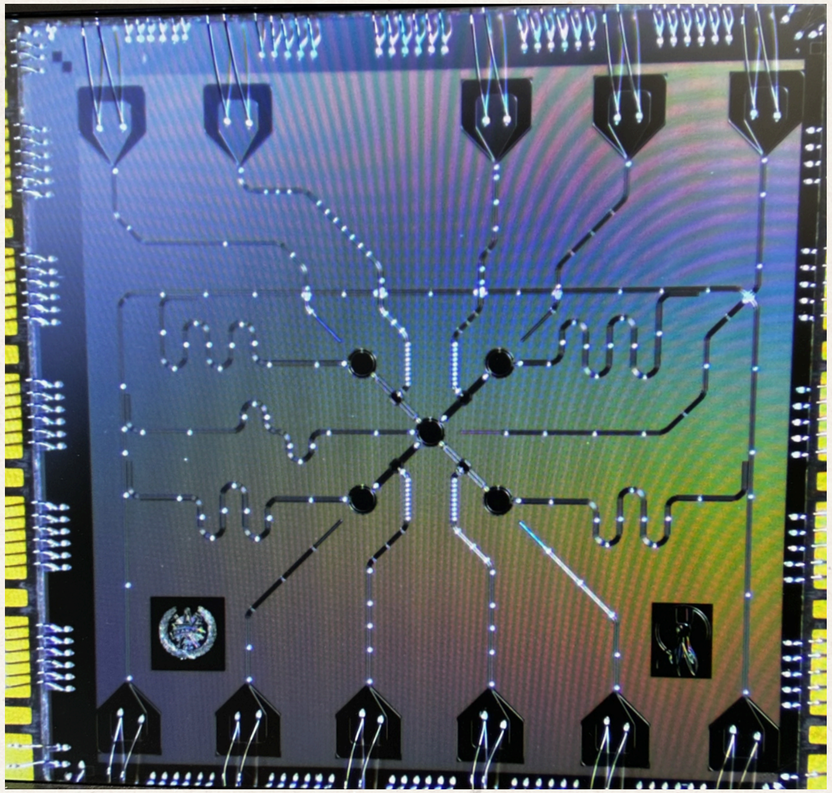
11/06/2025



ANALYZING MULTIPLEXED SUPERCONDUCTING QUBIT READOUTS

Using clustering, decision trees, and NNs

PRESENTED BY:
Daniel, Emil, Mattis, Rasmus and Victoria
Everyone contributed equally

Overview

	Raw data
	Experiment and labels
	Clustering 1
	Clustering 2
	Single Qubit Classifiers
	4 Qubit Classifiers
	Given infinite time
	Appendix

	Initial state				Features	Label			
Shots	**q_1**	**q_2**	**q_3**	**q_4**	**Q + I** (150 segments)	q_1	q_2	q_3	q_4
1, 2, . . 1000	0	0	0	0	[0.02 + 0.01i, ...]	*1,* *0,* *0,* . .	*0,* *0,* *1,* . .	*1,* *1,* *1,* . .	*0,* *1,* *0* . .
1, 2, . . 1000	0	0	0	1	. . . .	*0,* *0,* *1,* . .	*1,* *1,* *1,* . .	*0,* *1,* *0* . .	*1,* *0,* *0,* . .
. . . . . .	0	0	1	1	. . . .	*0,* *1,* *0* . .	*1,* *0,* *0,* . .	*0,* *0,* *1,* . .	*1,* *1,* *1,* . .

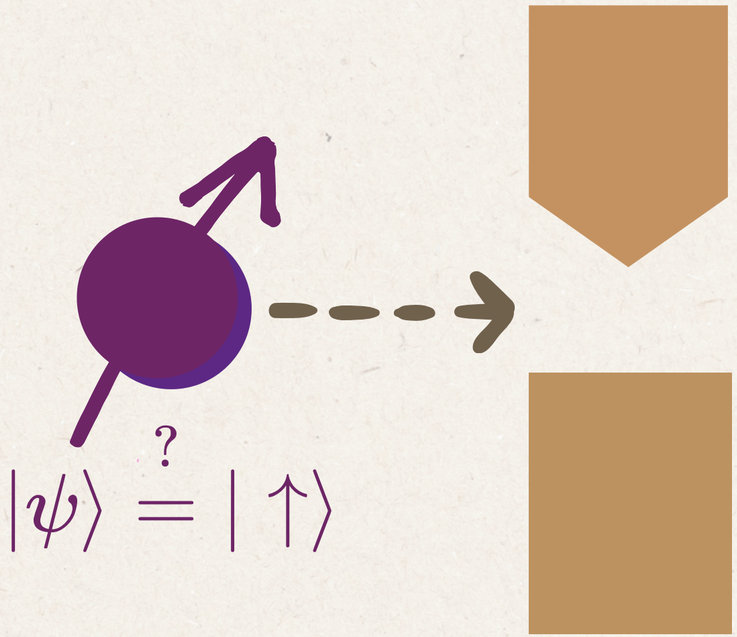
Data

Two experiments and their stuctures

01 Classification: Contains 1000 shots (experiments), initial states either 0 or 1, and *labels*, their outcomes.

02 ADC trace: Contains 1000 shots with 150 segments (time axis) and initial states either 0 or 1.

The experiment: control of a quantum state



$$|\psi\rangle \xrightarrow{x_1>0} |\uparrow\rangle$$

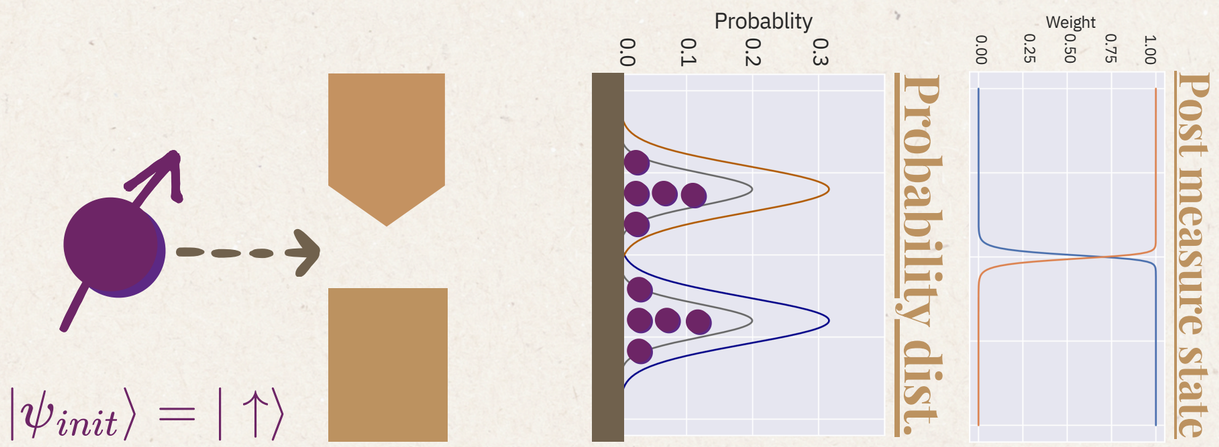
It was! :-)

Is the particle in the state we think it is in?

$|\psi_{init}\rangle = |\uparrow\rangle$

The experiment: control of a quantum state

Expectation

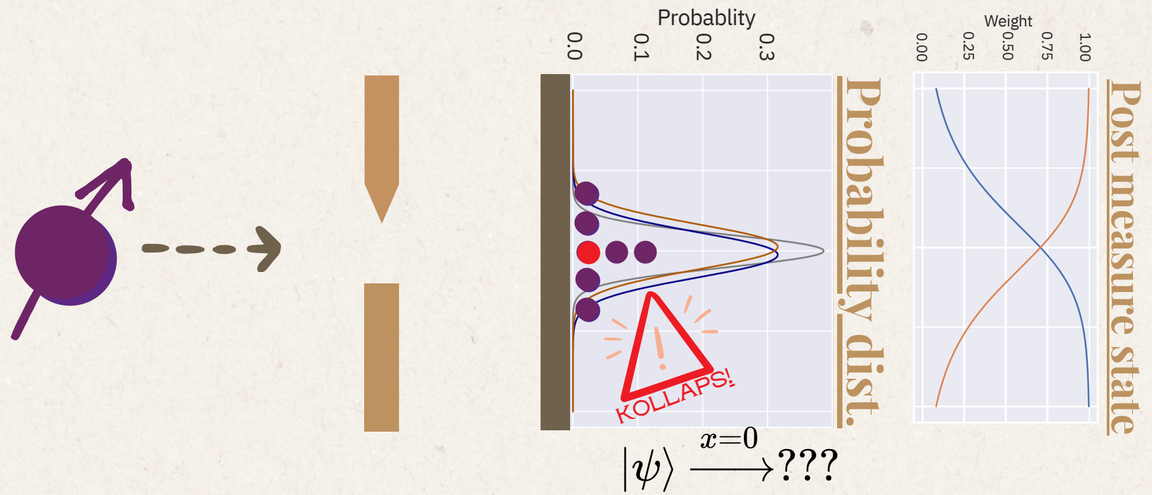


Distinct outcomes for different states

$|\uparrow\rangle$: Positive x

$|\downarrow\rangle$: Negative x

Reality



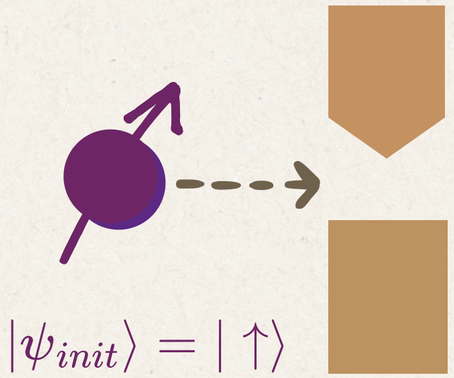
Overlapping outcomes for different states

$|\uparrow\rangle$: x near center (little shift up)

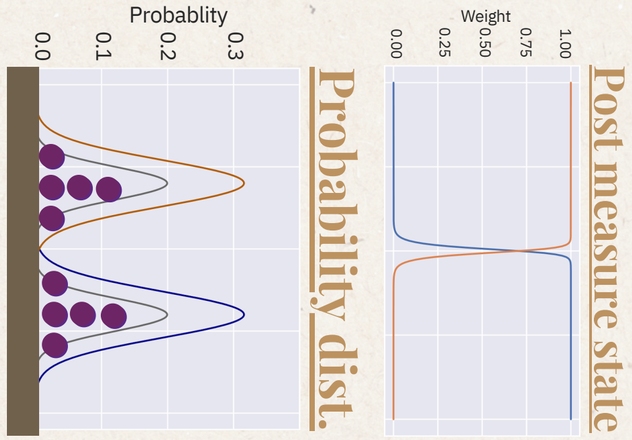
$|\downarrow\rangle$: x near center (little shift down)

The experiment: control of a quantum state

Expectation



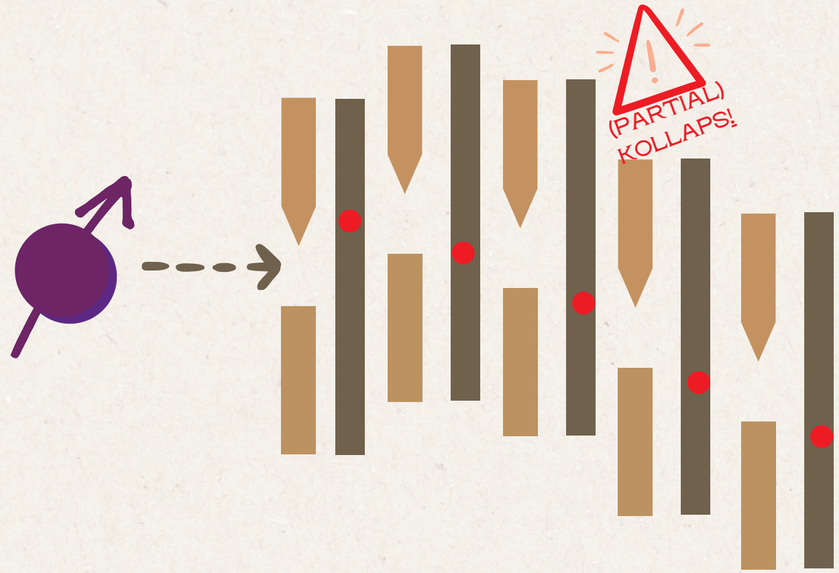
$|\psi_{init}\rangle = |\uparrow\rangle$



Distinct outcomes for different states

$|\uparrow\rangle$: **Positive x** $|\downarrow\rangle$: **Negative x**

Reality...?

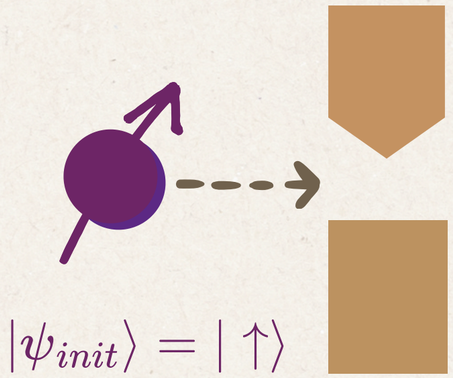


Several noisy measurements for better destinction

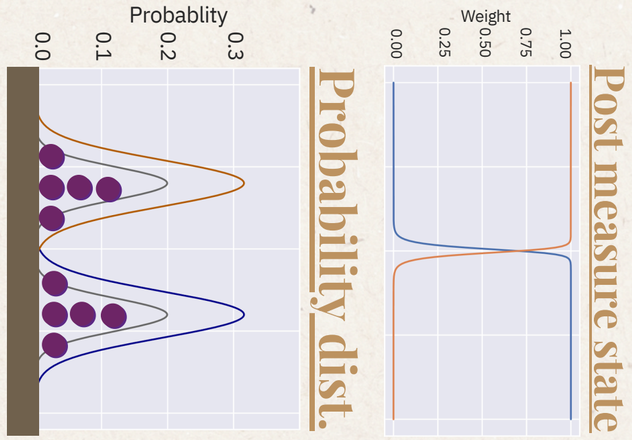
$|\uparrow\rangle$: **Positive sum x** $|\downarrow\rangle$: **negativ sum x**

The experiment: control of a quantum state

Expectation



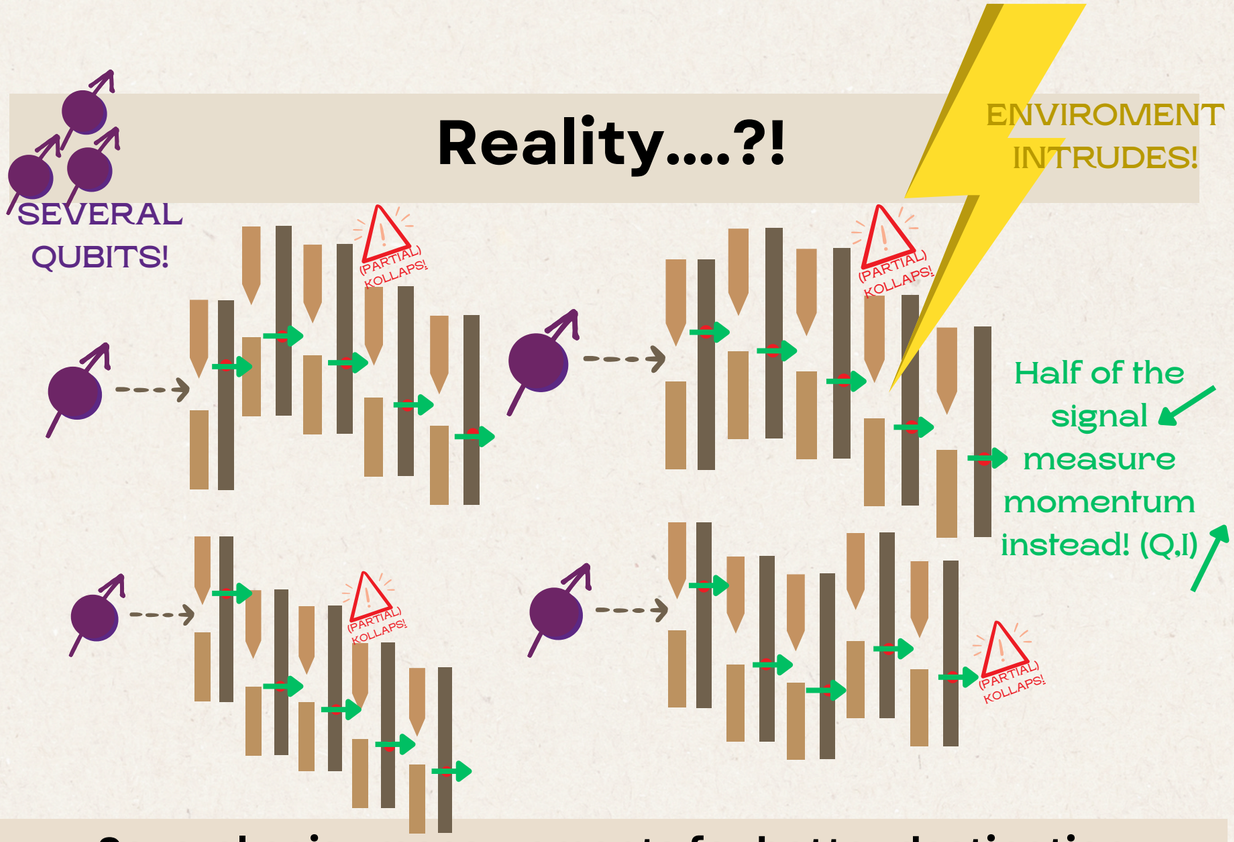
$|\psi_{init}\rangle = |\uparrow\rangle$



Distinct outcomes for different states

$|\uparrow\rangle$: **Positive x** $|\downarrow\rangle$: **Negative x**

Reality....?!

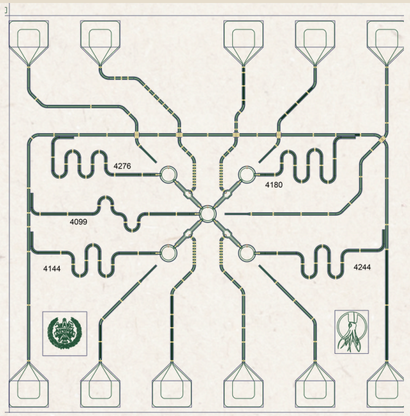


Several noisy measurements for better destinction

$|\uparrow\rangle$: **Positive sum x** $|\downarrow\rangle$: **neativ sum x**

The experiment: control of a quantum state

Actual experiment: Superconducting qubits



Aim of this project:

- Improve multi-qubit readout in the dispersive regime

Our approach

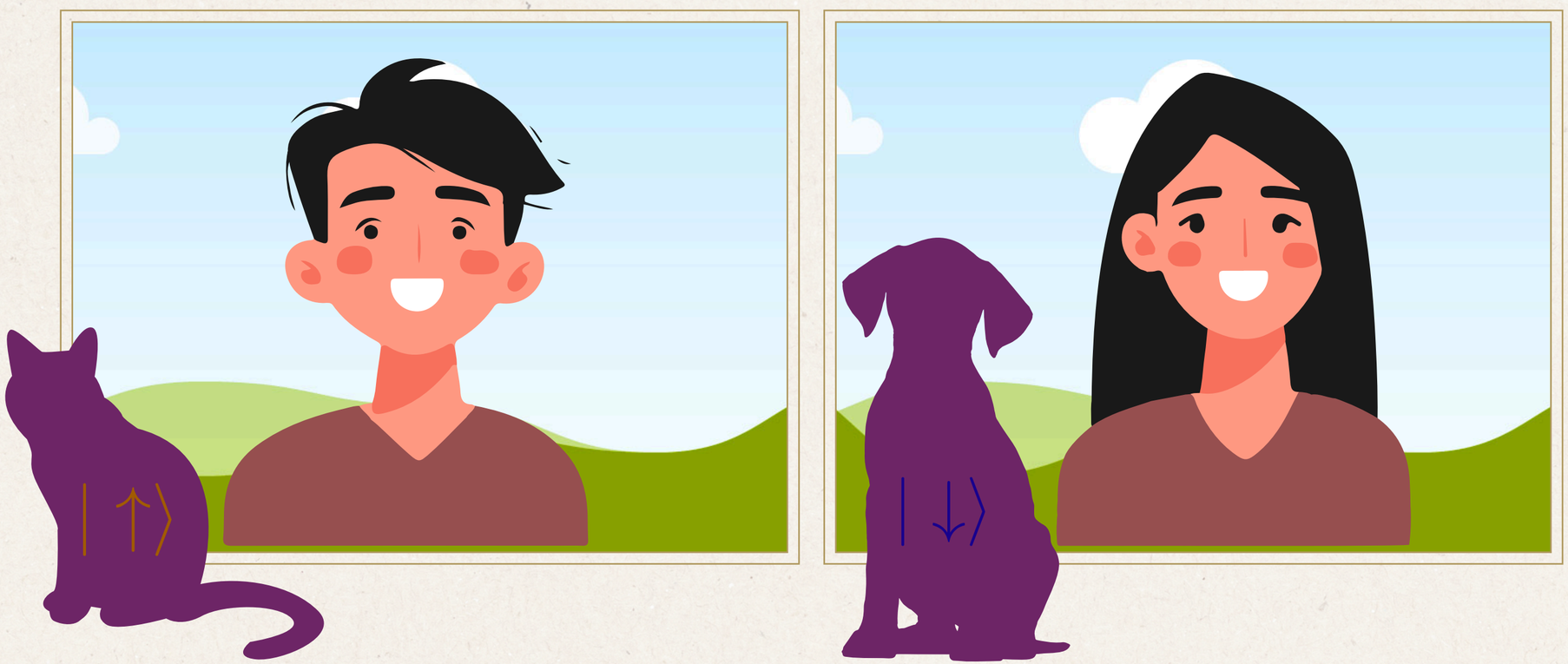
- Using quantum physics for assumptions for model building and interpretations.
- Using a lot of different clustering and classification from the course

	Initial state				Features	SQUID Label			
Shots	q_1	q_2	q_3	q_4	Q + I (150 segments)	q_1	q_2	q_3	q_4
1,	0	0	0	0	[0.02 + 0.01i, ...]	*1,*	*0,*	*1,*	*0,*
2,	.	.	.	.	.	*0,*	*0,*	*1,*	*1,*
.	.	.	.	.	.	*0,*	*1,*	*1,*	*0*
.	.	.	.	.	.	.	.	.	.
1000	.	.	.	.	.	.	.	.	.
1,	0	0	0	1	.	*0,*	*1,*	*0,*	*1,*
2,	.	.	.	.	.	*0,*	*1,*	*1,*	*0,*
.	.	.	.	.	.	*1,*	*1,*	*0*	*0,*
.	.	.	.	.	.	.	.	.	.
1000	.	.	.	.	.	.	.	.	.
.	0	0	1	1	.	*0,*	*1,*	*0,*	*1,*
.	.	.	.	.	.	*1,*	*0,*	*0,*	*1,*
.	.	.	.	.	.	*0*	*0,*	*1,*	*1,*
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.

... What?

Initialize Qubit state → Measure qubit (indirecly) → Predict qubit state

Predicting whether you have a cat or a dog based its owners - that's a problem, we have no reliable labels :(



Overview of ML models

Goal 1: Find good labels

After readout, what is the best estimate of the qubit state?

	Assumption	Method
1	The qubit is not changing in time (a dog stays a dog)	Use initial state at control state (Not ML)
2	Cummulated measurement record is clearly distinct for state |0> and |1>	**Clustering using Kmeans, GMM**
3	Each measurement depends on its current state	Use stochastic master equation for dispersive readout

Goal 2: Predit qubit state

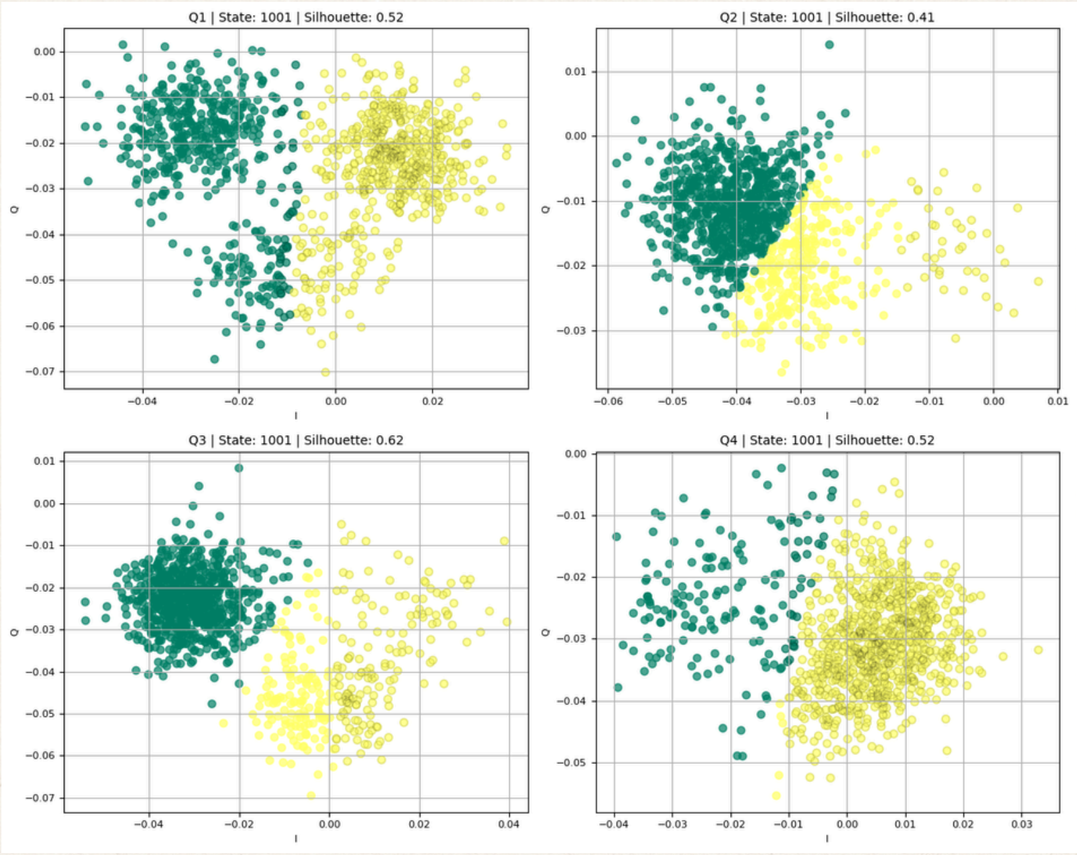
How to predict state for new experiments?

Model	Method	Label used
LightGBM classifier	LightGBM	2
Single qubit classifiers	LSTM	1
		2
4 Qubit classifiers	1DConv + FF	1
		2
Single Qubit regression	LightGBM	3

Clustering 1

Clustering using K-means

- First just an initial clustering with K-means and splitting into two clusters



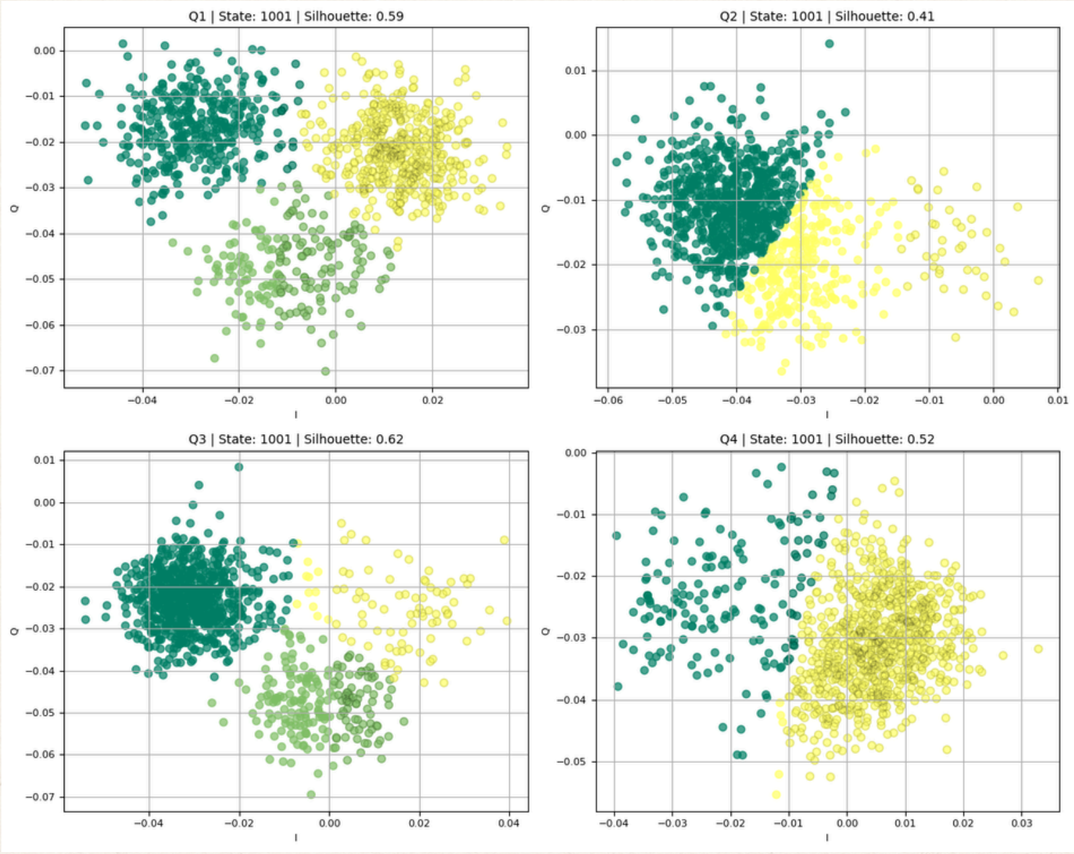
Clustering 1

Clustering using K-means

- First just an initial clustering with K-means and splitting into two clusters

Special cases for both qubit 1 and 3

- Seemed like 3 clusters instead of the expected 2



Clustering 1

Clustering using K-means

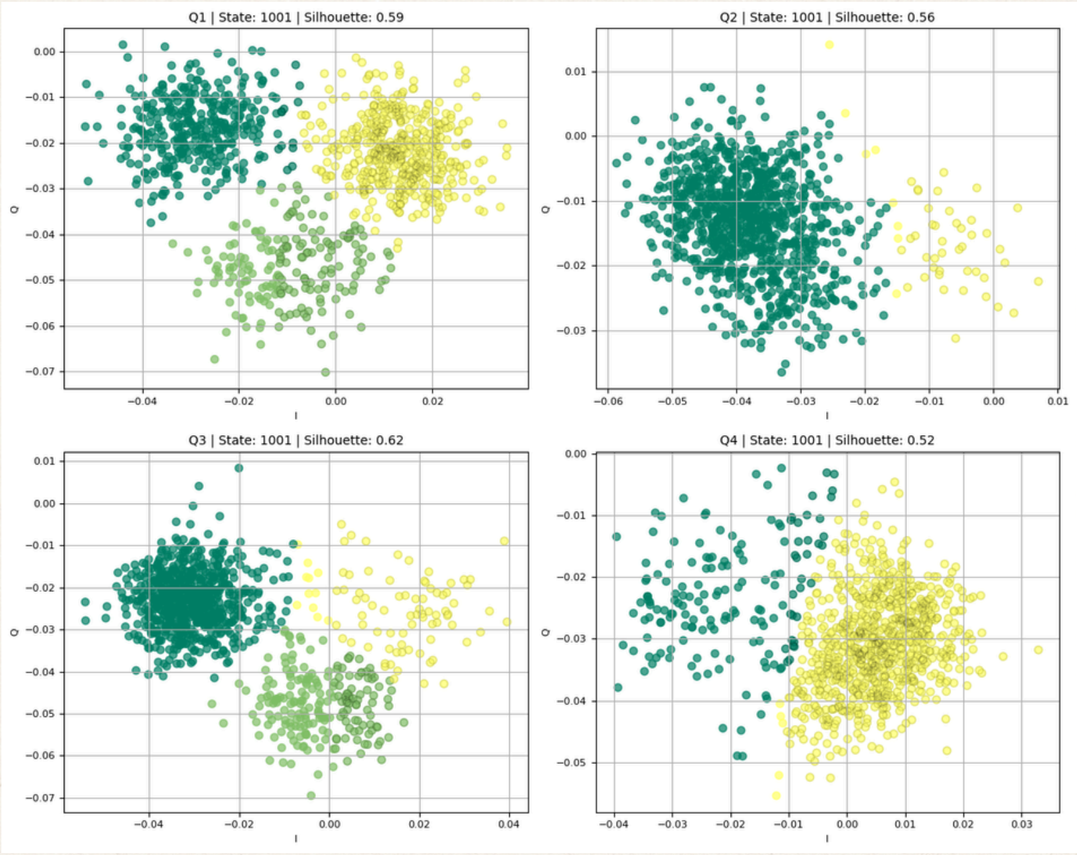
- First just an initial clustering with K-means and splitting into two clusters

Special cases for both qubit 1 and 3

- Seemed like 3 clusters instead of the expected 2

Special Case for qubit 2 initial state 0

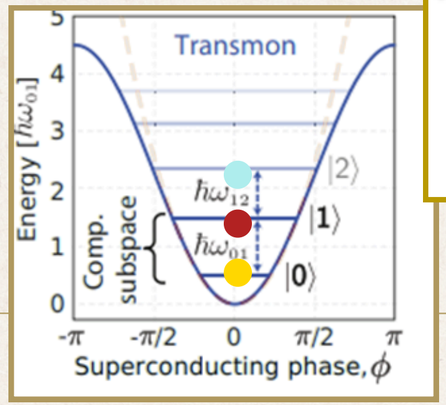
- More like a single cluster with some outliers to the right
- Eliptic envelope used for outlier detection.



Clustering 2

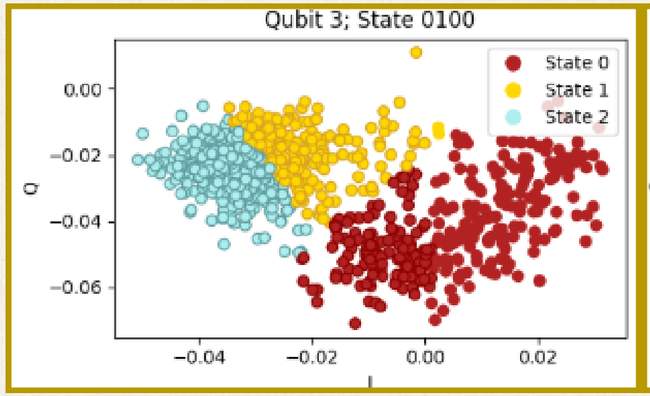
Clustering

- Attempted spectral, Kmeans, and GMM
- GMM winner
  - Force centers in 2D space
  - Consistent clusters
- Extra label for qubits in unwanted 2nd excited state!

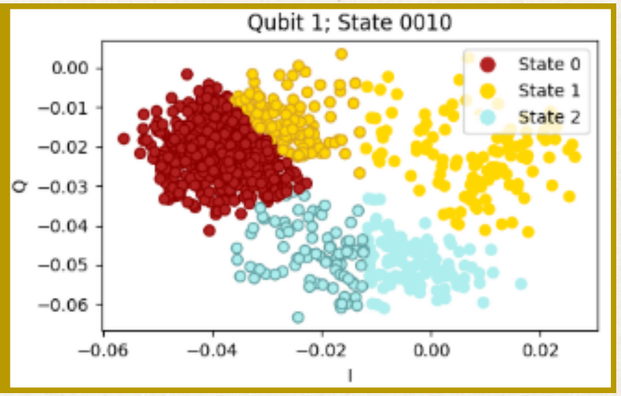


Clustering (the bad ones):

Kmeans

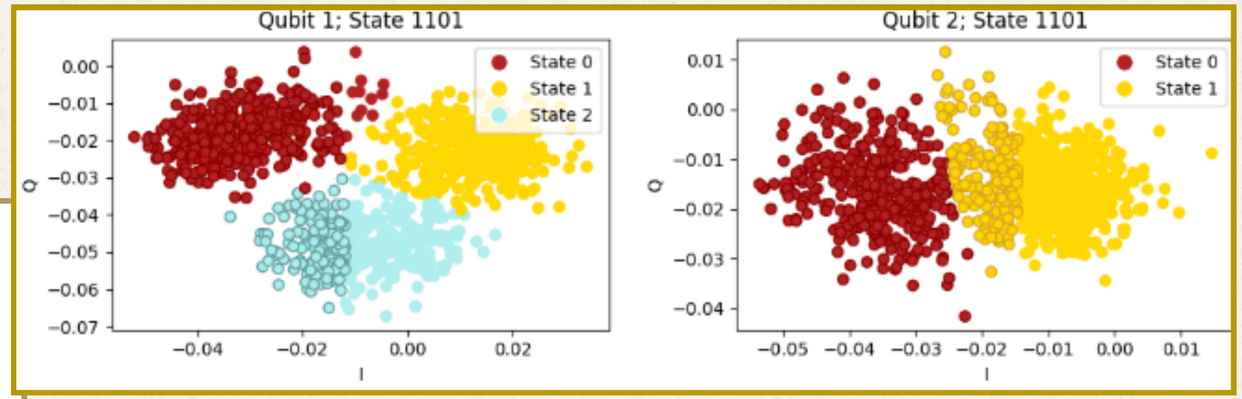


Spectral



Clustering (the good and slightly worse):

GMM

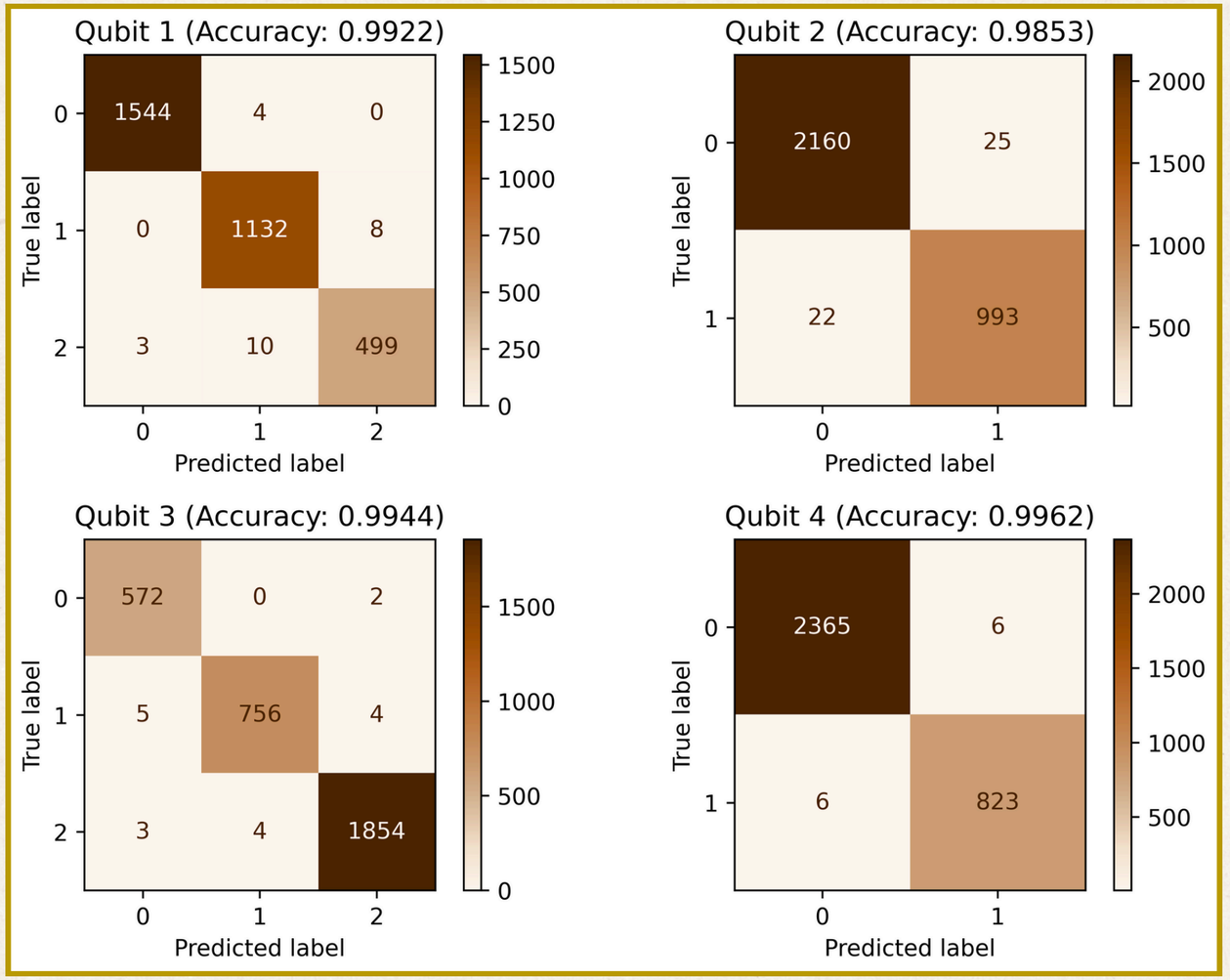


Classification
1 Qubit

LightGBM on clustering labels

- Not scientific (!!!)
- Accuracy of classifier limited by accuracy of clusters
- Done mostly because our data expert asked for it

Classifier confusion matrix



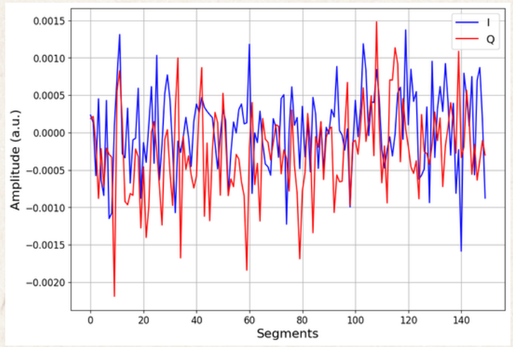
Preparing for ML classification

Data processing

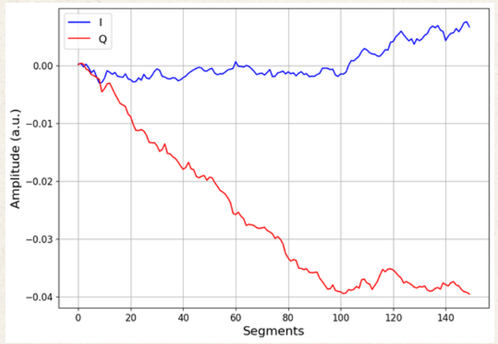
Organising data to show most important features.

- Individual measurement output show very little information about the state of the qubit
- The change of Q matters - more visible if we sum cumulatively.

Experimental data

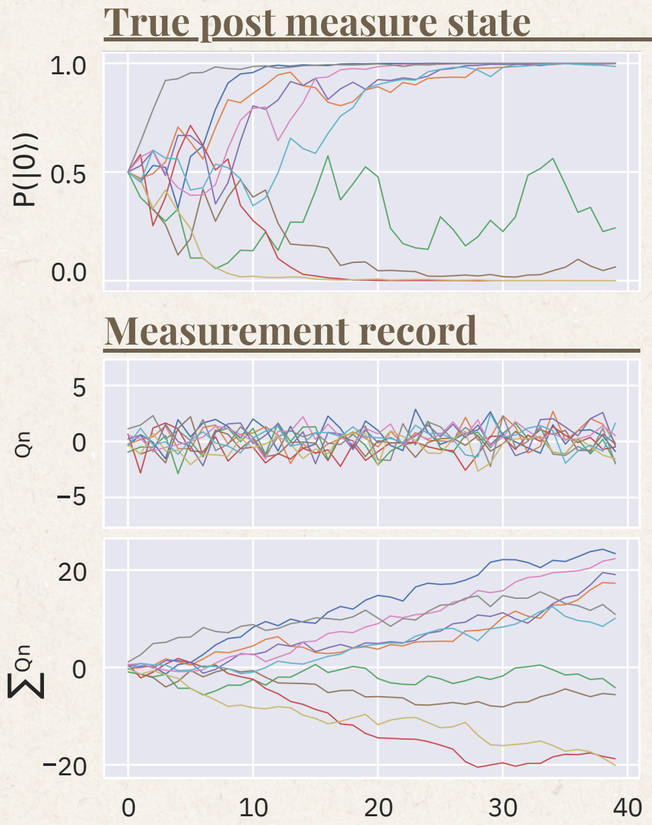


Raw data plot



Processed data

Simulating Q measure



Each color represents a measurement, from initial state

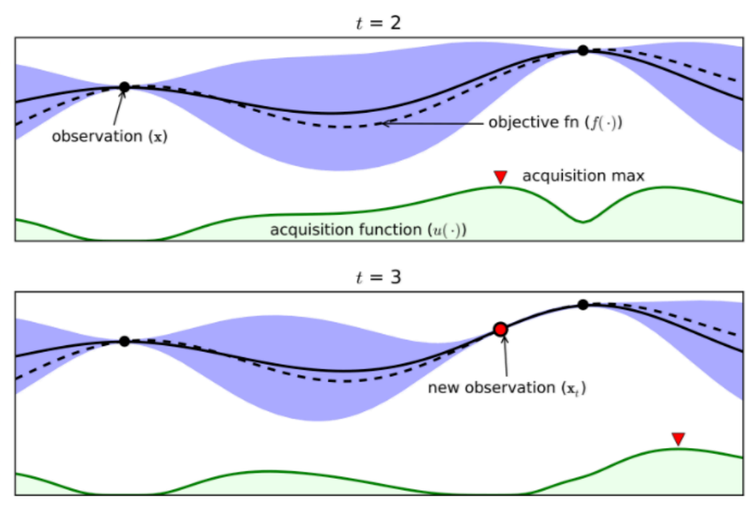
Raw measurement output: Now obvious difference

The cumulative sum distinguishes states! Example, blue is high

Overview of single qubit classifiers

Important Hyper parameters

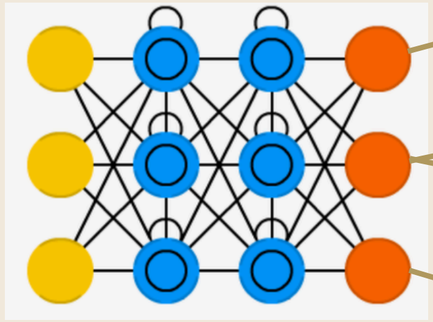
- Memory in the network
- How quickly does it learn



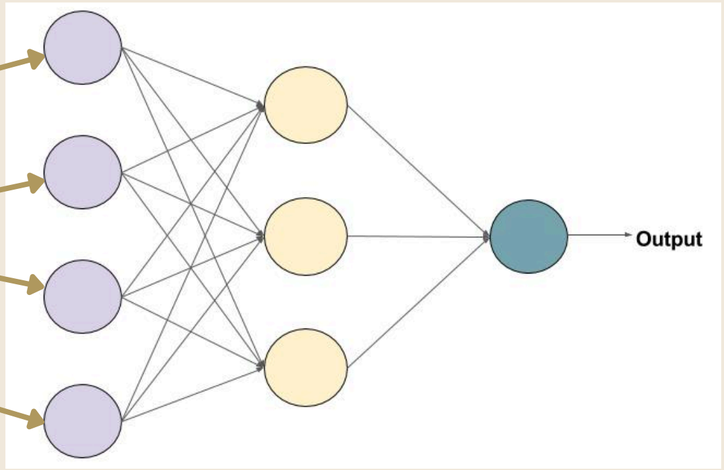
Bayesian Optimization

General Architecture

```
for i in range(num_layers):
    LSTM(n_units)
    Dropout(0.15)
Dense(relu)
Dense(sigmoid)
```



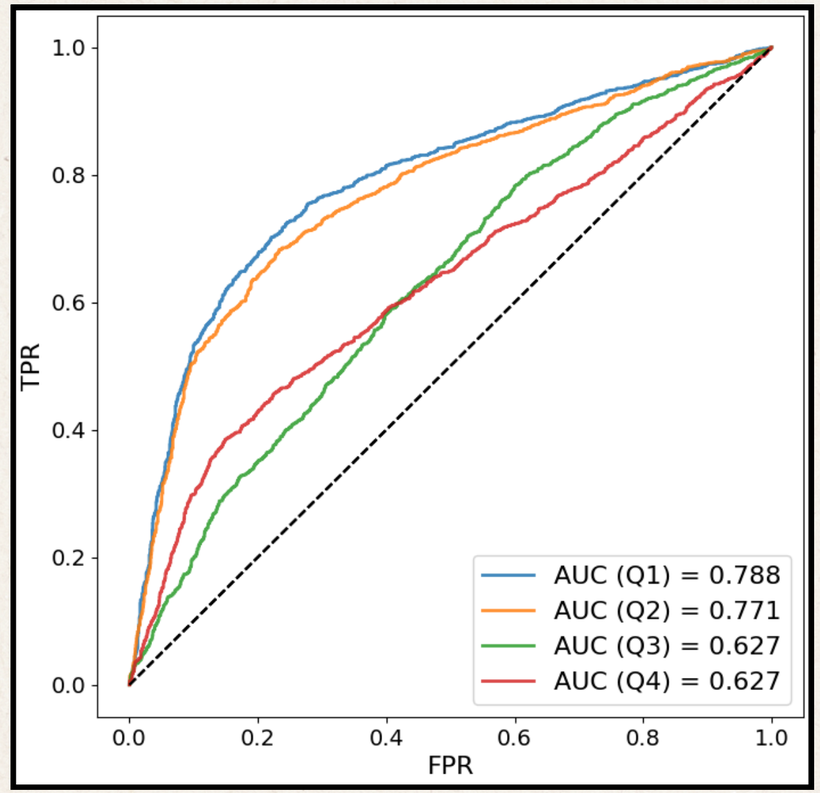
LSTM Memory units

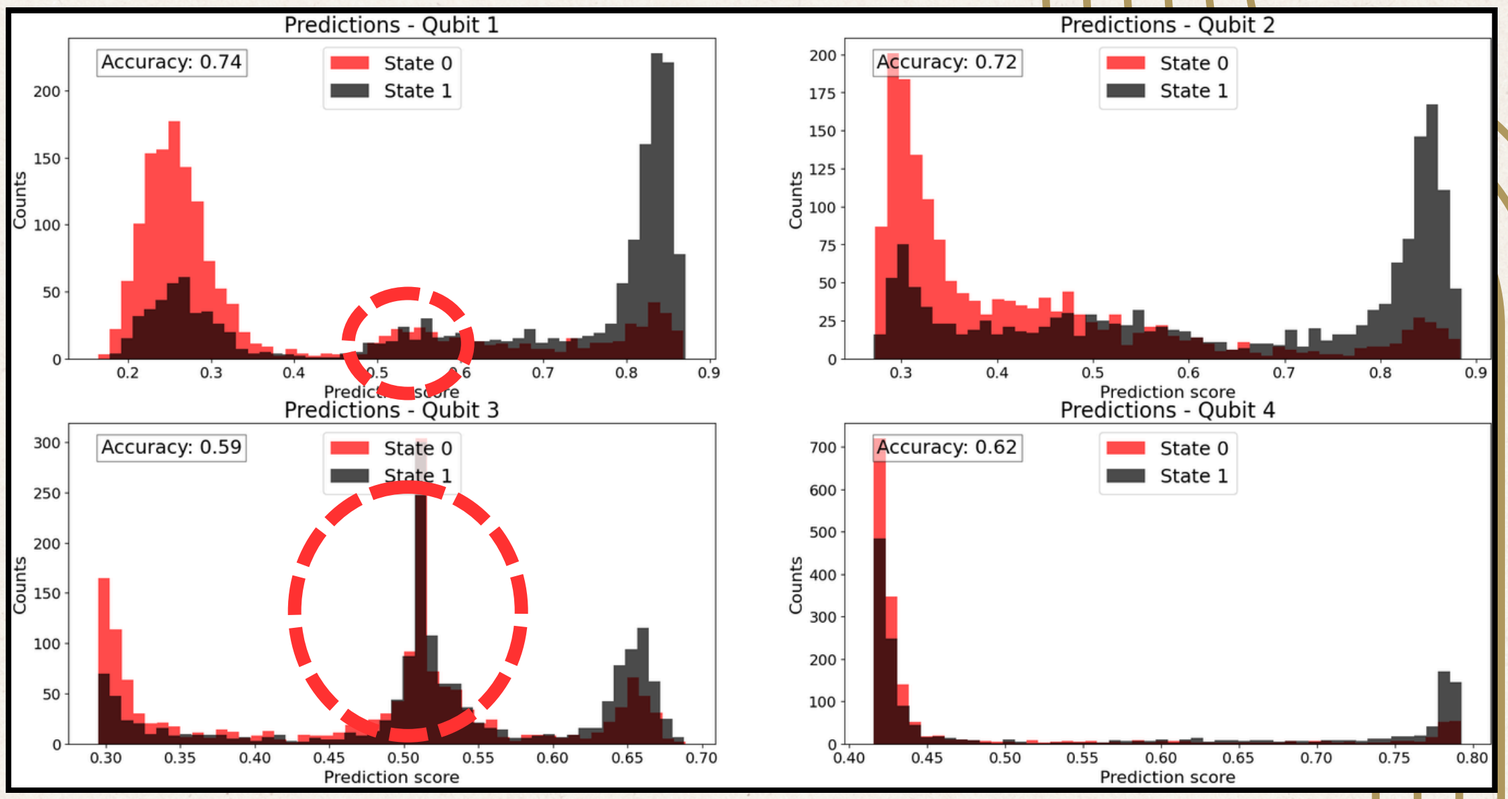


Compression to binary output (FFNN)

1st Tensorflow classifier
Initial state labels

Assumption: We assume the initial state and the final state is the same.

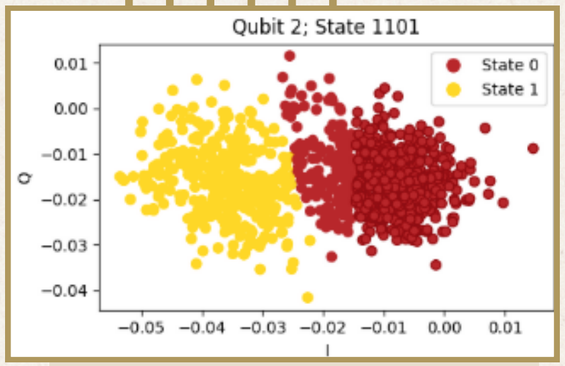




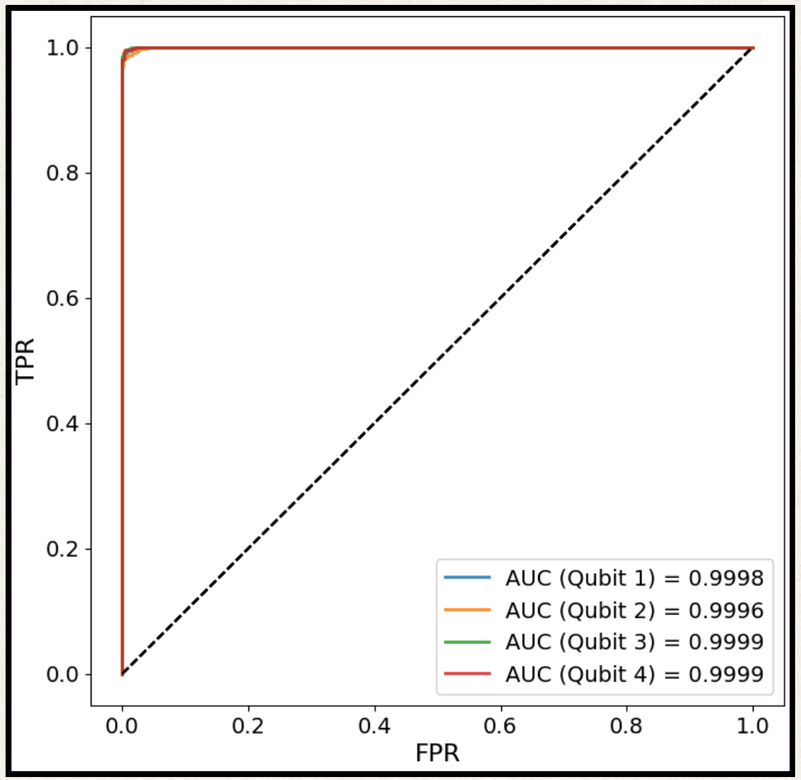
Possible 2nd excited state?

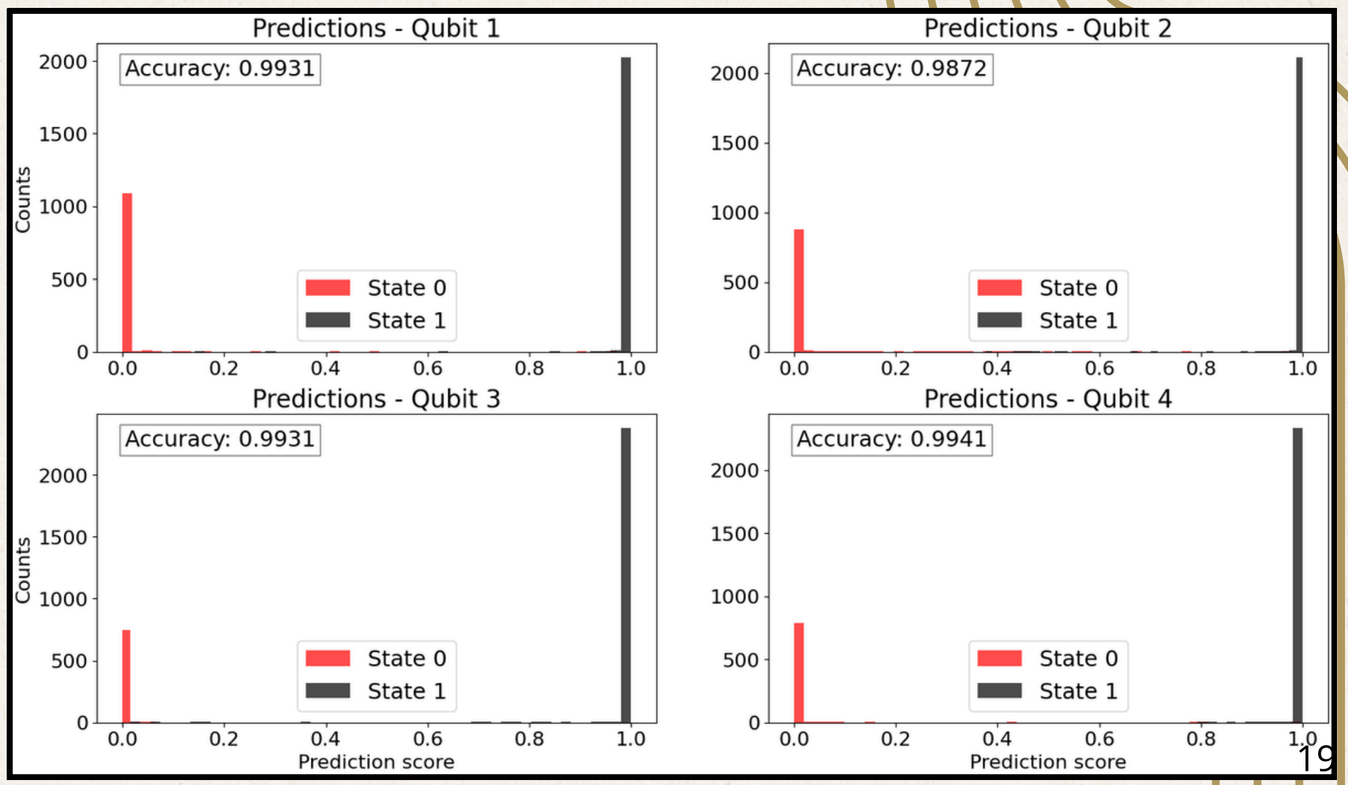
2nd Tensorflow classifier

Binary Clustering labels $|0\rangle$ or $|1\rangle$



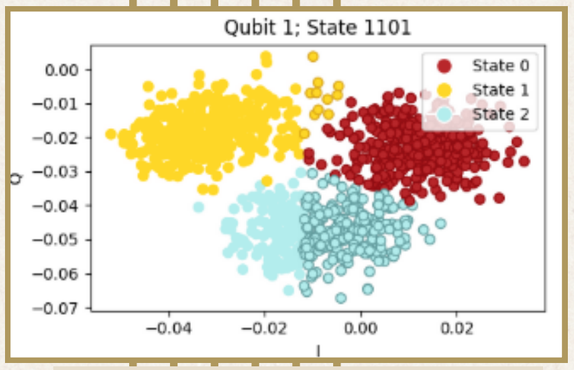
Labels obtained from clustering





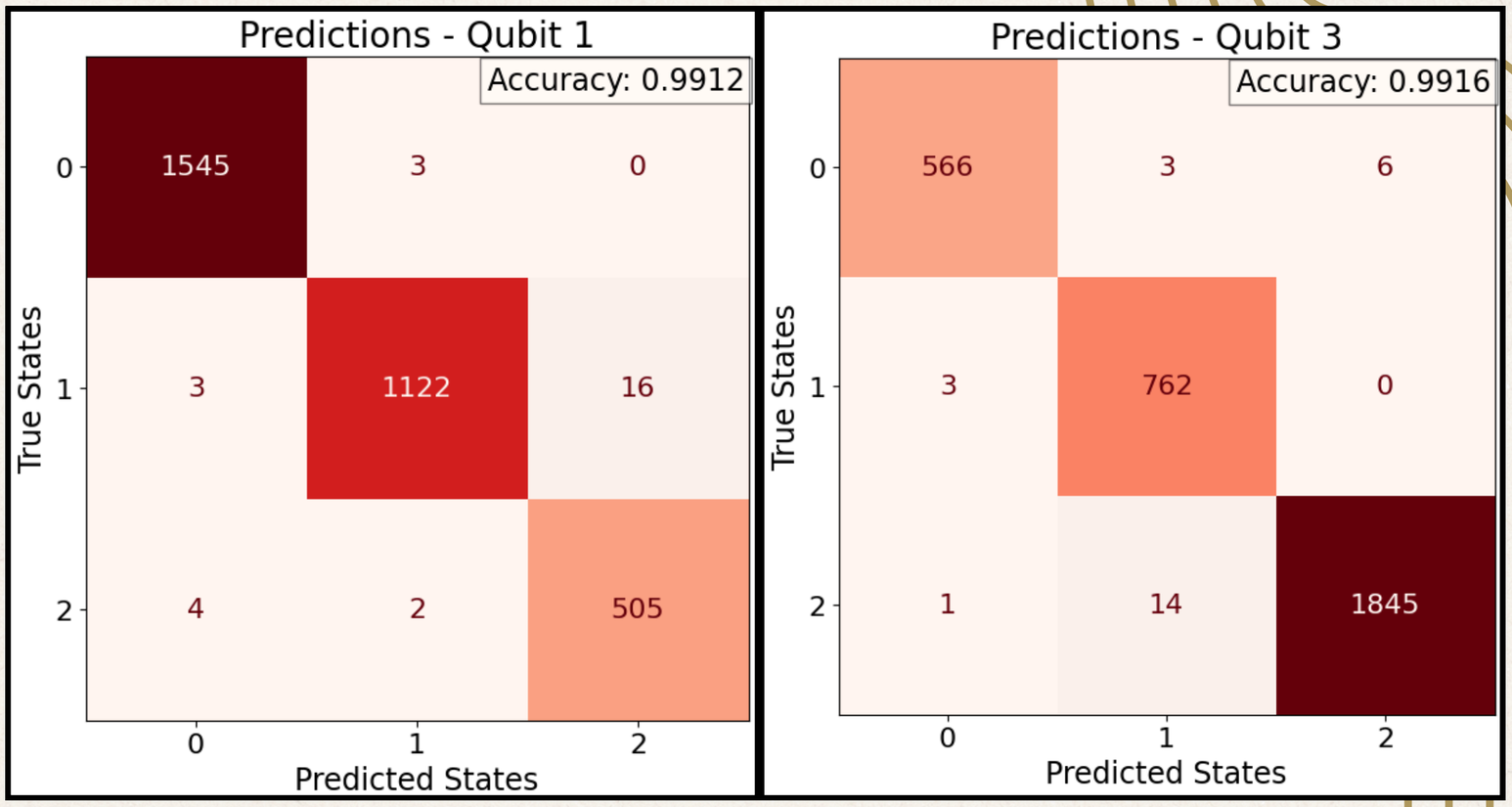
3rd Tensorflow classifier

3 State labels $|0\rangle, |1\rangle$ or $|2\rangle$



Labels obtained from clustering

Final Activation function change: sigmoid() → softmax()



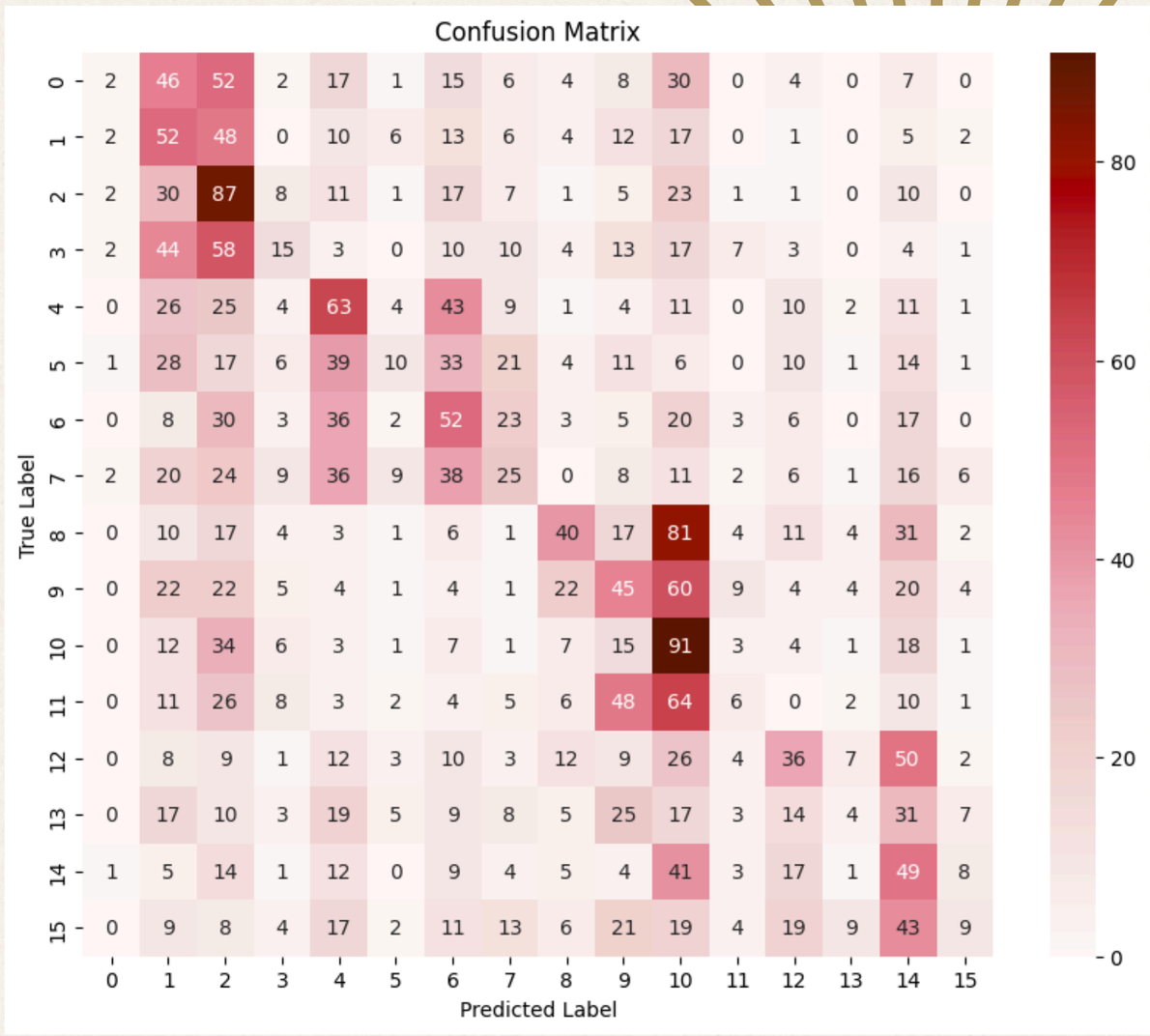
- **Only qubit 1 and 3 have 3 states**
- **High accuracy and changing majorities between states**

4 Qubit model with initial state as labels

$$|0101\rangle \rightarrow 5$$

$$|1000\rangle \rightarrow 8$$

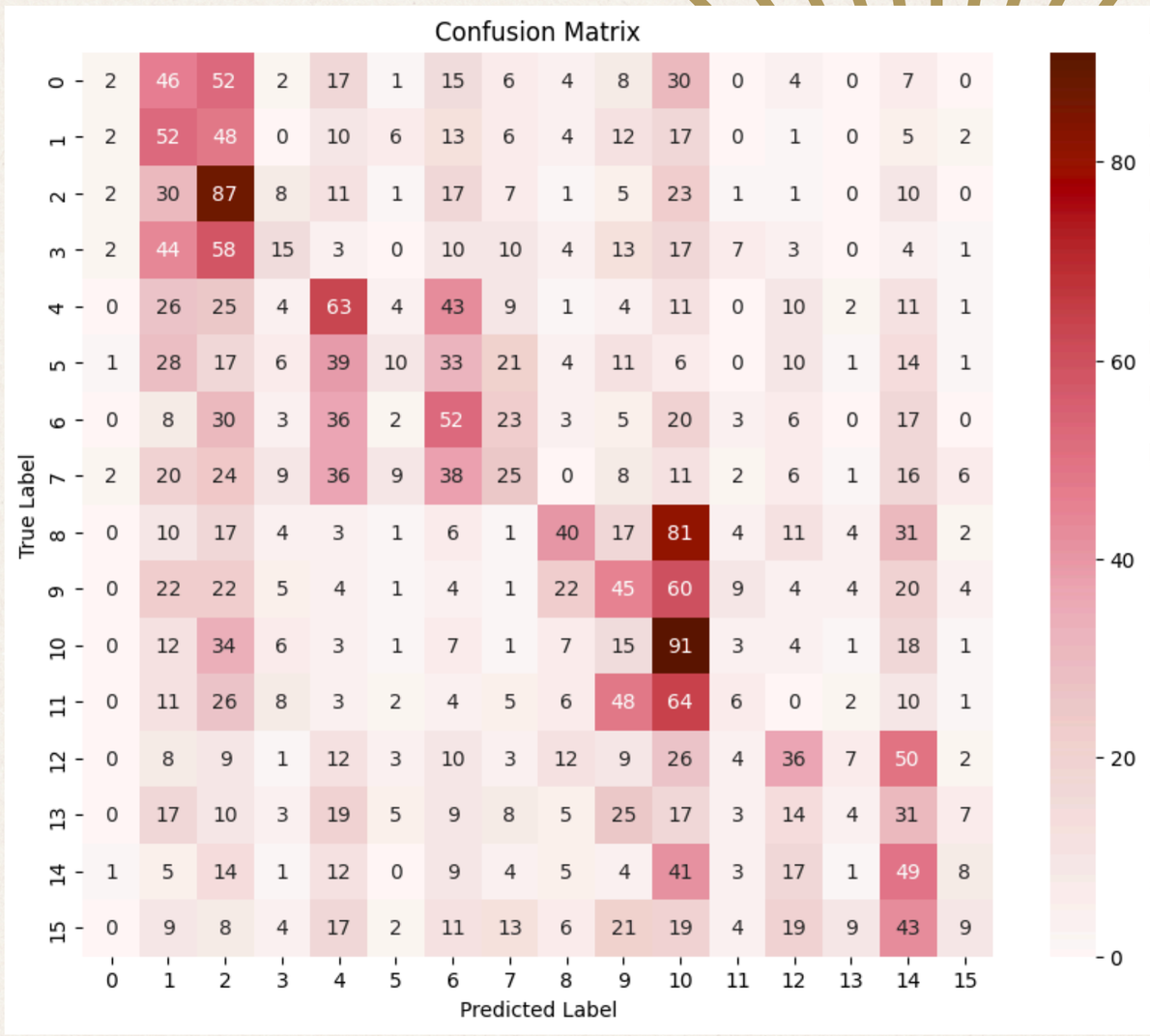
$$|0011\rangle \rightarrow 3$$



4 Qubit model with initial state as labels

1D Convolutional combined with ff layers

~18% Accuracy...

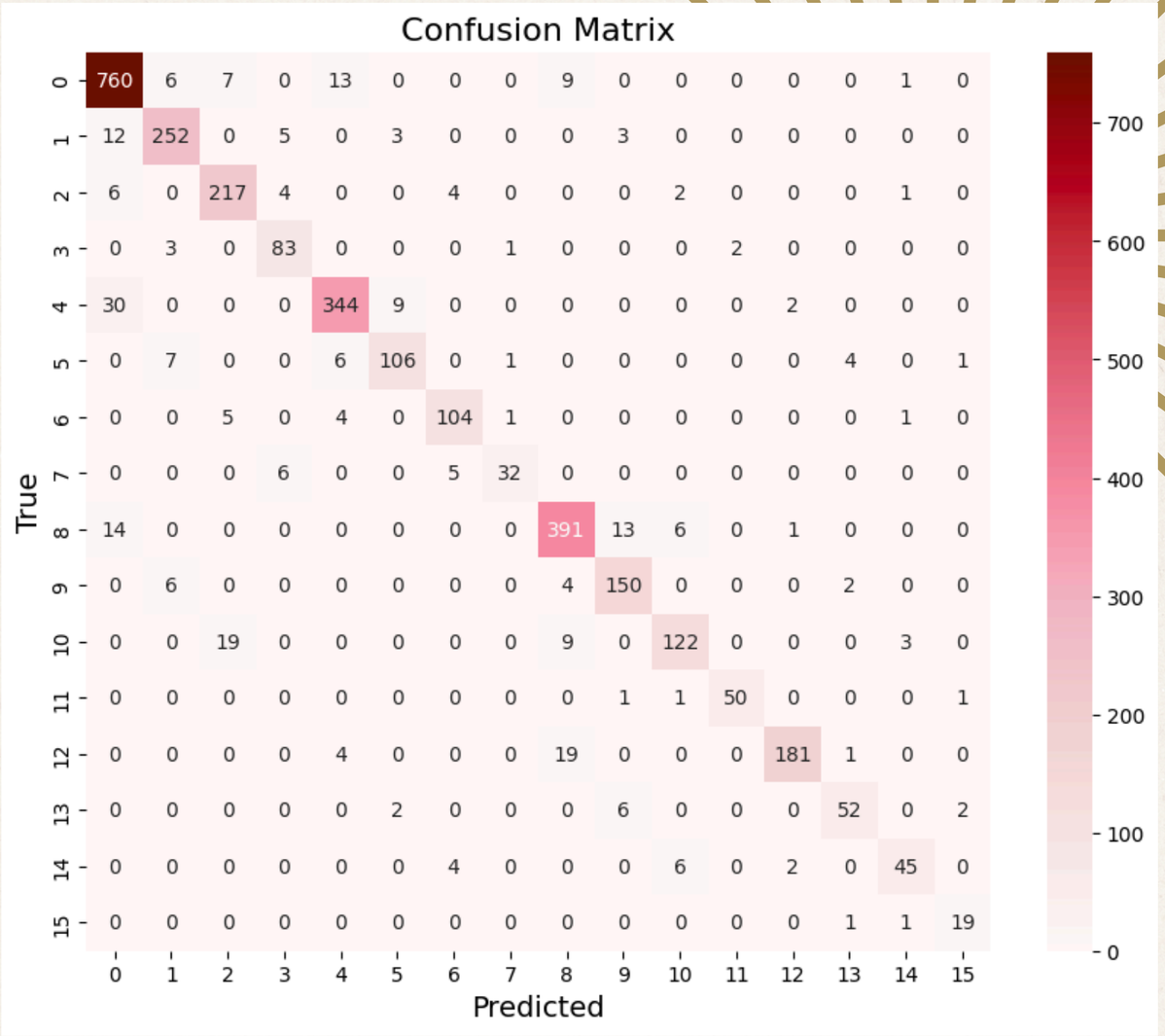


4 Qubit model with clustering labels (2 clusters)

1D Convolutional combined with ff layers

~500000 parameters

~91% Accuracy

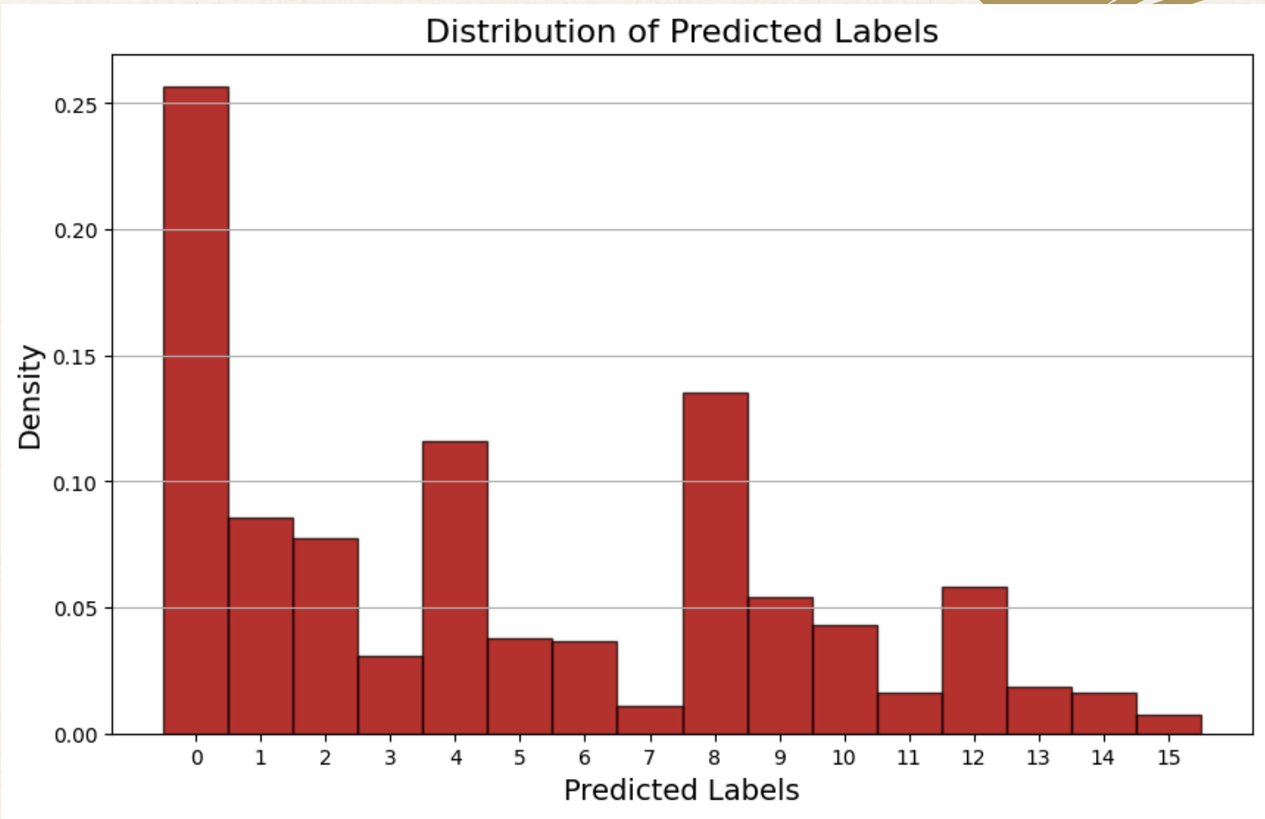


4 Qubit model with clustering labels (2 clusters)

1D Convolutional
combined with
ff layers

~500000 parameters

~91% Accuracy



Number of counts for each state

Qubit:	State 0	State 1	State 2
Qubit 1:	7705	5793	2502
Qubit 2:	10740	5260	0
Qubit 3:	3008	3873	9119
Qubit 4:	11810	4190	0

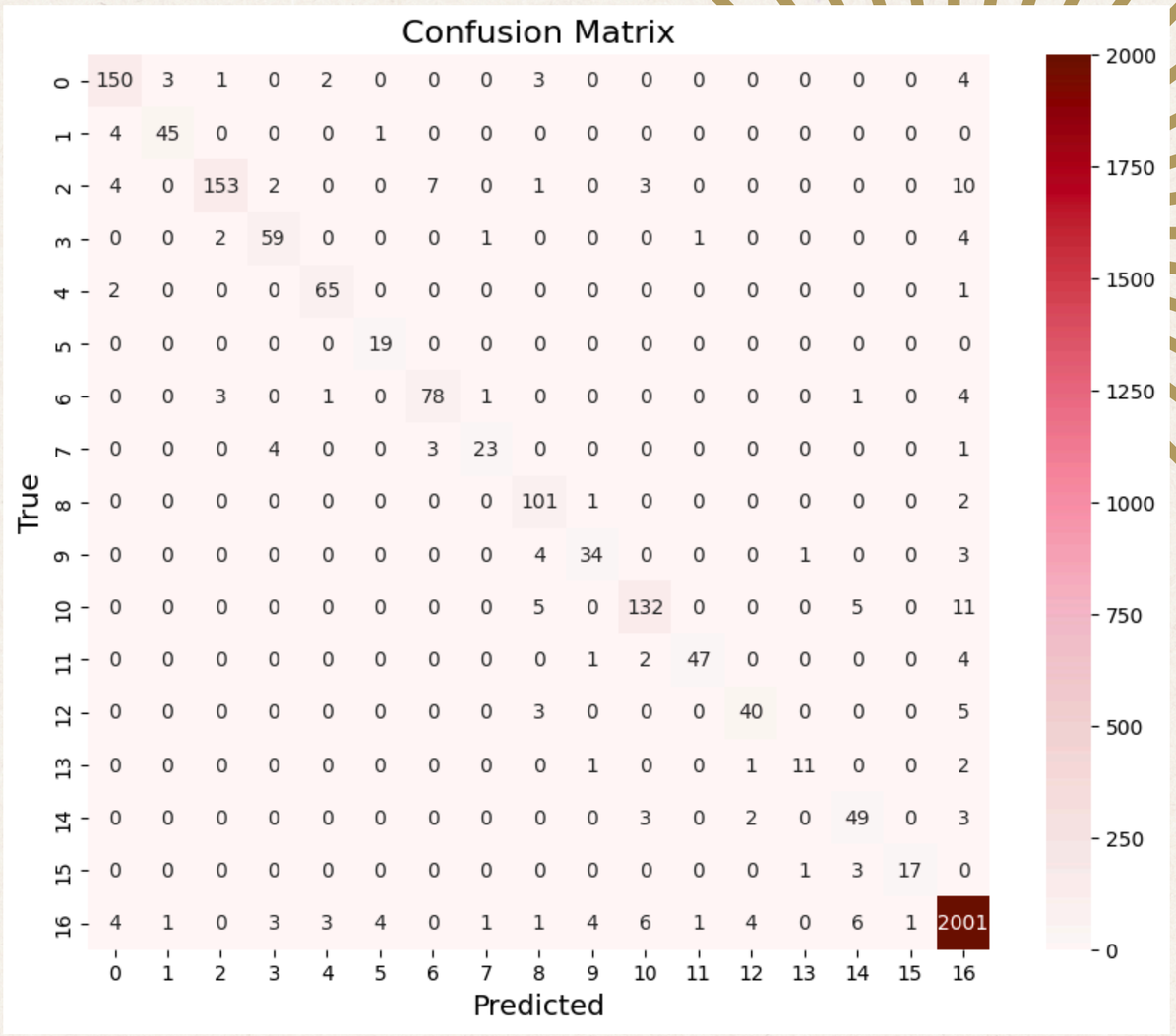
4 Qubit model with clustering labels (3 clusters)

$$|0101\rangle \rightarrow 5$$

$$|1000\rangle \rightarrow 8$$

$$|0021\rangle \rightarrow 16$$

$$|2000\rangle \rightarrow 16$$

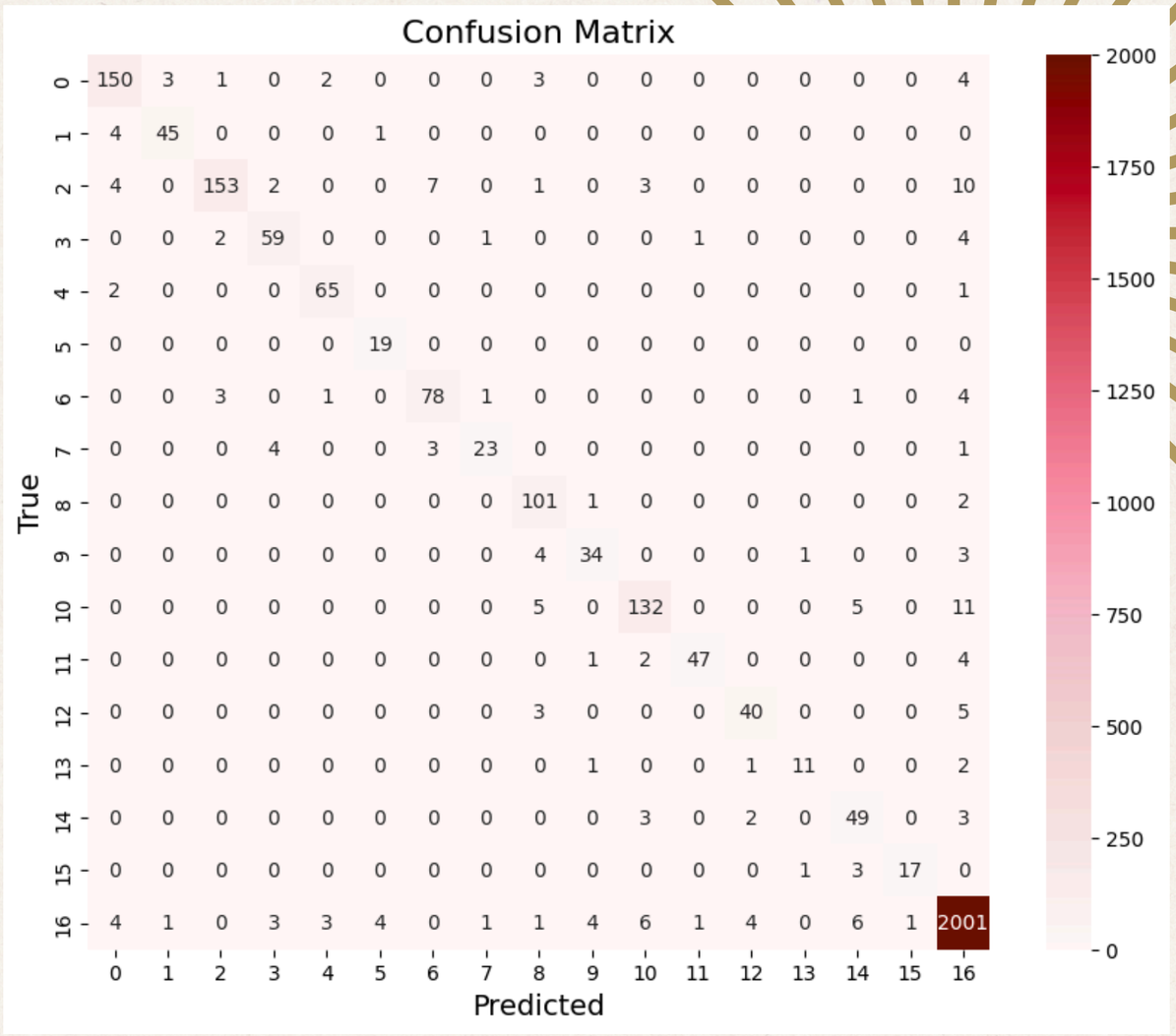


4 Qubit model with clustering labels (3 clusters)

1D Convolutional
combined with
ff layers

~300000 parameters

~95% Accuracy

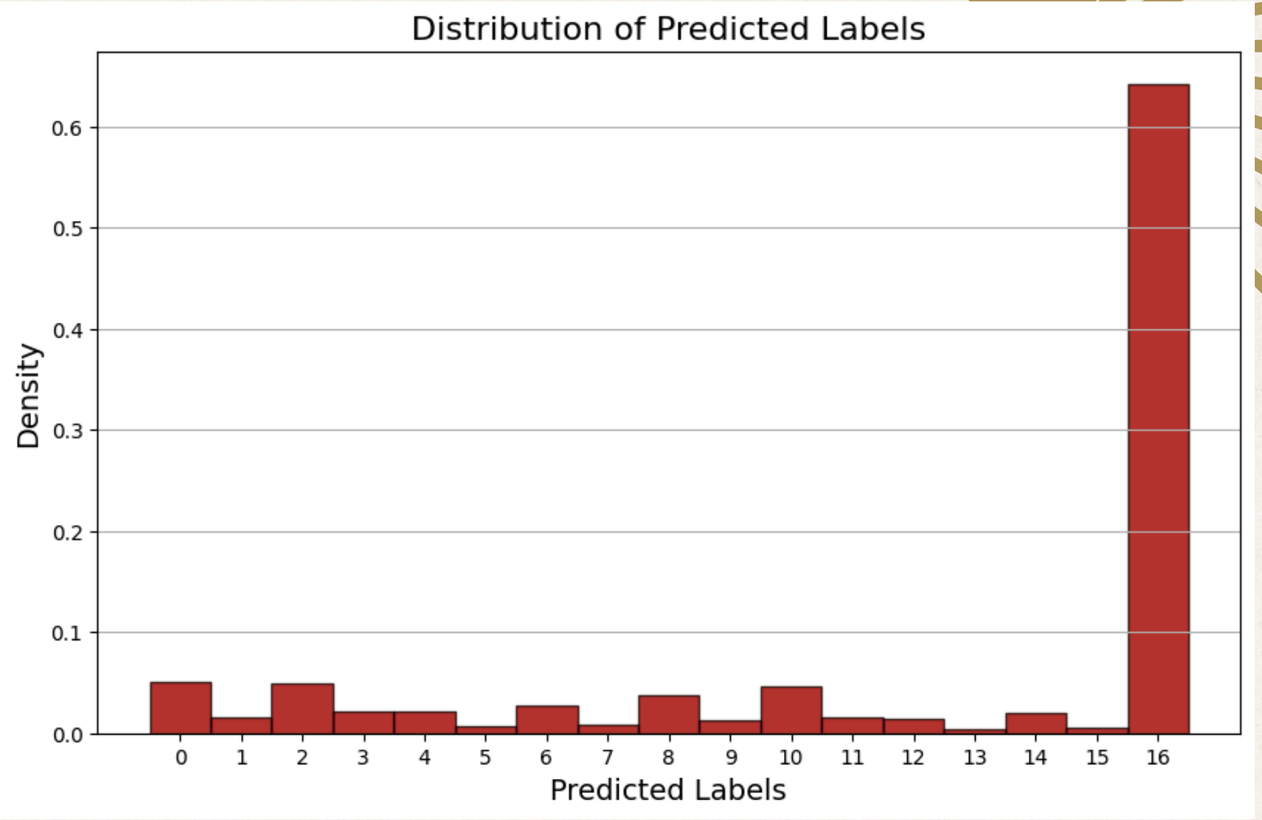


4 Qubit model with clustering labels (3 clusters)

1D Convolutional
combined with
ff layers

~300000 parameters

~95% Accuracy



Initial state vs clusters

Why do we see such a high improvement when using clustering?

- When looking at the 2 cluster result:
  - 2961/16000 (18.51%) are the same
- For 3 clusters:
  - 1324/16000 (8.28%) are the same

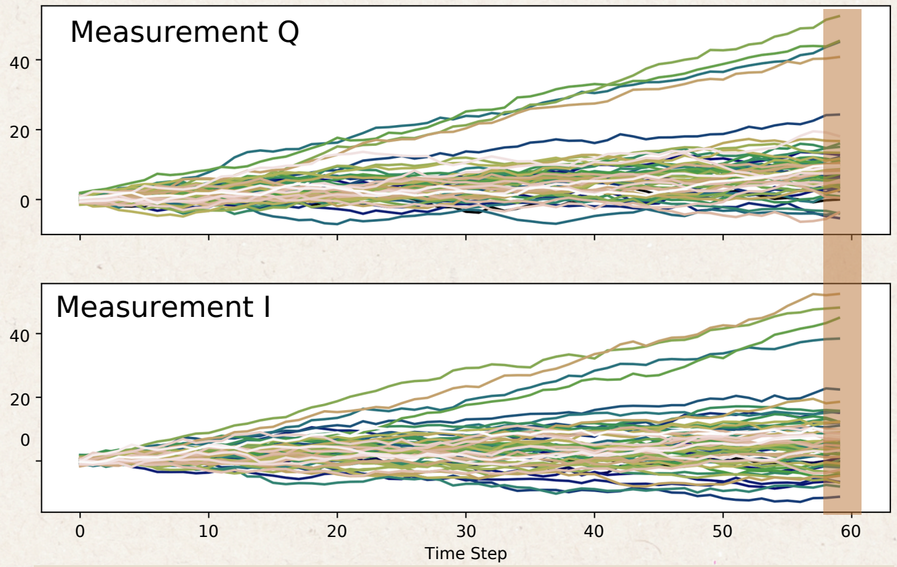
Comparing the initial state and the clustered readout

Initial:	Cluster:
$\|0000\rangle$	$\|2020\rangle$
$\|1010\rangle$	$\|1020\rangle$
$\|0101\rangle$	$\|1120\rangle$
$\|1111\rangle$	$\|2110\rangle$

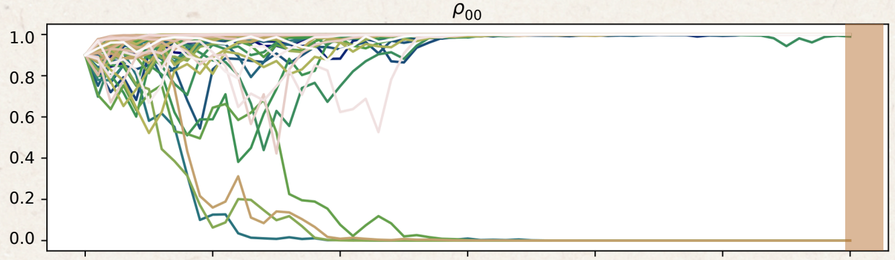
Given infinite time

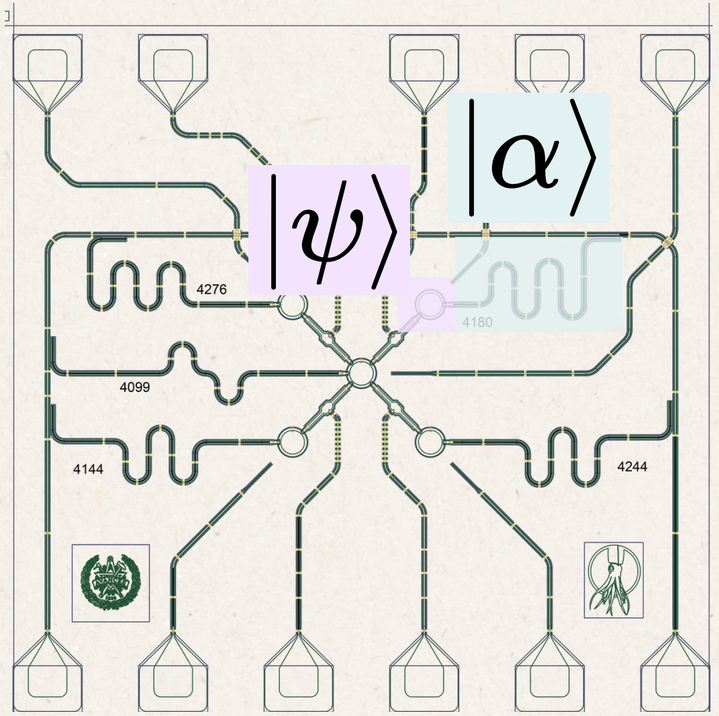
Single qubit classifier with simulated data

Input



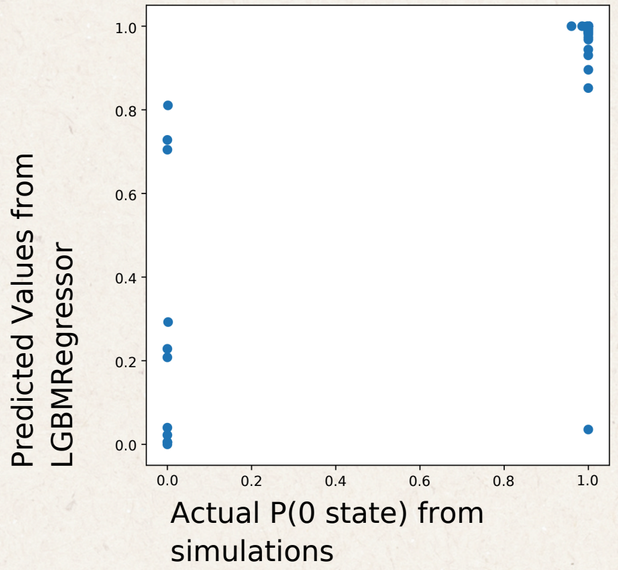
Output





Given infinite time..
Predict full quantum trajectory given all measurements!

Predicted vs. Actual Values for 50 measurements



Simulations

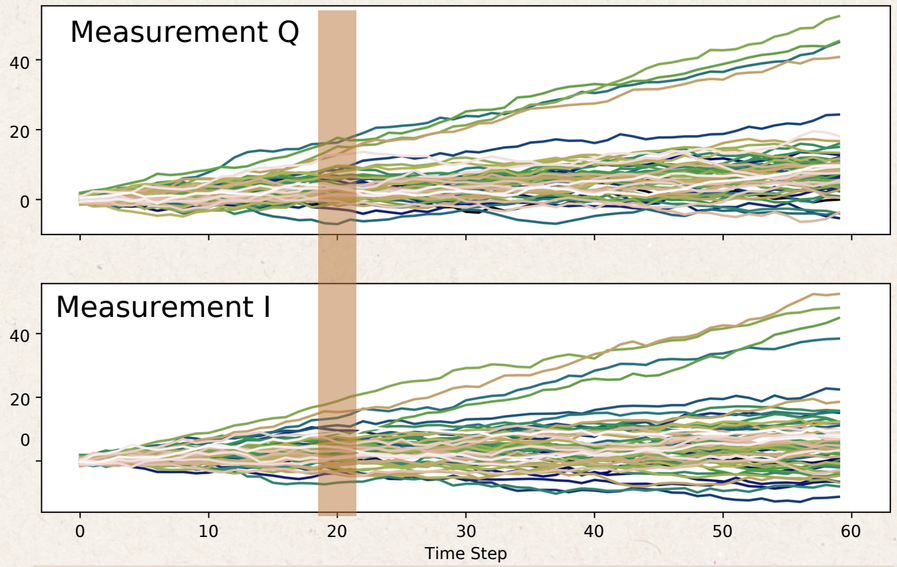
1. Entangle: $|\psi\rangle|\alpha\rangle \rightarrow e^{-i\chi a^\dagger a \sigma_z t}|\psi\rangle|\alpha\rangle$

2. Measure: $\rho_i = \Omega_i \rho \Omega_i^\dagger / P(i,\rho), i = Q, I$

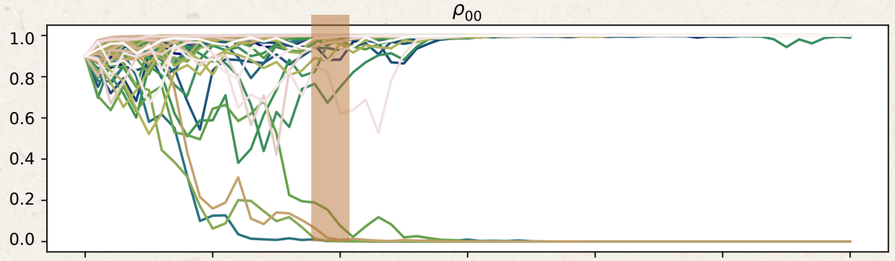
3. Update state: $\rho^{j+1} = (\rho_Q^j + \rho_I^j)/2$

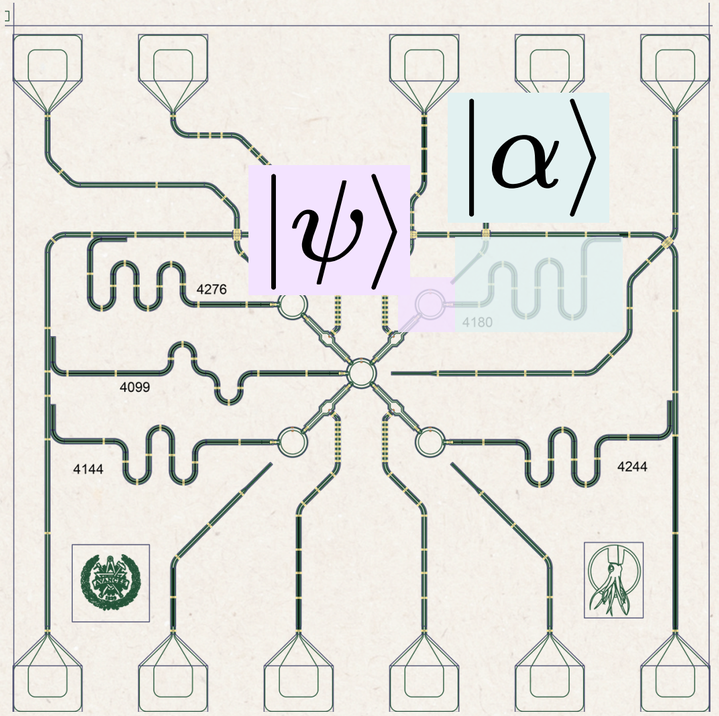
Single qubit classifier with simulated data

Input



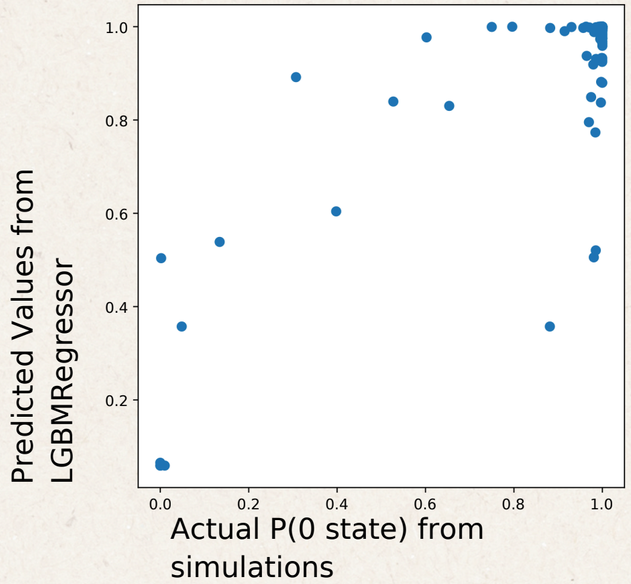
Output





Given infinite time..
Predict full quantum trajectory given all measurements!

Predicted vs. Actual Values for 20 measurements



Simulations

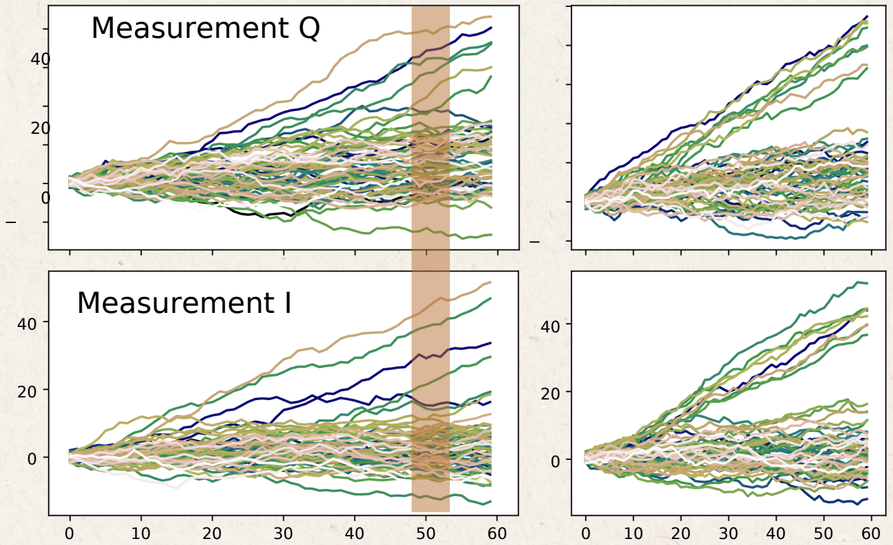
1. Entangle: $|\psi\rangle|\alpha\rangle \rightarrow e^{-i\chi a^\dagger a \sigma_z t}|\psi\rangle|\alpha\rangle$

2. Measure: $\rho_i = \Omega_i \rho \Omega_i^\dagger / P(i,\rho), i = Q, I$

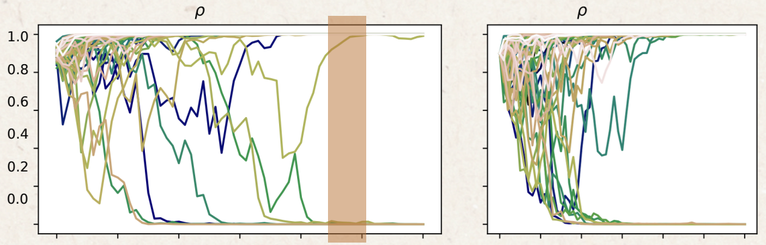
3. Update state: $\rho^{j+1} = (\rho_Q^j + \rho_I^j)/2$

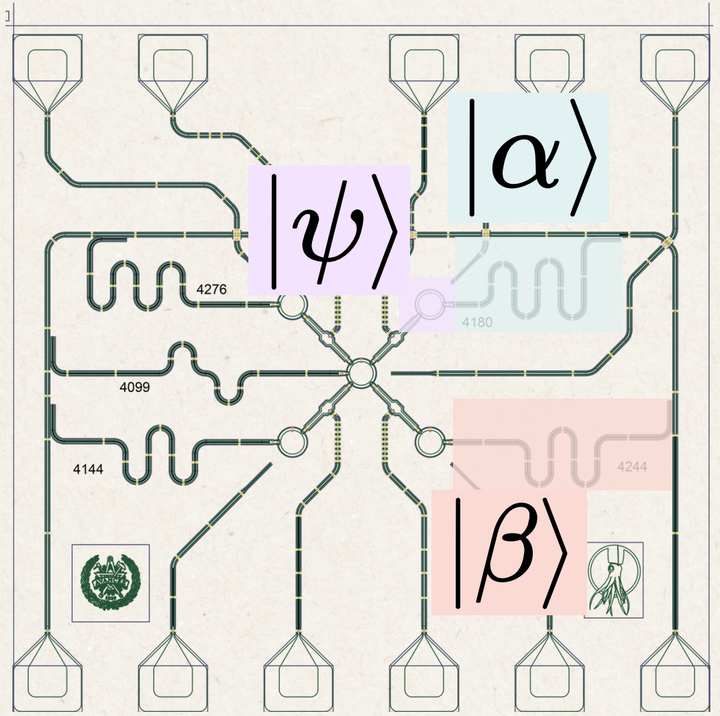
Single qubit classifier with simulated data including corelations!

Input



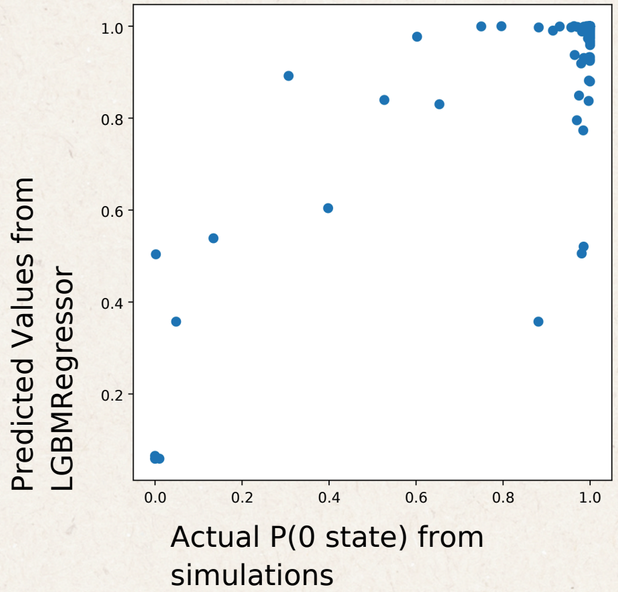
Output





Given infinite time..
Predict full quantum trajectory given all measurements!

Predicted vs. Actual Values for 50 measurements



Simulations

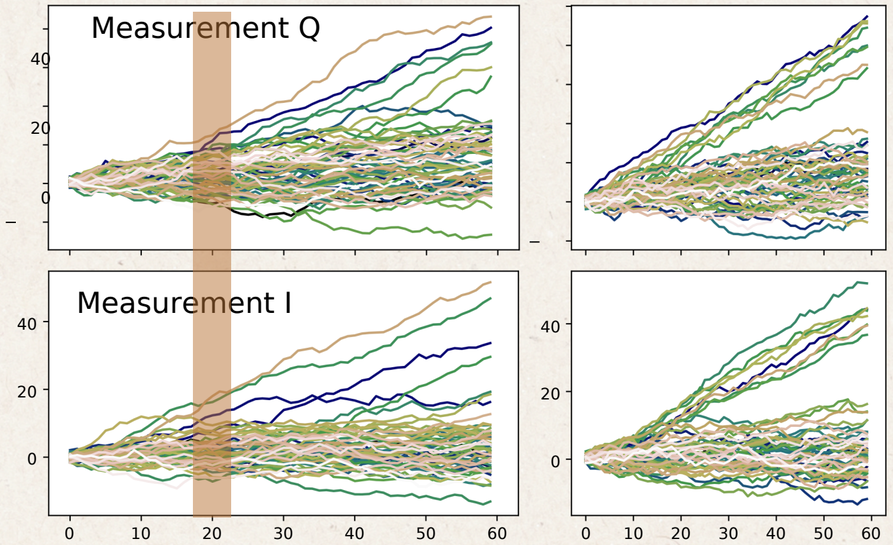
1. Entangle: $|\psi\rangle|\alpha\rangle|\beta\rangle \rightarrow e^{-i\chi a^\dagger a\sigma_z t}e^{-i\theta b^\dagger b\sigma_z t}|\psi\rangle|\alpha\rangle|\beta\rangle$

2. Measure: $\rho_i = \Omega_i\rho\Omega_i^\dagger/P(i,\rho), i = Q_1, I_1, Q_2, I_2,$

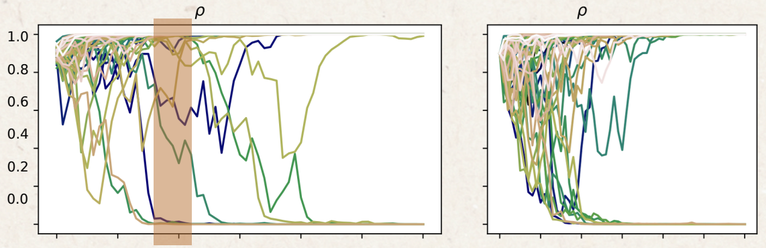
3. Update state: $\rho^{j+1} = (\rho^j_{Q_i} + \rho^j_{I_i})/2$

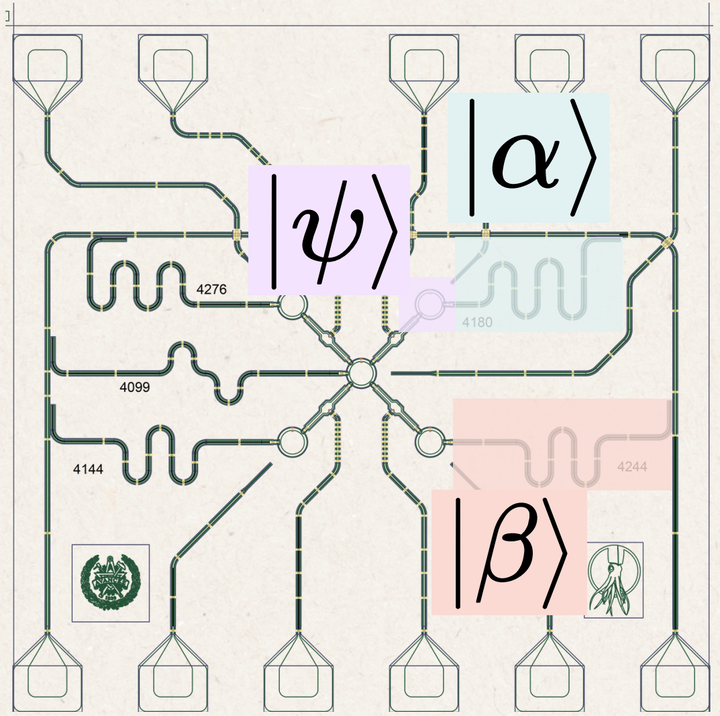
Single qubit classifier with simulated data including corelations!

Input



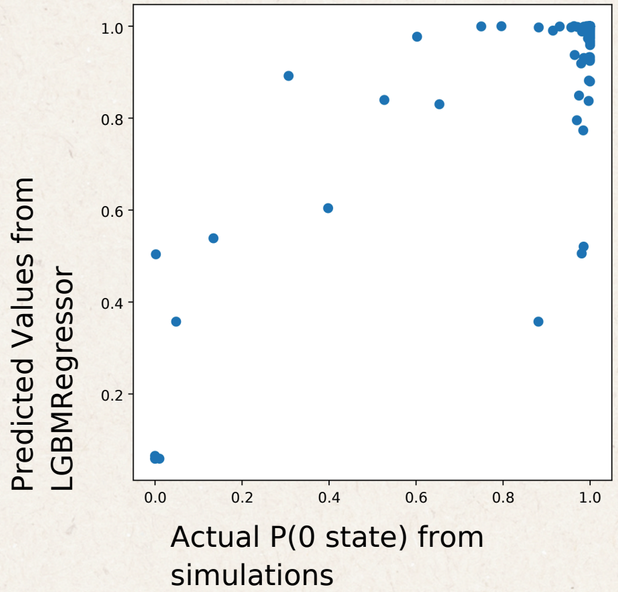
Output





Given infinite time..
Predict full quantum trajectory given all measurements!

Predicted vs. Actual Values for 20 measurements



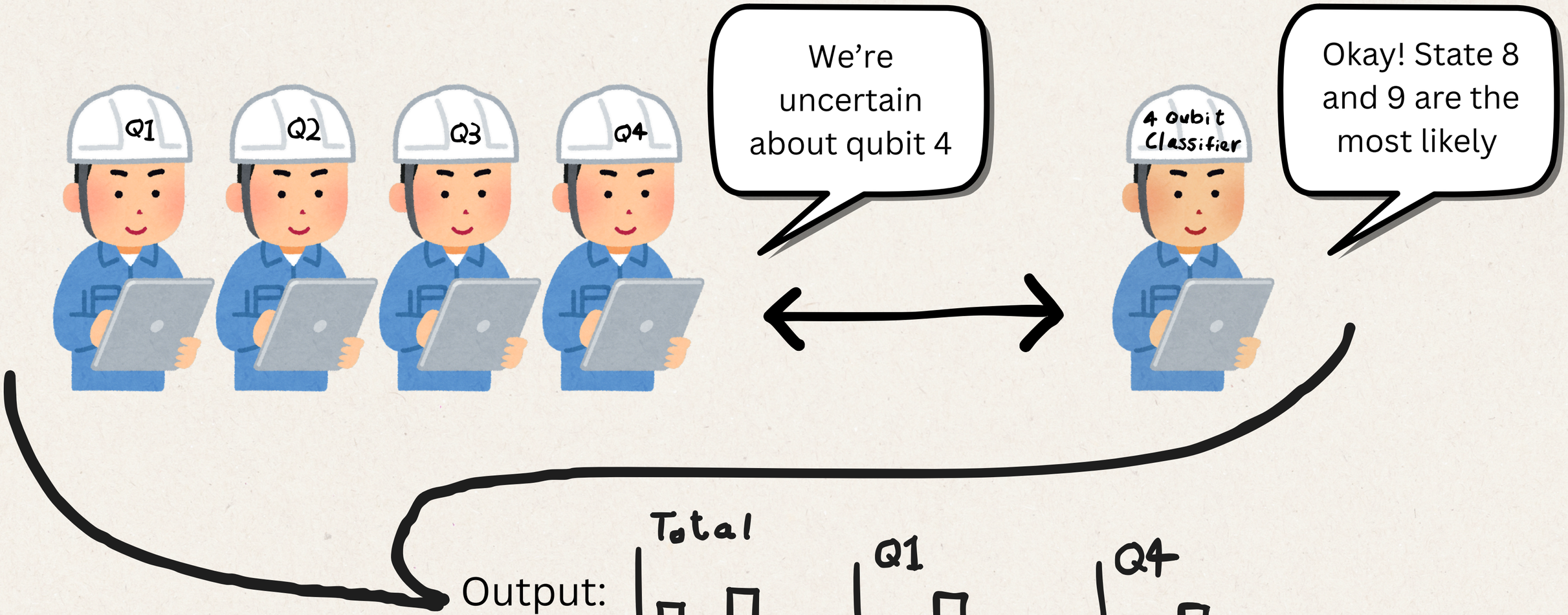
Simulations

1. Entangle: $|\psi\rangle|\alpha\rangle|\beta\rangle \rightarrow e^{-i\chi a^\dagger a\sigma_z t}e^{-i\theta b^\dagger b\sigma_z t}|\psi\rangle|\alpha\rangle|\beta\rangle$

2. Measure: $\rho_i = \Omega_i\rho\Omega_i^\dagger/P(i,\rho), i = Q_1, I_1, Q_2, I_2,$

3. Update state: $\rho^{j+1} = (\rho^j_{Q_i} + \rho^j_{I_i})/2$

Combined custom model with internal communication



What did we learn?

- Limited by accuracy of labels: our models will never be better than our labels
- The qubits are bad :c
- But we have many ideas for new models

Special thanks to

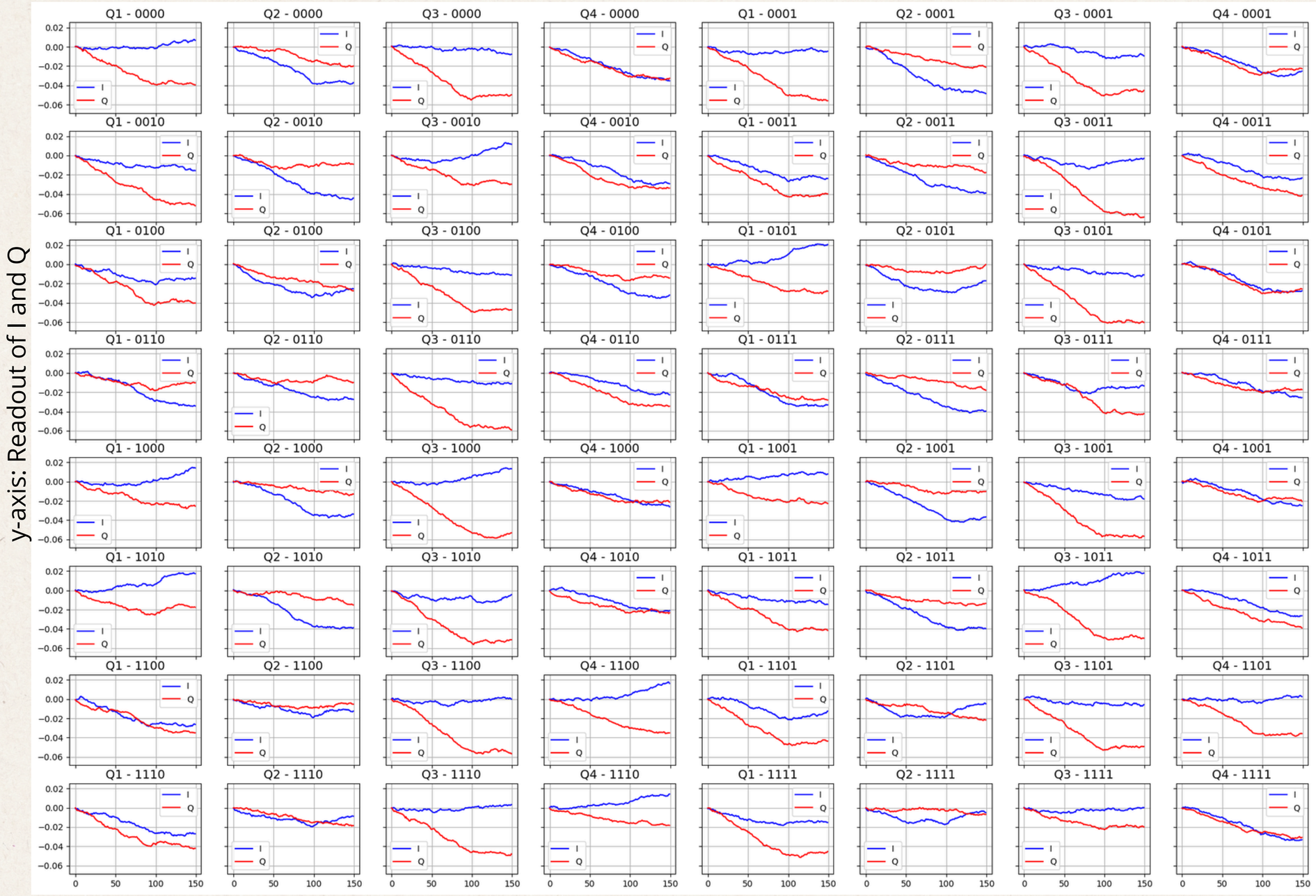
Helping us with experimental data! <3

SQUID Lab


Thank you!

Appendix

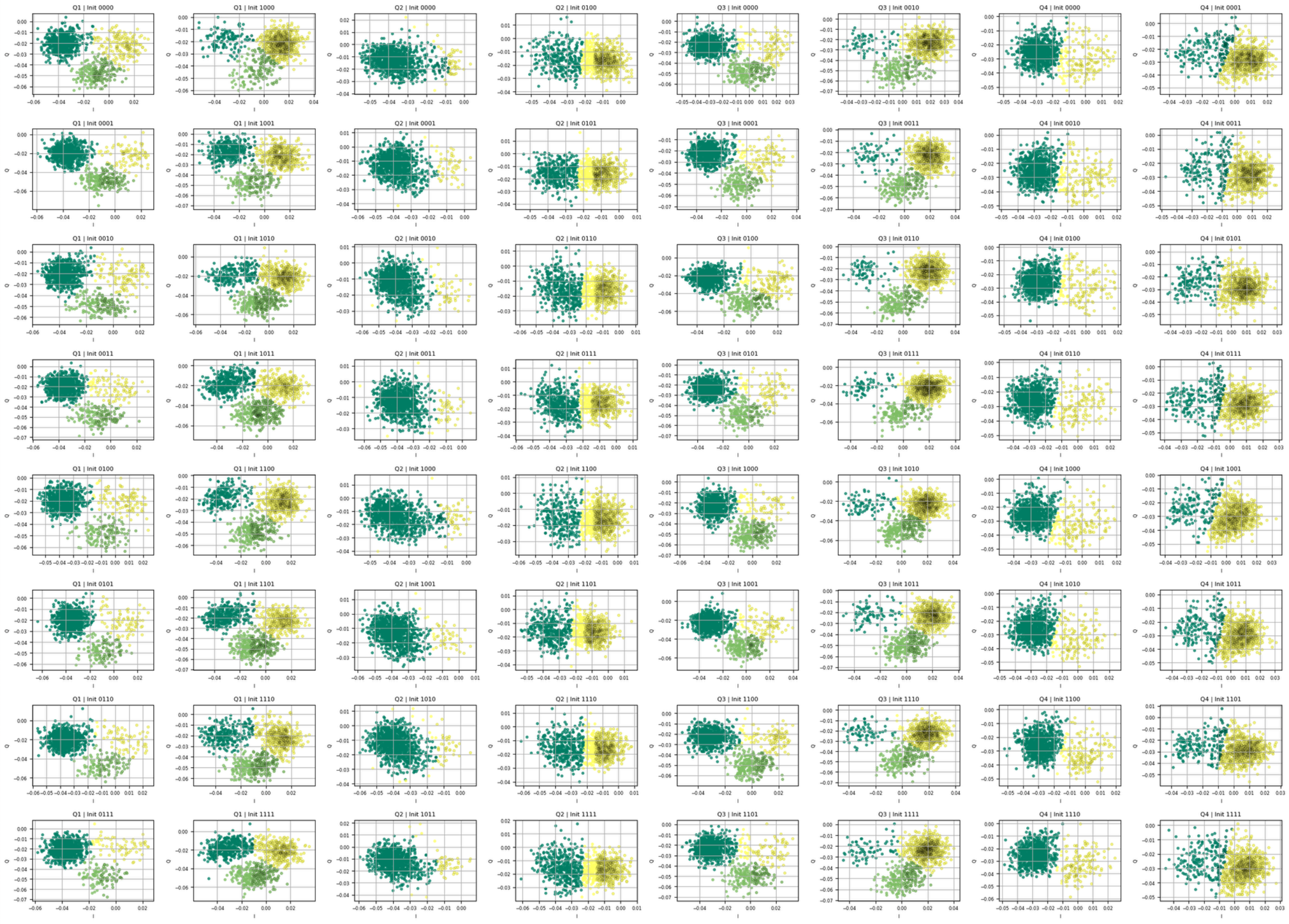
Data



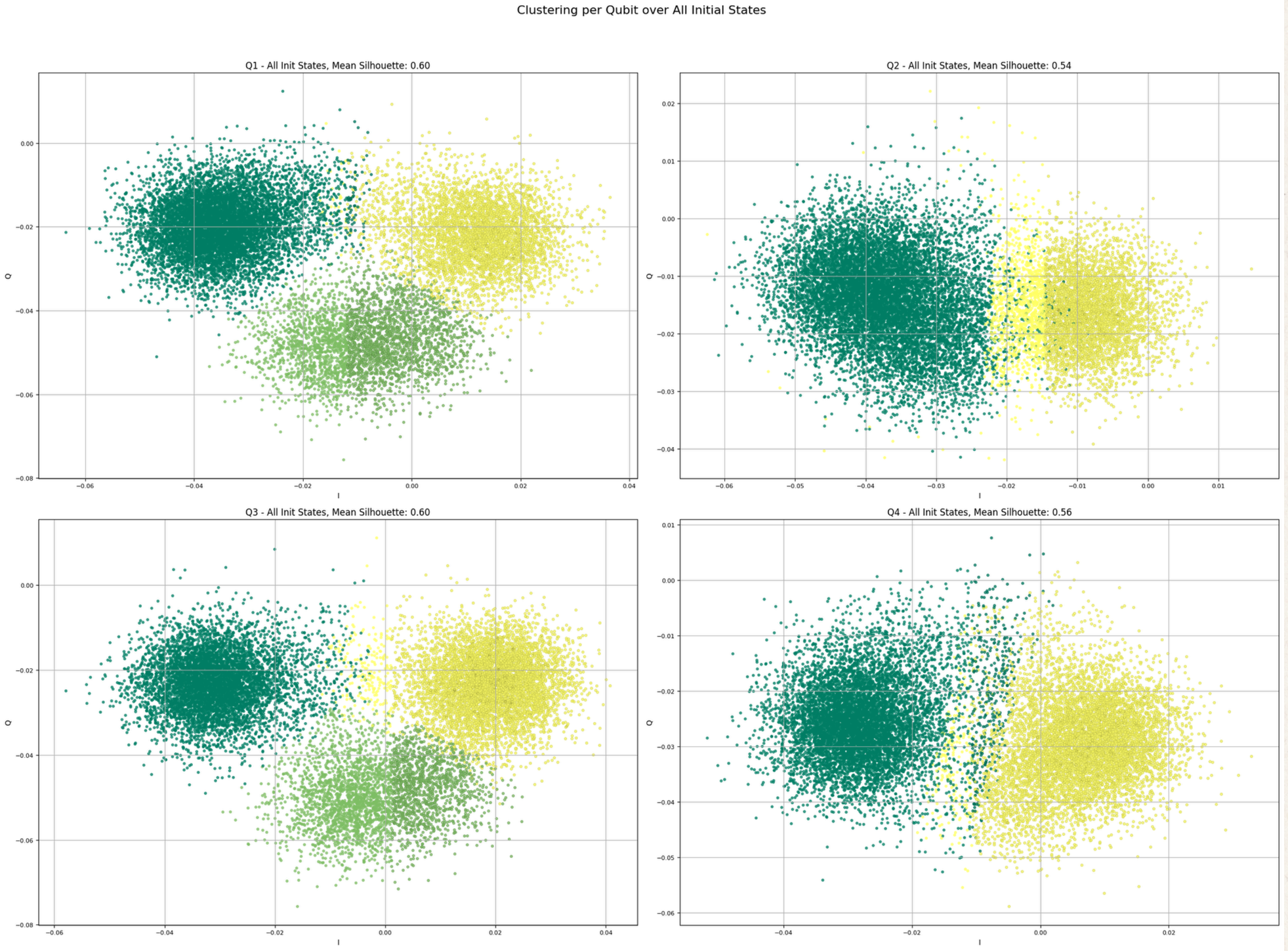
y-axis: Readout of I and Q

x - axis: Number of segments

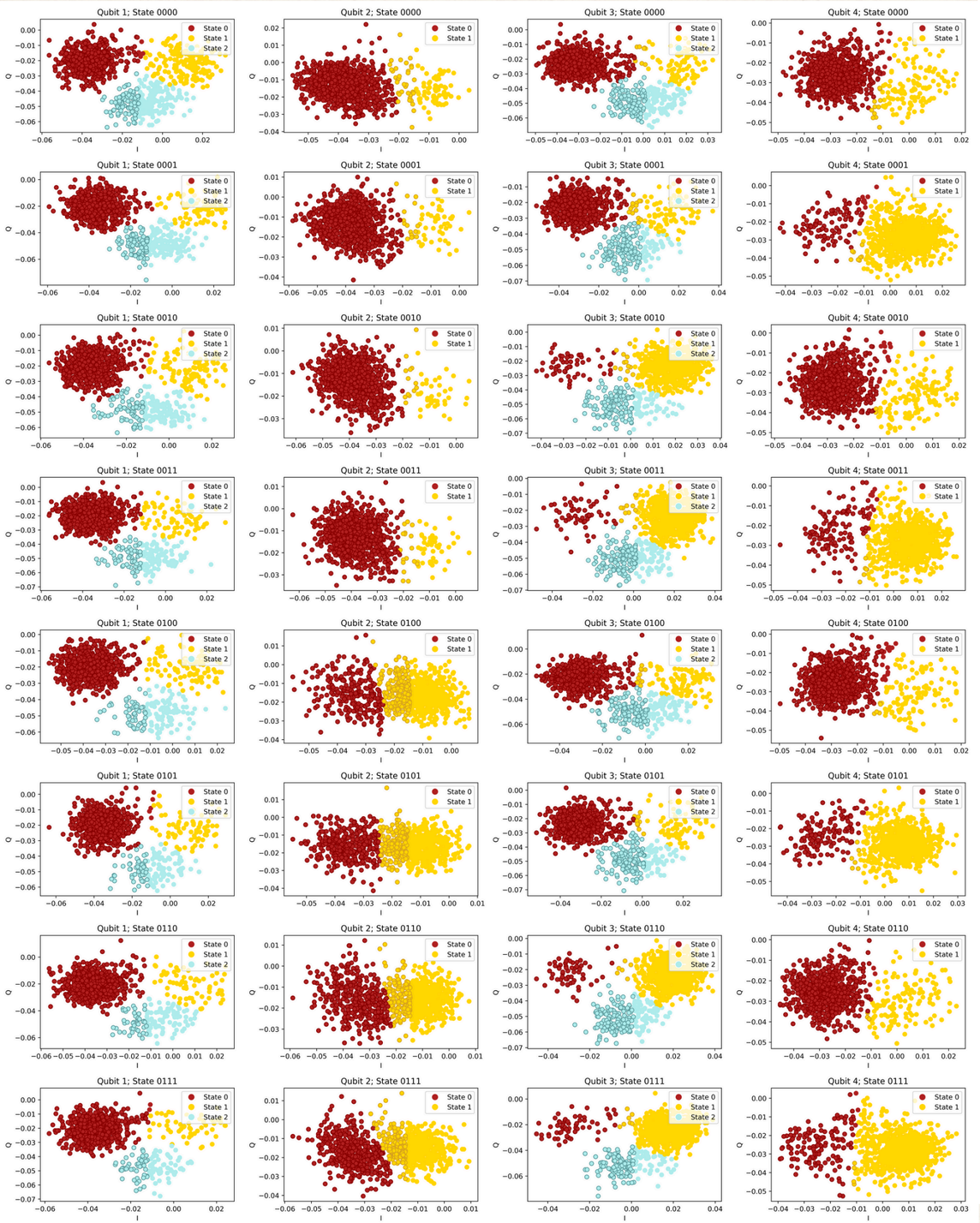
Clustering 1

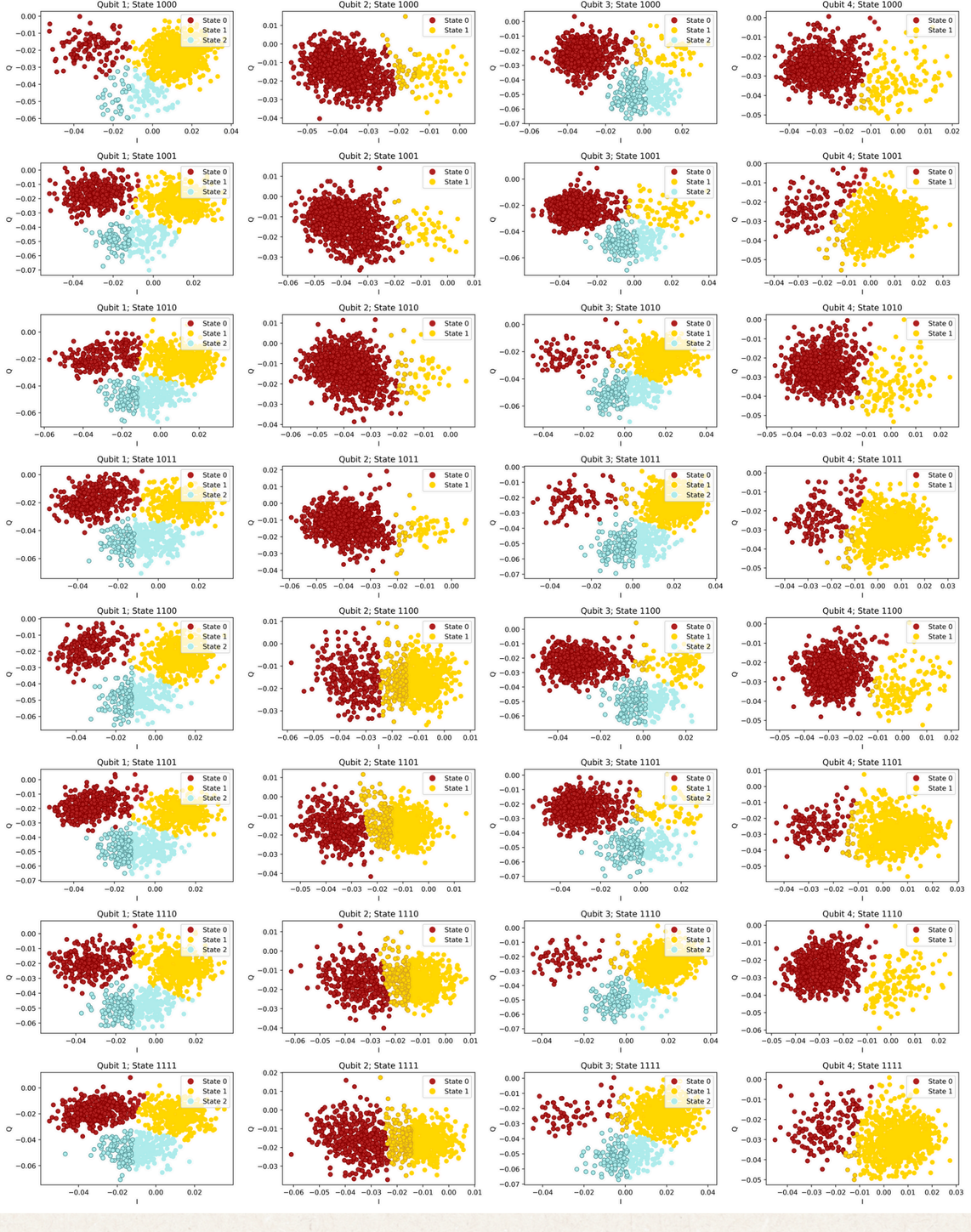


Clustering 1



Clustering 2

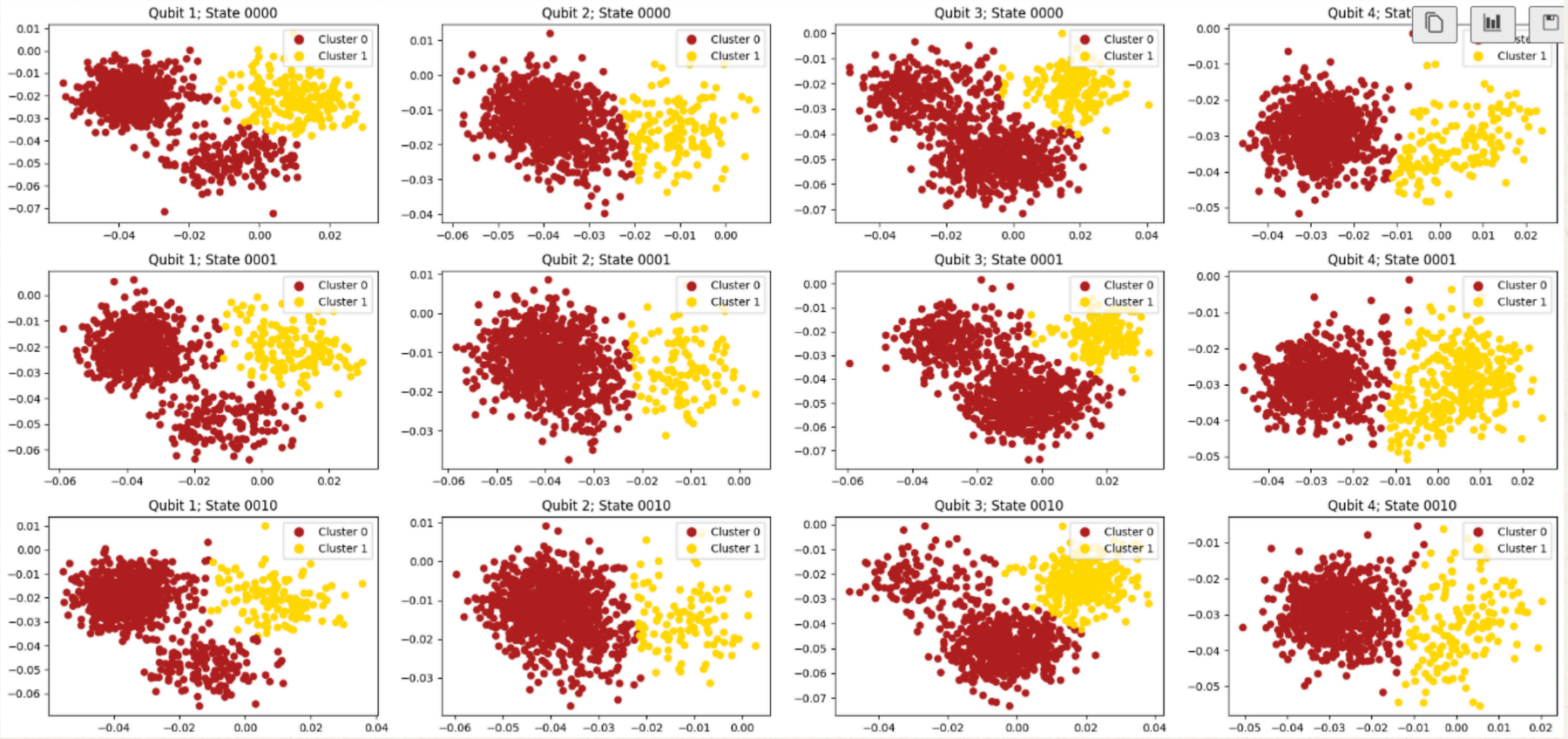




Clustering 2

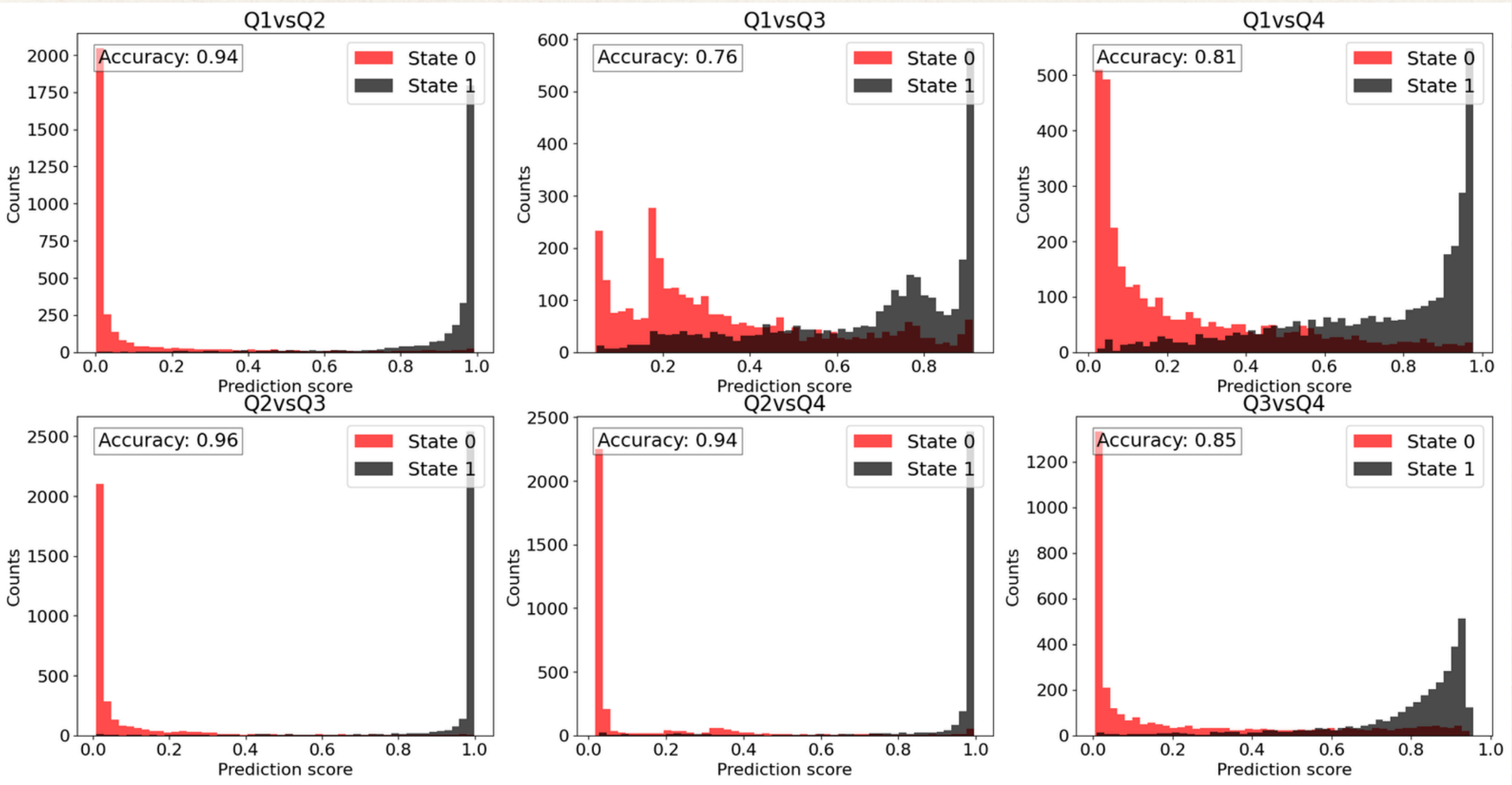
Example of the same clustering as before but now returning binary labels:

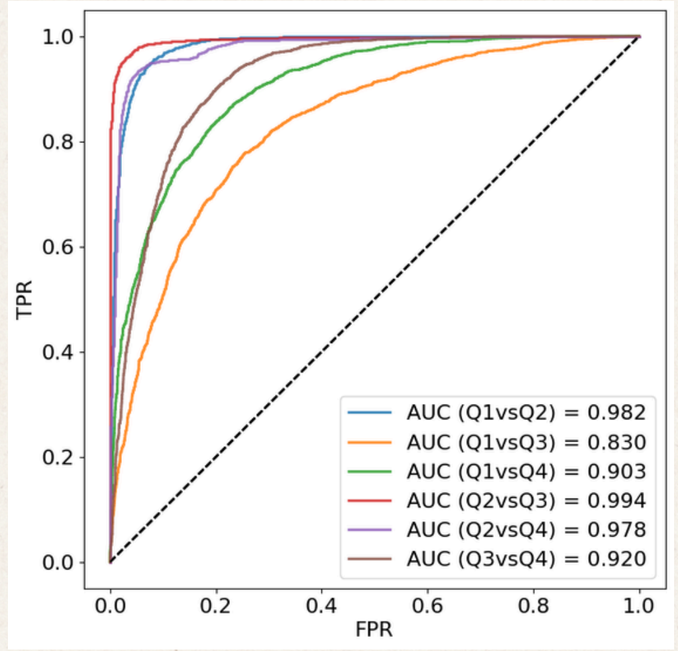
- Is the qubit in state 1 or not?



Distinguishing datasets with classifiers

In the very first classifiers we built, we noticed inconsistencies and didn't know if this was due to a wrong model or wrong data. A classifier were therefore build to check if ML models were capable of ditinguishing the different dataseries



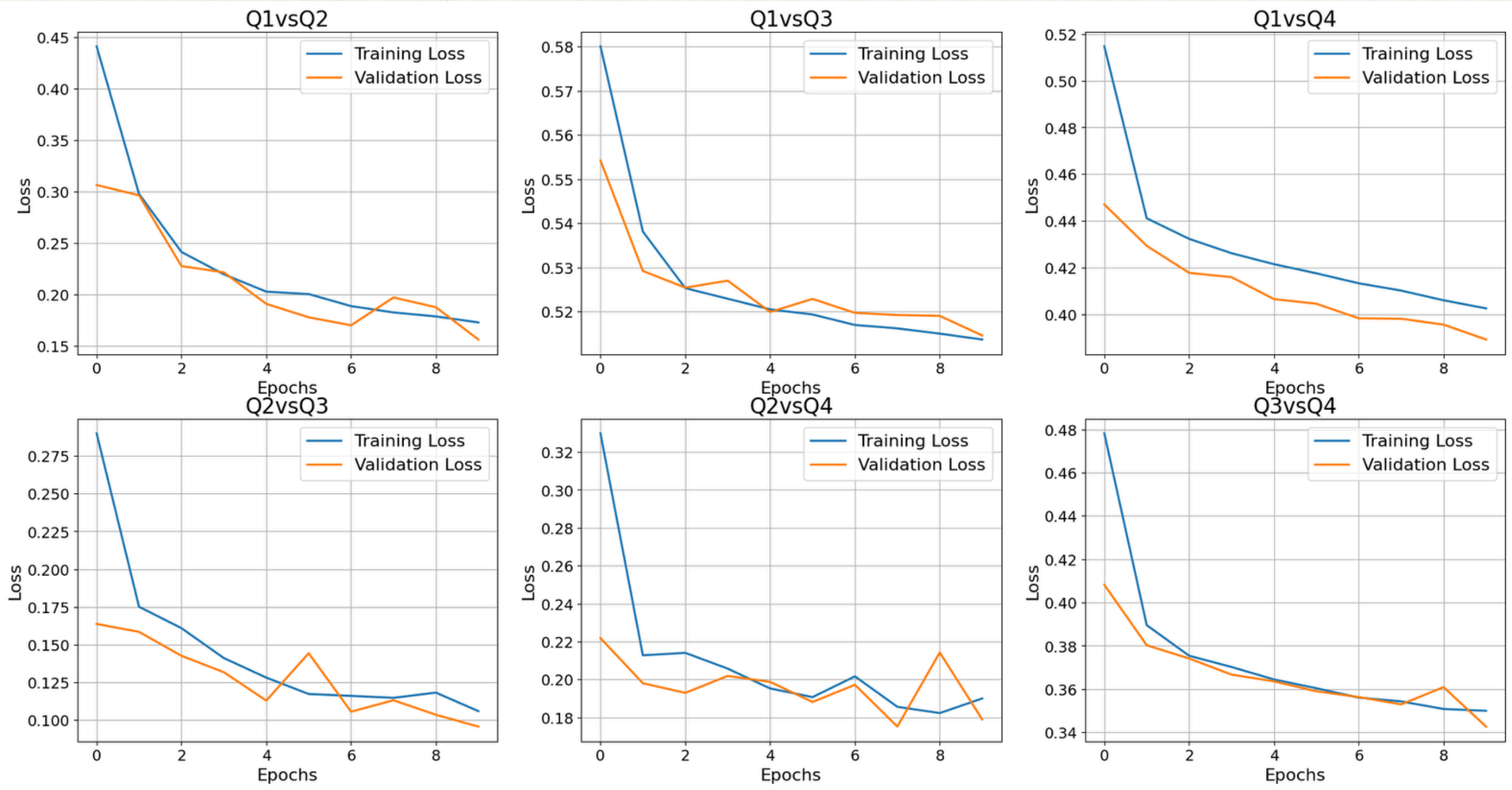


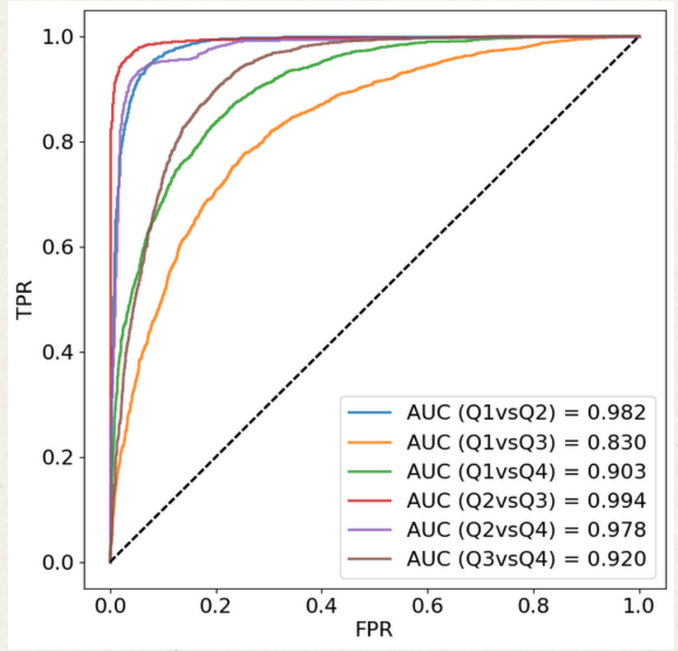
Conclusion

We indeed had some separation, meaning that the models could learn and the error was that we parsed the raw data and forgot to take the cumulative sum, which was a huge breakthrough, but not important enough to go into the main slides

Distinguishing datasets with classifiers

In the very first classifiers we built, we noticed inconsistencies and didn't know if this was due to a wrong model or wrong data. A classifier were therefore build to check if ML models were capable of ditinguishing the different dataseries





Conclusion

We indeed had some separation, meaning that the models could learn and the error was that we parsed the raw data and forgot to take the cumulative sum, which was a huge breakthrough, but not important enough to go into the main slides

1st Tensorflow Classifier

Initial state labels

HP - optimization (10 trials)

Qubit 1

Optimization time: 48m
Learning rate: 6.1E-3
n_layers: 2
n_units: 64, 64

Qubit 2

Optimization time: 37m
Learning rate: 8.6E-3
n_layers: 2
n_units: 32, 96

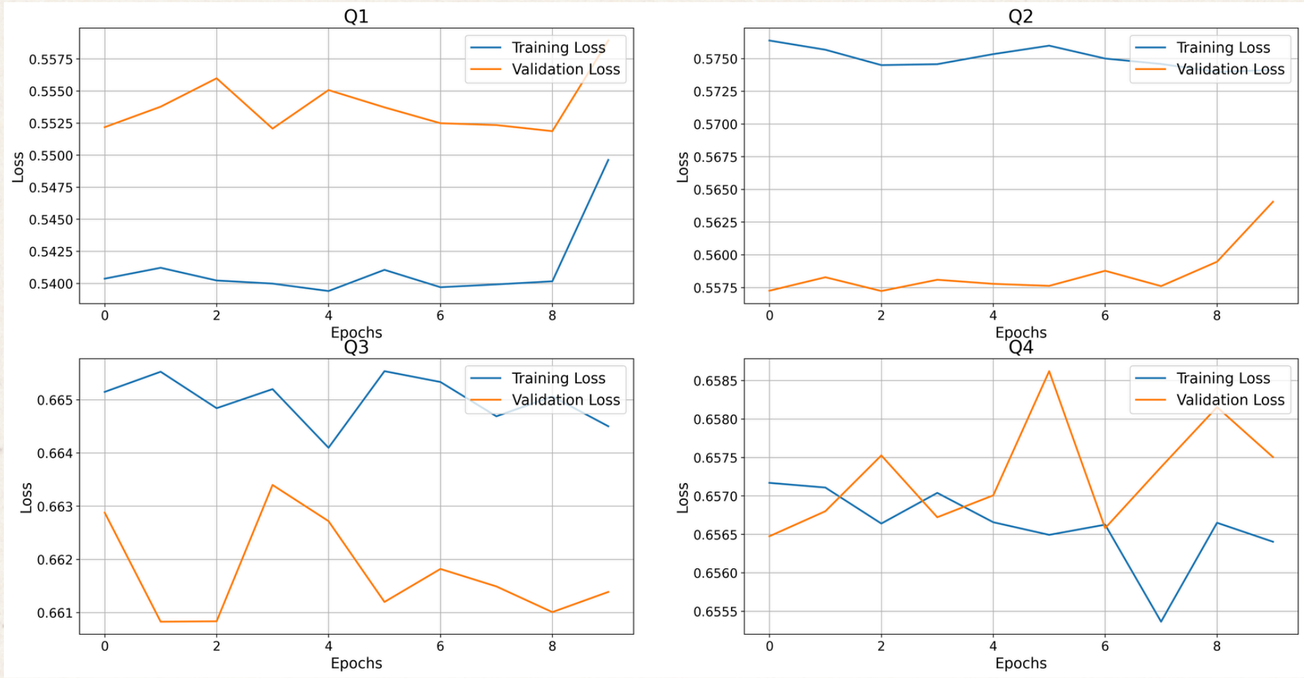
Qubit 3

Optimization time: 33m
Learning rate: 8.6E-3
n_layers: 2
n_units: 32, 96

Qubit 4

Optimization time: 43m
Learning rate: 8.6E-3
n_layers: 2
n_units: 32, 96

Training and validation loss



2nd Tensorflow Classifier

Binary Cluster labels

HP - optimization (10 trials)

Qubit 1

Optimization time: 48m
Learning rate: 6.1E-3
n_layers: 2
n_units: 64, 64

Qubit 2

Optimization time: 37m
Learning rate: 8.6E-3
n_layers: 2
n_units: 32, 96

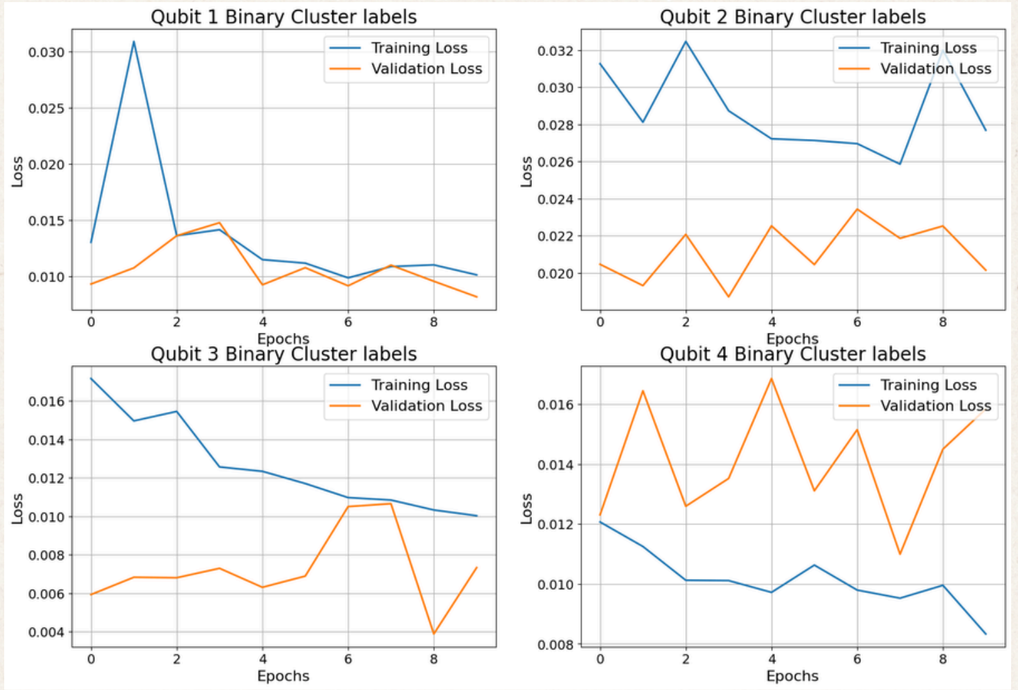
Qubit 3

Optimization time: 33m
Learning rate: 8.6E-3
n_layers: 2
n_units: 32, 96

Qubit 4

Optimization time: 43m
Learning rate: 8.6E-3
n_layers: 2
n_units: 32, 96

Training and validation loss



3rd Tensorflow Classifier

3 State Cluster labels

HP - optimization (10 trials)

- Only qubit 1 and 3 since these were the only qubits with 3 states

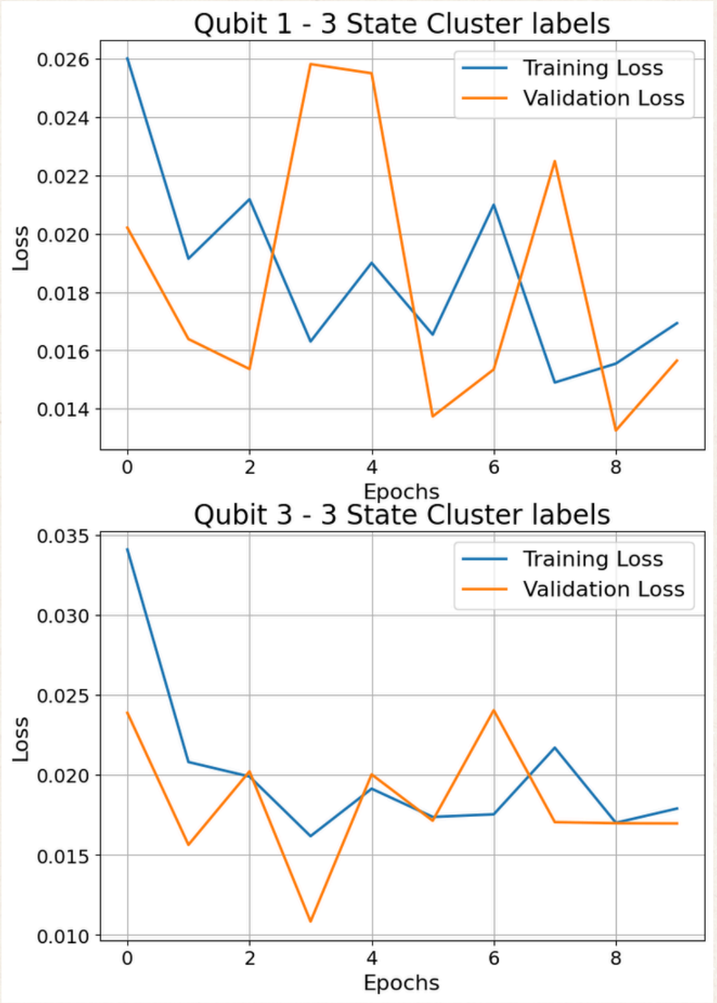
Qubit 1

Optimization time: 1h 26m
Learning rate: 8.6E-3
n_layers: 2
n_units: 32, 96

Qubit 3

Optimization time: 1h 33m
Learning rate: 8.6E-3
n_layers: 2
n_units: 32, 96

Training and validation loss



4 Qubit classifier
3 Clusters

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 150, 96)	1,632
max_pooling1d (MaxPooling1D)	(None, 75, 96)	0
flatten (Flatten)	(None, 7200)	0
dense (Dense)	(None, 32)	230,432
dense_1 (Dense)	(None, 224)	7,392
dense_2 (Dense)	(None, 256)	57,600
dense_3 (Dense)	(None, 17)	4,369

Total params: 301,425 (1.15 MB)

Trainable params: 301,425 (1.15 MB)

Non-trainable params: 0 (0.00 B)

4 Qubit classifier
2 Clusters

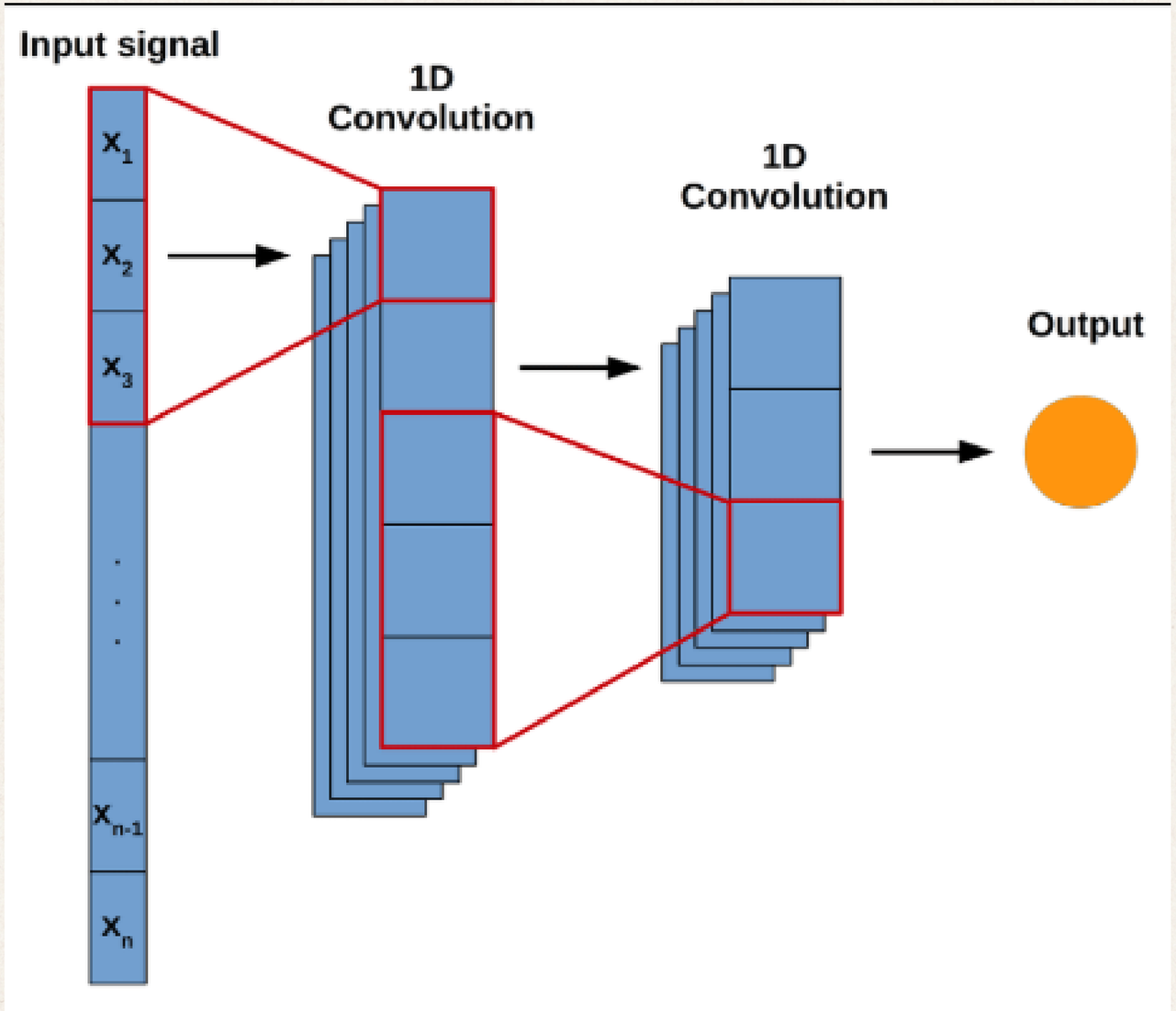
Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 150, 80)	3,280
max_pooling1d (MaxPooling1D)	(None, 75, 80)	0
conv1d_1 (Conv1D)	(None, 75, 128)	41,088
max_pooling1d_1 (MaxPooling1D)	(None, 37, 128)	0
conv1d_2 (Conv1D)	(None, 37, 128)	49,280
max_pooling1d_2 (MaxPooling1D)	(None, 18, 128)	0
flatten (Flatten)	(None, 2304)	0
dense (Dense)	(None, 192)	442,560
dense_1 (Dense)	(None, 16)	3,088

Total params: 539,296 (2.06 MB)

Trainable params: 539,296 (2.06 MB)

Non-trainable params: 0 (0.00 B)

1D Convolution



Taken from medium.com

1 qubit classifier using simulations with correlations with another resonator

A big issue we encountered using Machine learning for classifying the qubit state after readout is that we do not have any faithful labels. This makes it hard to train the model. It is however possible to simulate the system such that the state is known. This is an example of how to use simulations rather than experimental data.

Simulations

$$|\psi\rangle|\alpha\rangle \rightarrow e^{-i\chi a^\dagger a\sigma_z t}|\psi\rangle|\alpha\rangle = a|0\rangle|\alpha_0\rangle + b|0\rangle|\alpha_1\rangle$$

$$\hat{\Omega}_q = \langle q|e^{-ia^\dagger a\sigma_z t}|\alpha\rangle$$

$$P(q|\psi) = |\langle q|\alpha_0\rangle|^2 + |\langle q|\alpha_1\rangle|^2$$

$$|\psi_{i+1}^q\rangle = \frac{\hat{\Omega}_q|\psi_i\rangle}{\sqrt{P(q|\psi_i)}}$$

$$\rho_{j+1} = \frac{1}{2}(|\psi_{i+1}^q\rangle\langle\psi_{i+1}^q| + |\psi_{i+1}^I\rangle\langle\psi_{i+1}^I|)$$

$$|\psi\rangle|\alpha\rangle|\beta\rangle \rightarrow e^{-i\chi a^\dagger a\sigma_z t}e^{-i\theta b^\dagger b\sigma_z t}|\psi\rangle|\alpha\rangle|\beta\rangle$$
$$= a|0\rangle|\alpha_0\rangle|\beta_0\rangle + b|0\rangle|\alpha_1\rangle|\beta_1\rangle$$