Learning Entanglement Types

Luke Oeding (Auburn University, USA) & Hamza Jaffali (Colibritd, Paris, France)

Learning Algebraic Models of Entanglement [Jaffali-Oeding, Quant. Inf. Proc. 2020]

Determine Membership

"Given" an algebraic variety $X \subset \mathbb{P}^n$ and point $[z] \in \mathbb{P}^n$, determine if $[z] \in X$.

- For example, given the $n \times n$ matrix multiplication tensor (algorithm) does it lie on the border-rank R variety for some $R = O(n^2)$?
- Is the $m \times m$ permanent polynomial p_m a point of the orbit (closure) of the $n \times n$ determinant polynomial for $n = \text{poly}(m)$?
- What entanglement type is a sample quantum state?

Determine Membership

"Given" an algebraic variety $X \subset \mathbb{P}^n$ and point $[z] \in \mathbb{P}^n$, determine if $[z] \in X$.

- If $X = \mathcal{V}(f_1, \ldots, f_t)$ easy: just check if $f_i(z) = 0$ for all $i = 1..t$.
- If X is *parametrized* by a rational map $\varphi \colon \mathbb{P}^m \dashrightarrow \mathbb{P}^n$, then for some open $U \subset \mathbb{P}^m$,

$$X = \overline{\varphi(U)}$$

Can try to learn the ideal $I(X)$: the polynomial equations that define X:

- By symbolic elimination? *Ideals, Varieties, and Algorithms* [Cox-Little-Oshea (1989)]
- By sampling? Can generate as many samples of X as we want via φ. Try to guess the polynomial generators $f_i \in \mathbb{Q}[x_0, \ldots, x_n]$ from noisy samples. *Learning Algebraic varieties from samples*, [Brieding-Kališnik-Sturmfels-Weinstein (2018)]
- By interpolation? (with symmetry). Many examples, but shameless plugs:
  - ⋆ *Equations for the fifth secant variety of Segre products of projective spaces*, [Oeding-Sam (2016)]
  - ⋆ and *Hyperdeterminants from E8*, [Holweck-Oeding (2021)]

Complexity Issues

- Elimination theory: If $[\varphi(x)] = [\varphi_0(x_0, \dots x_m) : \cdots : \varphi_n(x_0, \dots, x_m)]$ form $z_i - \varphi_i(x_0, \dots x_m)$, and try to eliminate the x_i's using Gröbner basis algorithms - worst case complexity is doubly exponential in n.
- Interpolation: If $X \subset \mathbb{P}^n$ is a hypersurface of degree d, then it is defined by a polynomial F with potentially $N = \binom{n+d}{d}$ monomials, so N evaluations and linear algebra with an $N \times N$ matrix can find all coefficients, complexity is roughly $O(N^3)$.

Might reduce complexity by exploiting symmetry. What other ways?

Entanglement types for 3 qubits

given data $T = \sum_I T_I \left|I\right\rangle$ determine which type up to symmetry.

$$\frac{1}{\sqrt{2}}\left(|000\rangle + |111\rangle\right)$$

$$\downarrow$$

$$\frac{1}{\sqrt{3}}\left(|001\rangle + |010\rangle + |100\rangle\right)$$

$$\swarrow \qquad \downarrow \qquad \searrow$$

$$\frac{1}{\sqrt{2}}\left(|001\rangle + |010\rangle\right) \qquad \frac{1}{\sqrt{2}}\left(|001\rangle + |100\rangle\right) \qquad \frac{1}{\sqrt{2}}\left(|010\rangle + |100\rangle\right)$$

$$\searrow \qquad \downarrow \qquad \swarrow$$

$$|000\rangle$$

$$\downarrow$$

$$0$$

- Three qubits can be entangled in two inequivalent ways [Dür-Vidal-Cirac (2000)]
- Discriminants, resultants and multidimensional determinants, [Gelfand-Kapranov-Zelevinsky (1994)]

Pure States Formalism for QI and Algebraic Geometry AG

QI studies	**AG studies**
Hilbert space $\mathcal{H} = \mathbb{C}^{n+1}$ unit vectors are states $\|\varphi\rangle$ with $\langle\varphi, \varphi\rangle = 1$.	projective space $\mathbb{P}^n$ (elements are lines thru 0). $[x] = [\lambda x]$ for every $\lambda \in \mathbb{C}$.
A **measurement basis** for $\mathcal{H}$ $\|0\rangle, \dots, \|n\rangle$	a basis of variables for $\mathbb{C}^{n+1}$ $x_0, \dots, x_n$
orthonormal basis	[*sometimes ignores this*]
complex inner product $\langle , \rangle$.	[*hates complex conjugation*]
Measurement basis for $\mathcal{H}^{\otimes m}$ $\|0\cdots 0\rangle$, $\|0\cdots 01\rangle$, ..., $\|n\cdots n\rangle$	tensor product basis $\{x_{i_1} \otimes \cdots \otimes x_{i_m} \mid I = (i_1, \dots, i_m) \in \{0, \dots n\}^m\}$
a **state** of an m-particle ensemble $\Phi \in \mathcal{H}^{\otimes m}$ with $\langle\Phi, \Phi\rangle = 1$	an element of a tensor product space $[x] \in \mathbb{P}^N$ and $N = (n+1)^m - 1$ with $[x] = [\lambda x]$ for every $\lambda \in \mathbb{C}$.
Separable State $\|\varphi\rangle_1 \|\varphi\rangle_2 \cdots \|\varphi\rangle_m$	tensor product $[a_1 \otimes \cdots \otimes a_m]$, with $a_i \in \mathcal{H}$.
.	.

Algebraic Stats (AS) interpretations:

AS studies	**AG studies**
Probability Simplex $\Delta \in \mathbb{R}^{n+1}$ Multi-modal contingency tables $P = (P_{00\cdots 0} \cdots P_{11\cdots 1})$	projective space $\mathbb{P}^n$ tensors $\{x_{i_1} \otimes \cdots \otimes x_{i_m} \mid I = (i_1, \ldots, i_m) \in \{0, \ldots n\}^m\}$
Independence Model P with $\det_{i,j}(P_(i, j, * \cdots *)) = 0$	**Segre variety** $\mathrm{Seg}(\mathbb{P}^m \times \cdots \mathbb{P}^m)$
Hidden Markov Model	**Secant variety**
Phylogenetic Invariants	**Defining Equations**

Entangled and Degenerate States

Definition

The state of an m-particle ensemble $\Phi \in \mathcal{H}^{\otimes m}$ is **unentangled** if $\Phi = \varphi_1 \otimes \cdots \otimes \varphi_n$ for some $\varphi_i \in \mathcal{H}$.
Unentangled ensambles satisfy **independence**:
$P(\Phi_i = a \mid \Phi_j = b) = P(\Phi_i = a)P(\Phi_j = b)$.

Example (Entangled States)

Take $\Phi = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Measure one particle at a time (separate experiments):
The probabilities are: $P(\varphi_1 = |0\rangle) = P(\varphi_1 = |1\rangle) = 50\%$,
$P(\varphi_2 = |0\rangle) = P(\varphi_2 = |1\rangle) = 50\%$.
But, the conditional probabilities are: $P(\varphi_2 = |0\rangle \mid \varphi_1 = |0\rangle) = 100\%$ and
$P(\varphi_2 = |0\rangle \mid \varphi_1 = |1\rangle) = 0\%$.
The independence condition $P(\varphi_2 = b \mid \varphi_1 = a) = P(\varphi_2 = b)P(\varphi_1 = a)$ fails.

Example (Rank 1 Tensors)

For AG the set of rank 1 tensors is a projective variety called the Segre variety:

$$\mathrm{Seg}(\mathbb{P}^n \times \cdots \times \mathbb{P}^n) = \{[a \otimes b \otimes \cdots \otimes c] \mid [a], [b], \ldots [c] \in \mathbb{P}^n\} \subset \mathbb{P}^N, \quad \text{with } N = (n+1)^m - 1.$$

QI calls these elements **separable states**, **product states**, or **unentangled**.

AS, QI and AG all know that the Segre variety (set of product states) is defined by all 2×2 minors of flattenings of tensors, that this variety is actually a smooth manifold, it is homogeneous for the action of $SL(n+1)^{\times m}$, its cone has dimension $m \cdot n + 1$ etc.

Higher secant varieties are quite mysterious.

Hyperdeterminants detect highly entangled states

Our AG mantra is to use algebraic invariants to separate entanglement types.

AG likes to have polynomial expressions, also OK with evaluation methods (like SLPs).

The determinant is a polynomial that measures failure to be low rank (and hence failure to be entangled), but only for 2-particle systems. Determinants have complicated expressions (n! terms), but easy evaluations (via Gaussian Elimination).

Hyperdetermiants (and other invariants) play this role for multi-particle systems and detect highly entangled states. Even more complicated than determinants, but maybe also have easy evaluations?

Determinant: Dual to Rank 1 Matrices (unentangled states)

A matrix has rank 1 if, up to Gaussian elimination it is of the form:

$$A = \begin{pmatrix} * & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \dots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix} \qquad = \textit{column} \cdot \textit{row}$$

Dually, a matrix is singular if up to Gaussian elimination it is of the form:

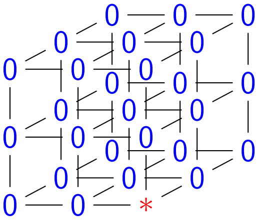
$$A^\vee = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 \\ 0 & * & * & \dots & * \\ \vdots & \vdots & \ddots & \dots & \vdots \\ 0 & * & * & \dots & * \end{pmatrix} \qquad = \textit{annihilates row and column}$$

The set of all singular matrices is defined by the vanishing of a determinant.

GKZ Hyperdeterminant: (Unentangled States)

Parametrized by the following geometric construction:

Rank 1 | Tangent | Hyperplanes $\perp$



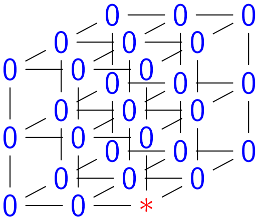
$$\pitchfork$$

$$\mathrm{Seg}(\mathbb{P}^n \times \mathbb{P}^n \times \mathbb{P}^n)$$

GKZ Hyperdeterminant: (Unentangled States)

Parametrized by the following geometric construction:

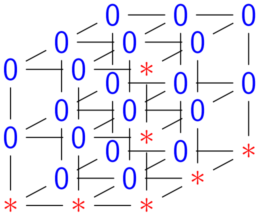
Rank 1



$\pitchfork$

$\mathrm{Seg}(\mathbb{P}^n \times \mathbb{P}^n \times \mathbb{P}^n)$

Tangent



$\pitchfork$

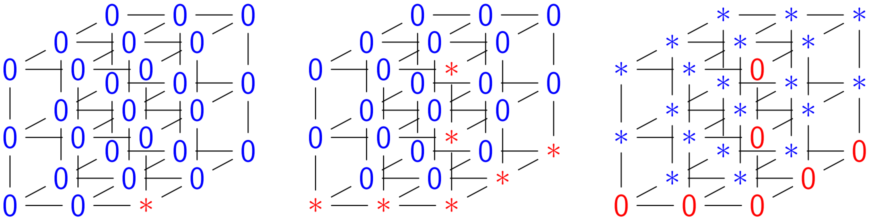
$T_p\,\mathrm{Seg}(\mathbb{P}^n \times \mathbb{P}^n \times \mathbb{P}^n)$

Hyperplanes $\perp$

GKZ Hyperdeterminant: (Unentangled States)

Parametrized by the following geometric construction:

Rank 1 | Tangent | Hyperplanes $\perp$



$\pitchfork$ | $\pitchfork$ | $\pitchfork$

$\mathrm{Seg}(\mathbb{P}^n \times \mathbb{P}^n \times \mathbb{P}^n)$ | $T_p\,\mathrm{Seg}(\mathbb{P}^n \times \mathbb{P}^n \times \mathbb{P}^n)$ | $\mathrm{Seg}(\mathbb{P}^n \times \mathbb{P}^n \times \mathbb{P}^n)^\vee = \mathcal{V}(Det)$

Random arrays with zeros and change coordinates – get points on hyperdeterminant.

Hyperdeterminants are Complicated

- [Cayley (1850s)]: The $2 \times 2 \times 2$ hyperdeterminant is a degree 4 polynomial on 8 variables with 12 terms.
- [Huggins-Sturmfels-Yu-Yuster (2008)]: The $2 \times 2 \times 2 \times 2$ hyperdeterminant is a polynomial of degree 24 on 16 variables with $2,894,276$ terms (about .012% of the possible $N = 2.5 * 10^{10}$ monomials.
- [GKZ] The $2 \times 2 \times 2 \times 2 \times 2$ hyperdeterminant is a polynomial of degree 128 on 32 variables with ? terms. Here $N = 3.7 * 10^{33}$

This linear algebra seems completely out of reach at the moment.

A new idea: use ML to study tensors (not the reverse)

Build an Artificial Neural Network (ANNs) classifier that takes the coordinates of a tensor (quantum state / contingency table) and outputs its entanglement class.

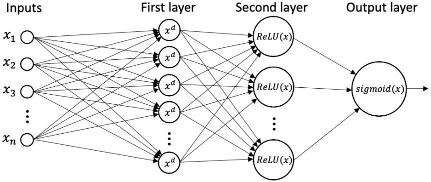


FIGURE 8. Representation of a hybrid network for learning a homogeneous polynomial equation of degree d in n variables and coefficients in $\mathbb{R}$.

ML Basics

Goal: Given classified training data (x_i, b_i) with $[x_i] \in \mathbb{P}^n$ and $b_i \in \{0,1\}$ select a function $F \in \mathcal{F} = \{F_\lambda \mid \lambda \in \mathbb{R}^N\}$ so that $F(x_i) = b_i$ for all i.

Model selection: Choose the family $\mathcal{F}$ depending on anticipated data features.

Training: Solve optimization problem:

$$F = \min_\lambda |F_\lambda(x_i) - b_i|$$

Use, say, 90% of data for training, and 10% for validation. May require many epochs.

Classification: Set a threshold, say $\theta = 0.5$, and declare:

$$F(x) = \begin{cases} 0 & \text{if } |F(x)| < \theta, \\ 1 & \text{else.} \end{cases}$$

Artificial Neural Networks

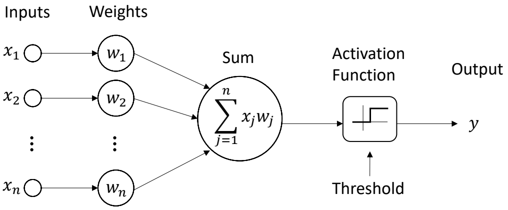


Figure: Illustration of an artificial neuron.

https://insights.sei.cmu.edu/sei_blog/2018/02/deep-learning-going-deeper-toward-meaningful-patterns-in-complex-data.html

Artificial Neural Networks with power activations - interpolation makes a comeback.

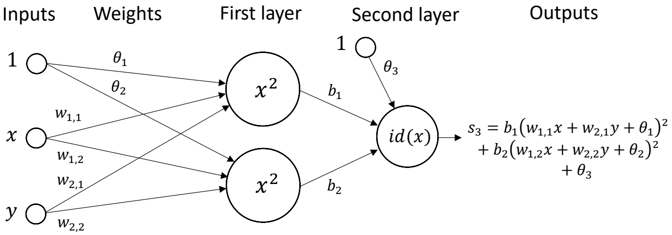
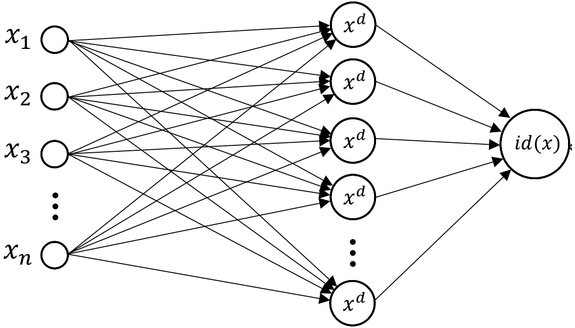


Figure: A network for a ternary quadric.

ANNs with power activations = polynomial interpolation.



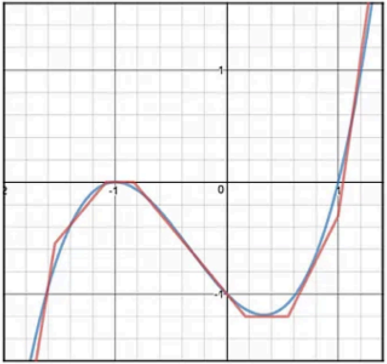
The output is a homogeneous polynomial depending on parameters. Polynomial Interpolation.

For multiple power function layers see [Kileel-Trager-Bruna'19] *On the expressive power of deep polynomial neural networks*.

Universal Approximation Theorem(s)

Theorem

ANNs (with conditions on width, depth, activations, etc.) can approximate any reasonable function



$$n_1(x) = Relu(-5x - 7.7)$$
$$n_2(x) = Relu(-1.2x - 1.3)$$
$$n_3(x) = Relu(1.2x + 1)$$
$$n_4(x) = Relu(1.2x - .2)$$
$$n_5(x) = Relu(2x - 1.1)$$
$$n_6(x) = Relu(5x - 5)$$

$$Z(x) = -n_1(x) - n_2(x) - n_3(x) + n_4(x) + n_5(x) + n_6(x)$$

https://towardsdatascience.com/can-neural-networks-really-learn-any-function-65e106617fc6

Idea 1. Use a hybrid network with powers and ReLU's:

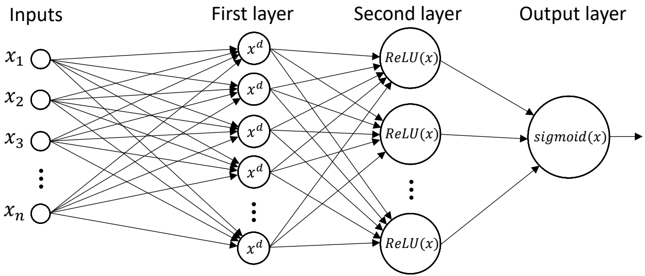


Figure: A network for binary classification for algebraic variety membership.

Example results for degenerate / non-degenerate classification

Tensor size	Architecture	Training acc.	Valid. acc.	Testing acc.	Loss
$2 \times 2 \times 2$	(100,50,25,16,1)	93.44%	92.53%	92.74%	0.1629
$2^{\times 4}$	(200,100,50,16,1)	99.50%	95.95%	95.94%	0.01791
$2^{\times 5}$	(100,50,25,16,1)	99.95%	98.74%	98.83%	0.001533
$3 \times 3 \times 3$	(100,50,25,16,1)	98.18%	96.78%	96.83%	0.04770

Table: LeakyReLU network architectures and accuracies for each tensor size, for degenerate and non-degenerate states classification.

Naively make use of symmetry: orbit recycling.

Suppose we trained the classifier $F$to accuracy 95%. Then we evaluate $F(z) = b$. How sure are we that z is in class b?

What if we trained F to accuracy 60%?

Suppose data we trained on has symmetry, i.e. for $g \in G$ (some group) if $F(x_i) = b_i$ then we know that $F(g.x_i) = b_i$ even if we didn't train on data point $(g.x_i, b_i)$.
Idea: make a histogram of results for $F(g.z)$ for randomly chosen $g \in G$.

Some other things you might discover

13.4% of tensors (with norm 1) in $\mathbb{R}^{2\times 2\times 2}$ have rank 3, 86.6% have rank 2.

Might have expected over $\mathbb{C}$ that 100% of tensors have rank 2 (the generic rank).

Typical real ranks are separated by the hyperdeterminant.

Tensor rank and the ill-posedness of the best low-rank approximation problem [de Silva - Lim 2006]

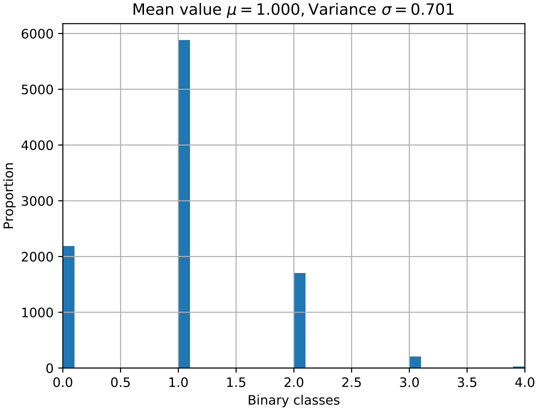


Figure: Histogram of the border rank classifier predictions for 10000 points SLOCC equivalent to the state $|W_5\rangle$. The plot predicts that the state is of border rank 2 (class ‘1’).

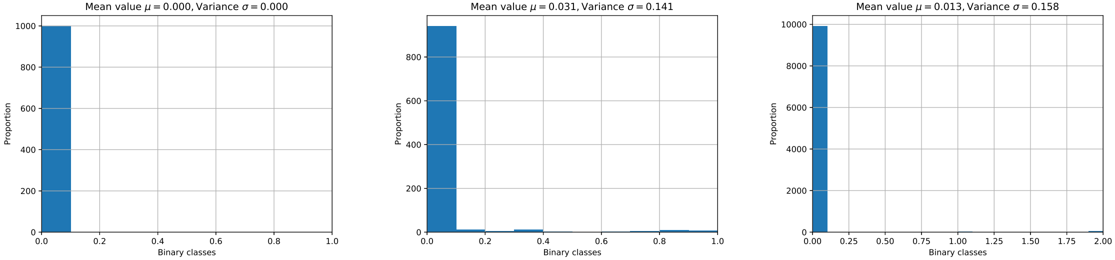


Figure: Histogram predictions for 1000 points SLOCC equivalent to $|000\rangle$ using our trained classifiers for (in order, from left to right) separable states, degenerate states and tensor rank. Being class '0' in each plot respectively predicts that the state is separable, degenerate, and of rank one.

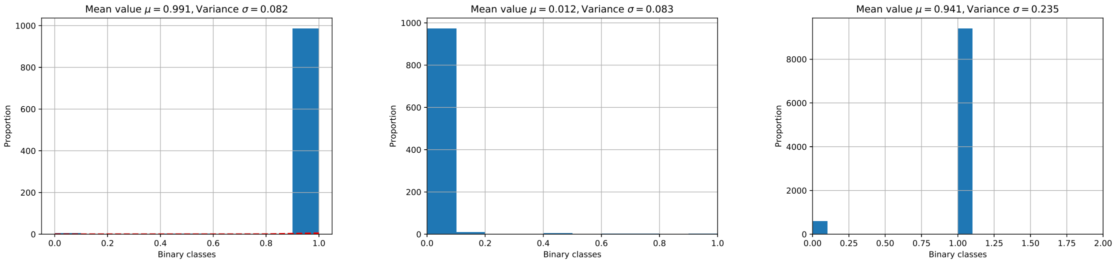


Figure: Histogram predictions for 1000 points that are SLOCC equivalent to $\frac{1}{\sqrt{2}}(|000\rangle + |011\rangle)$ using our trained classifier for (in order, from left to right) separable states, degenerate states and tensor rank classifiers. The plots predict the state is entangled (class '1' on the left), degenerate (class '0' in the middle), and of rank two (class '1' on the right).

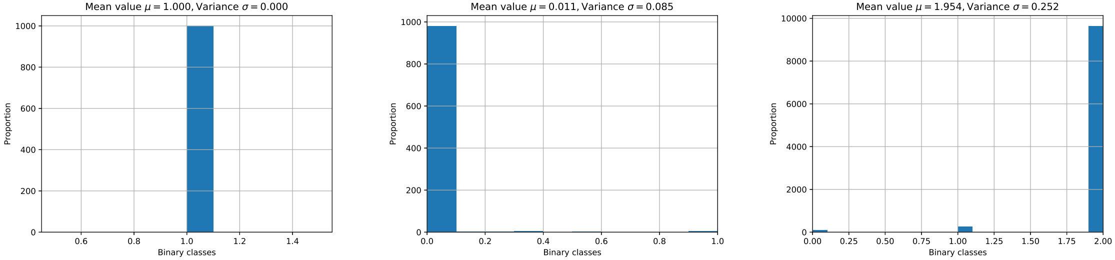


Figure: Histogram predictions for points that are SLOCC equivalent to the W-state $\frac{1}{\sqrt{3}}(|001\rangle + |010\rangle + |100\rangle)$ using our trained classifiers for (in order, from left to right) separable states, degenerate states and tensor rank. The left plot and middle plots use 1000 points and respectively predict that the state is entangled (class '1') and degenerate (class '0'). The right plot with 10000 equivalent points predicts that the state is of rank three (class '2').

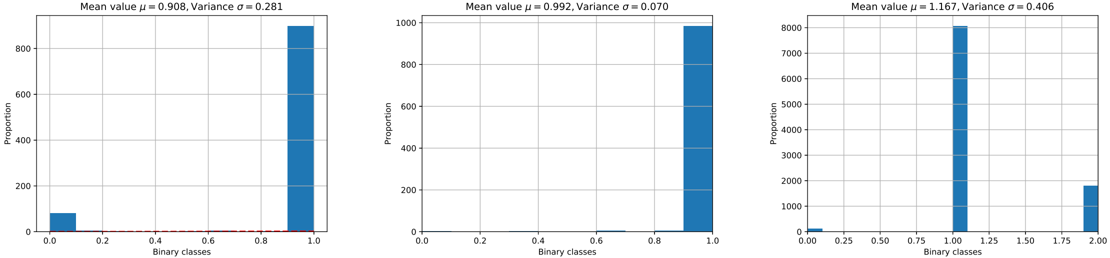


Figure: Histogram predictions for points that are SLOCC equivalent to the GHZ-state $\frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$ using our trained classifiers for (in order, from left to right) separable states, degenerate states and tensor rank. The left plot and middle plots use 1000 points and respectively predict that the state is entangled (class '1') and non-degenerate (class '1'). The right plot with 10000 equivalent points predicts that the state is of rank two (class '1').

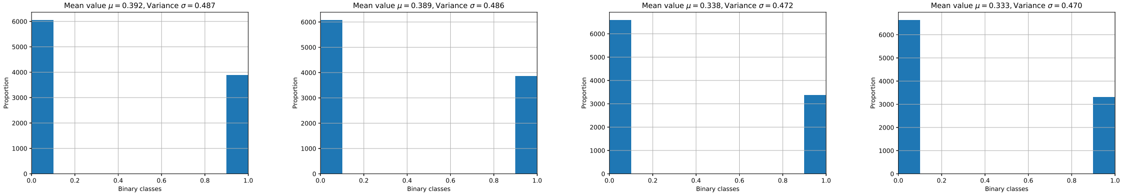


Figure: Histograms for the degenerate states classifier for 10000 points respectively SLOCC equivalent to the states $|\Phi_1\rangle$, $|\Phi_2\rangle$, $|\Phi_3\rangle$, and $|\Phi_4\rangle$, from left to right. Classes '0' and '1' respectively refer to degenerate and non-degenerate states. Here all states are predicted to be degenerate.

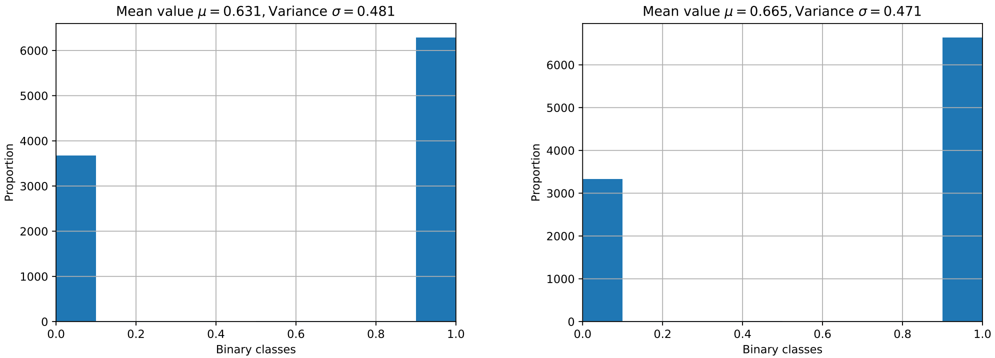


Figure: Histograms for the degenerate states classifier on 10000 points respectively SLOCC equivalent to $|\delta_1\rangle$ (left) and $|\delta_2\rangle$ (right). Classes '0' and '1' respectively refer to degenerate and non-degenerate states. Here both states are predicted to be non-degenerate.