle HH|)) + p((1-e)|\psi\rangle\langle\psi| + e|\psi_S\rangle\langle\psi_S|),\end{aligned}$$

where $e$ is the adjustable parameter determined, like above, by gradient descent, and $|\psi_S\rangle$ is obtained by swapping $|HV\rangle \to |HH\rangle$ and $|VH\rangle \to |VV\rangle$. Table II summarizes the results of the correction. Important to note is that $|\psi\rangle$ was not just $E_0(\eta,\chi)$ for the target $\eta$ and $\chi$—since we realized the UVHWP was offset from true home by $-1.029°$, we added back this correction factor and determined the correct $\eta, \chi$ using the Jones matrix formalism and gradient descent, with the loss function

$$L = 1 - \sqrt{\mathcal{F}_{AT}}.$$

These updated angle parameters are used in the calculation of $\rho_{AT}$; however, when plotting the results in Figure 9, the original $\eta, \chi$ are used.

As one may note, most of the corrections are in fact quite small yet allow for a fidelity from the corrected theoretical state to the experimental state around or above 0.99, as illustrated in the model summary tables as mentioned above. In direct comparison as shown in Figure 10, it is patently clear that the corrections do not yield witness values that even somewhat match the experimental results—except for $W'_{t_1}$. Figure 11 allows us to get a the root of the problem: the differences in the density matrices themselves. We see that the adjustment, here shown using Model 1, does correct the off-diagonal components, but fails to correct the diagonal HV and VH components; the magnitude of the phase remains uncorrected. The Stokes parameters are quite different—XX and YY, with the adjusted theory, have moved the wrong direction (to a total of about 0.13 off), whereas XY has moved closer to target (to a total of about 0.11 off). This difference explains why the adjustments have failed to correct the witness values. However, it is strange that the state in Figure 11, $\eta = 60°, \chi = 90°$, has a fidelity of 0.986 and yet its Stokes parameters are so different. The best state that we created with the InstaQ algorithm ($\eta = 60°, \chi = 18°$) had a fidelity of 0.9902, for which

TABLE I: Results for Model 3 correction[a]. $\mathcal{F}$ is the fidelity of the experimental state to the theoretical (with the UVHWP corrected), $\mathcal{F}_{AT}$ is the fidelity of the experimental to the adjusted theoretical. $\gamma$ is purity of the experimental state, $\gamma_{AT}$ is the purity of the adjusted theoretical state. $e_1$ and $e_2$ are the parameters determined by gradient descent in the model.

| $\eta$ | $\chi$ | $\mathcal{F}$ | $\mathcal{F}_{AT}$ | $\gamma$ | $\gamma_{AT}$ | $e_1$ | $e_2$ |
|---|---|---|---|---|---|---|---|
| 45.0 | 0.0 | 0.991 | 0.995 | 0.984 | 0.984 | 0.003 | 0.004 |
| 45.0 | 18.0 | 0.988 | 0.993 | 0.991 | 0.991 | 0.001 | 0.001 |
| 45.0 | 36.0 | 0.986 | 0.986 | 0.989 | 0.986 | 0.001 | 0.001 |
| 45.0 | 54.0 | 0.984 | 0.986 | 0.987 | 0.979 | 0.0 | 0.0 |
| 45.0 | 72.0 | 0.97 | 0.978 | 0.965 | 0.954 | 0.0 | 0.0 |
| 45.0 | 90.0 | 0.963 | 0.983 | 0.949 | 0.949 | 0.0 | 0.0 |
| 30.0 | 0.0 | 0.972 | 0.978 | 0.982 | 0.972 | 0.0 | 0.002 |
| 30.0 | 18.0 | 0.974 | 0.978 | 0.981 | 0.976 | 0.0 | 0.0 |
| 30.0 | 36.0 | 0.979 | 0.984 | 0.986 | 0.985 | 0.0 | 0.0 |
| 30.0 | 54.0 | 0.972 | 0.99 | 0.963 | 0.963 | 0.001 | 0.002 |
| 30.0 | 72.0 | 0.969 | 0.989 | 0.957 | 0.957 | 0.001 | 0.001 |
| 30.0 | 90.0 | 0.966 | 0.99 | 0.944 | 0.944 | 0.0 | 0.001 |
| 60.0 | 0.0 | 0.988 | 0.995 | 0.981 | 0.981 | 0.003 | 0.002 |
| 60.0 | 18.0 | 0.99 | 0.995 | 0.985 | 0.985 | 0.0 | 0.002 |
| 60.0 | 36.0 | 0.988 | 0.986 | 0.983 | 0.982 | 0.0 | 0.0 |
| 60.0 | 54.0 | 0.982 | 0.985 | 0.975 | 0.968 | 0.0 | 0.0 |
| 60.0 | 72.0 | 0.97 | 0.985 | 0.956 | 0.949 | 0.0 | 0.0 |
| 60.0 | 90.0 | 0.964 | 0.986 | 0.948 | 0.948 | 0.0 | 0.0 |

[a] In the file `oscar/machine_learning/process_expt.py`, this is called Model 3 because it's simpler for the gradient descent algorithm to determine all 3 parameters with the condition they sum to 1.

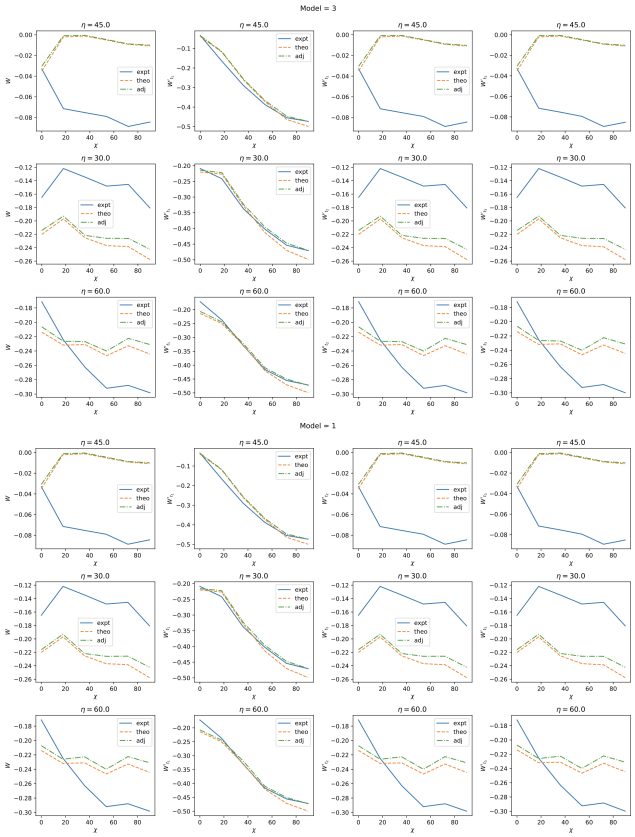


FIG. 10: Comparison of Models 3 and 1.

TABLE II: Results for Model 1 correction. $\mathcal{F}$ is the fidelity of the experimental state to the theoretical (with the UVHWP corrected), $\mathcal{F}_{AT}$ is the fidelity of the experimental to the adjusted theoretical. $\gamma$ is purity of the experimental state, $\gamma_{AT}$ is the purity of the adjusted theoretical state. $e$ is the parameter determined by gradient descent in the model.

| $\eta$ | $\chi$ | $\mathcal{F}$ | $\mathcal{F}_{AT}$ | $\gamma$ | $\gamma_{AT}$ | $e$ |
|---|---|---|---|---|---|---|
| 45.0 | 0.0 | 0.991 | 0.992 | 0.984 | 0.984 | 0.007 |
| 45.0 | 18.0 | 0.988 | 0.991 | 0.991 | 0.991 | 0.003 |
| 45.0 | 36.0 | 0.986 | 0.988 | 0.989 | 0.985 | 0.002 |
| 45.0 | 54.0 | 0.984 | 0.987 | 0.987 | 0.979 | 0.001 |
| 45.0 | 72.0 | 0.97 | 0.979 | 0.965 | 0.954 | 0.0 |
| 45.0 | 90.0 | 0.963 | 0.984 | 0.949 | 0.949 | 0.001 |
| 30.0 | 0.0 | 0.972 | 0.973 | 0.982 | 0.977 | 0.0 |
| 30.0 | 18.0 | 0.974 | 0.976 | 0.981 | 0.977 | 0.0 |
| 30.0 | 36.0 | 0.979 | 0.983 | 0.986 | 0.985 | 0.001 |
| 30.0 | 54.0 | 0.972 | 0.99 | 0.963 | 0.963 | 0.004 |
| 30.0 | 72.0 | 0.969 | 0.99 | 0.957 | 0.957 | 0.005 |
| 30.0 | 90.0 | 0.966 | 0.99 | 0.944 | 0.944 | 0.002 |
| 60.0 | 0.0 | 0.988 | 0.994 | 0.981 | 0.981 | 0.005 |
| 60.0 | 18.0 | 0.99 | 0.994 | 0.985 | 0.982 | 0.005 |
| 60.0 | 36.0 | 0.988 | 0.989 | 0.983 | 0.968 | 0.009 |
| 60.0 | 54.0 | 0.982 | 0.986 | 0.975 | 0.968 | 0.001 |
| 60.0 | 72.0 | 0.97 | 0.985 | 0.956 | 0.949 | 0.0 |
| 60.0 | 90.0 | 0.964 | 0.987 | 0.948 | 0.948 | 0.001 |

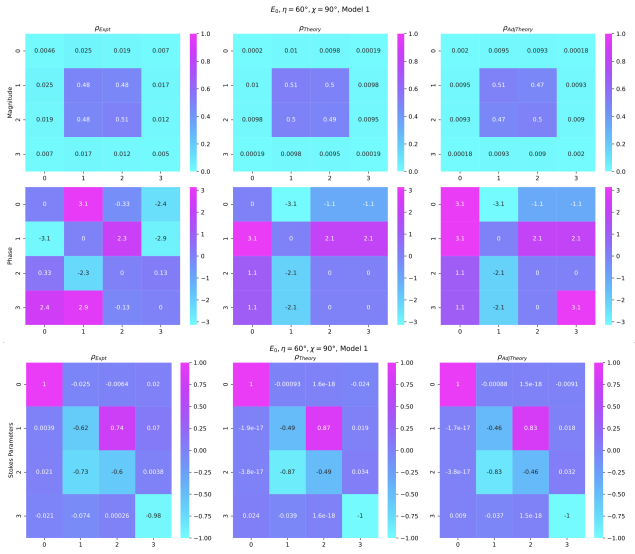


FIG. 11: Comparison of the magnitude-angle values of experimental, theoretical, and adjusted theoretical states, as well as the Stokes parameters.

the witness values nicely agree. Thus, it is possible to create and measure states of interest through this algorithm, but the sensitivity of the Stokes parameters illustrates the experimental difficulty of this kind of approach, which relies on approximating the state through the fidelity. Moreover, a lack of intepretability with this machine learning approach makes understanding experimental sources of noise difficult. Unfortunately, the essence of that section is we find the Stokes parameters very sensitive to differences in magnitude and phase of entries in the density matrices despite fidelities of the experimental to adjusted theoretical states being around 0.99.

For clarity, we ran the same code on data that Richard gathered using his algorithm. Moreover, we were able to successfully employ a correction just accounting for phase smearing:

$$\rho_{AT} = p|\psi\rangle\langle\psi| + (1-p)(a|HV\rangle\langle HV| + b|VH\rangle\langle VH|).$$

Therefore, the discrepancy with the states determined via my model has potential for better understanding fidelity as a metric of similarity.

## IV. CONCLUSION

This summer, we made good progress on improving the results of our neural network to predict the optimal set of entanglement witnesses defined by [13] as well as with our ability to create and measure experimental states. On the theoretical side, we explored state creation and I gained an additional 4% on states for which the minimal witness of the set defined by [1] did not verify entanglement but at least one of the witnesses defined by [13] would based on projective local measurements. Based on further tests with neural network configurations, I am fairly confident we have reached the limit of the statistical relevance of the input parameters. We will be presenting these results at SQuInT 2023. Moreover, we investigated a problem by [2] to classify whether a state was entangled or separable based on a set of conditional probabilites they define, and despite having a completely distinct model architecture, based on the data we generate we are able to match their results to within a tenth of a percent. On the experimental side, though not explicitly covered in this writeup, the code for interacting with the experimental setup was completely revamped, headed by Alec Roberson. Richard Zheng's work focused on an analytical algorithm for generating states of the form considered in section III, which we will be showcasing at SQuInT. I attempted also to create my own algorithm based on a gradient descent optimization using Jones matrix formalism to determine the settings of our optical equipment. However, the calculated witness values were in disagreement with theoretical predictions, even after attempted correction despite around 0.99 fidelity—in contrast to Richard's results. Therefore, more work to understand sources of error contributing to this discrepancy is needed.

## V. ACKNOWLEDGEMENTS

We thank Theresa Lynn for a great deal of advice and support throughout this summer, as well as the entire physics community for making the summer a fantastic time.

[1] A. Riccardi, D. Chruściński, and C. Macchiavello, Physical Review A **101**, 062319 (2020).
[2] J. Roik, K. Bartkiewicz, A. Černoch, and K. Lemr, Physical Review Applied **15**, 054006 (2021), publisher: American Physical Society.
[3] O. Scholin, (2023).
[4] C. Couteau, Contemporary Physics **59**, 291 (2018), arXiv:1809.00127 [physics, physics:quant-ph].
[5] Note that Roberson has spent a good deal of time thinking about randomized states, so see his writeup and poster for more details.
[6] By local, we mean non-conditional or entangling—one can entangle previously unentangled states through such a scheme.
[7] M. A. Nielsen and I. L. Chuang, English *Quantum computation and quantum information*, 10th ed. (Cambridge University Press, Cambridge, 2000).
[8] As you will do in Optics Lab!
[9] J. S. Bell, Physics Physique Fizika **1**, 195 (1964), publisher: APS.
[10] J. Yin, Y. Cao, Y.-H. Li, S.-K. Liao, L. Zhang, J.-G. Ren, W.-Q. Cai, W.-Y. Liu, B. Li, H. Dai, G.-B. Li, Q.-M. Lu, Y.-H. Gong, Y. Xu, S.-L. Li, F.-Z. Li, Y.-Y. Yin, Z.-Q. Jiang, M. Li, J.-J. Jia, G. Ren, D. He, Y.-L. Zhou, X.-X. Zhang, N. Wang, X. Chang, Z.-C. Zhu, N.-L. Liu, Y.-A. Chen, C.-Y. Lu, R. Shu, C.-Z. Peng, J.-Y. Wang, and J.-W. Pan, Science **356**, 1140 (2017), publisher: American Association for the Advancement of Science.
[11] See chapter 9 of Lynn's quantum information textbook for a great introduction to the density matrix formalism of quantum mechanics.
[12] W. K. Wootters, Physical Review Letters **80**, 2245 (1998).
[13] E. Yang, B. Verghese, and B. Hartley, Unpublished (2022).
[14] C.-K. Li, R. Roberts, and X. Yin, International Journal of Quantum Information **11**, 1350015 (2013), publisher: World Scientific Publishing Co.
[15] J. Maziero, Brazilian Journal of Physics **45**, 575 (2015).
[16] K. Team, en"Keras documentation: Character-level text generation with LSTM,".
[17] A. Mao, M. Mohri, and Y. Zhong, arXiv.org (2023).
[18] XGBoost, "Introduction to Boosted Trees — xgboost 1.7.6 documentation,".
[19] B. Verghese and L. Goldman, Unpublished (2023).
[20] E. Yang, Unpublished (2023).

[21] O. Scholin, Unpublished (2023).
[22] E. Hecht, en*Optics* (Pearson Education, Incorporated, 2017) google-Books-ID: ZarLoQEACAAJ.
[23] R. Simon and N. Mukunda, Physics Letters A **143**, 165 (1990).