Workshop on Quantum Machine Learning

Olivier Nahman-Lévesque
olivier.nahman-levesque@usherbrooke.ca
Nov. 25 2025

UNIVERSITÉ DE SHERBROOKE

Outline

1. Introduction
2. Our toy problem
3. Classification in machine learning
4. Quantum refresher
5. Data embedding in quantum feature space
6. Quantum measurement and the quantum classifier
7. Training a quantum classifier

Problem sets on Colab

Jupyter Notebooks

Create a google account to have access to Colab!

Institut quantique

Cooperative R&D environnement for academia and industry

Quantum tech platforms

Quantum Fab Lab

Quantum computation platform

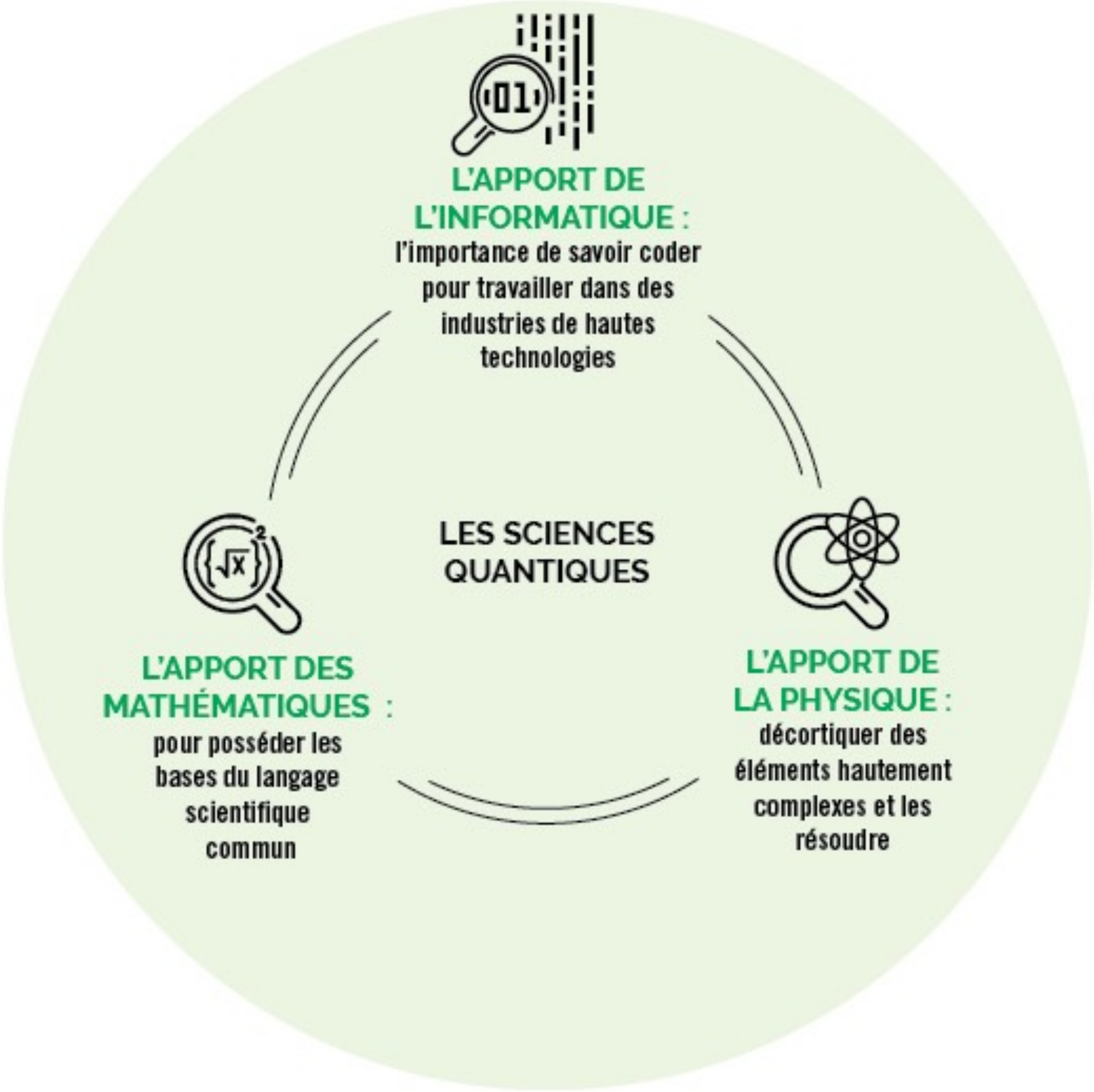
New BSc in quantum sciences

First BSc in quantum sciences in Quebec

Offered by the University of Sherbrooke
Starting in the fall semester of 2022

For more info: https://bit.ly/3KXxpHx

Université de Sherbrooke



THE QUANTUM ENIGMAS

UdS Université de Sherbrooke

INSTITUT QUANTIQUE
UNIVERSITÉ DE SHERBROOKE

0 à 1

Summary School in quantum information

5 to 9 May 2025

One week of courses and hands-on sessions to expand your knowledge of quantum computing

- Introductory courses in quantum programming with Qiskit
- Visit of Sherbrooke's quantum ecosystem
- Registration fee 150$ including accommodation and meals

Who is it for

Open to francophone undergraduate students in science (physics, maths, computer science, engineering, chemistry…)

Deadline to submit your application : 17 January 2025

Scan the QR code for details and application

INSTITUT QUANTIQUE UdS

ALGOLAB QUANTIQUE

QUANTACET

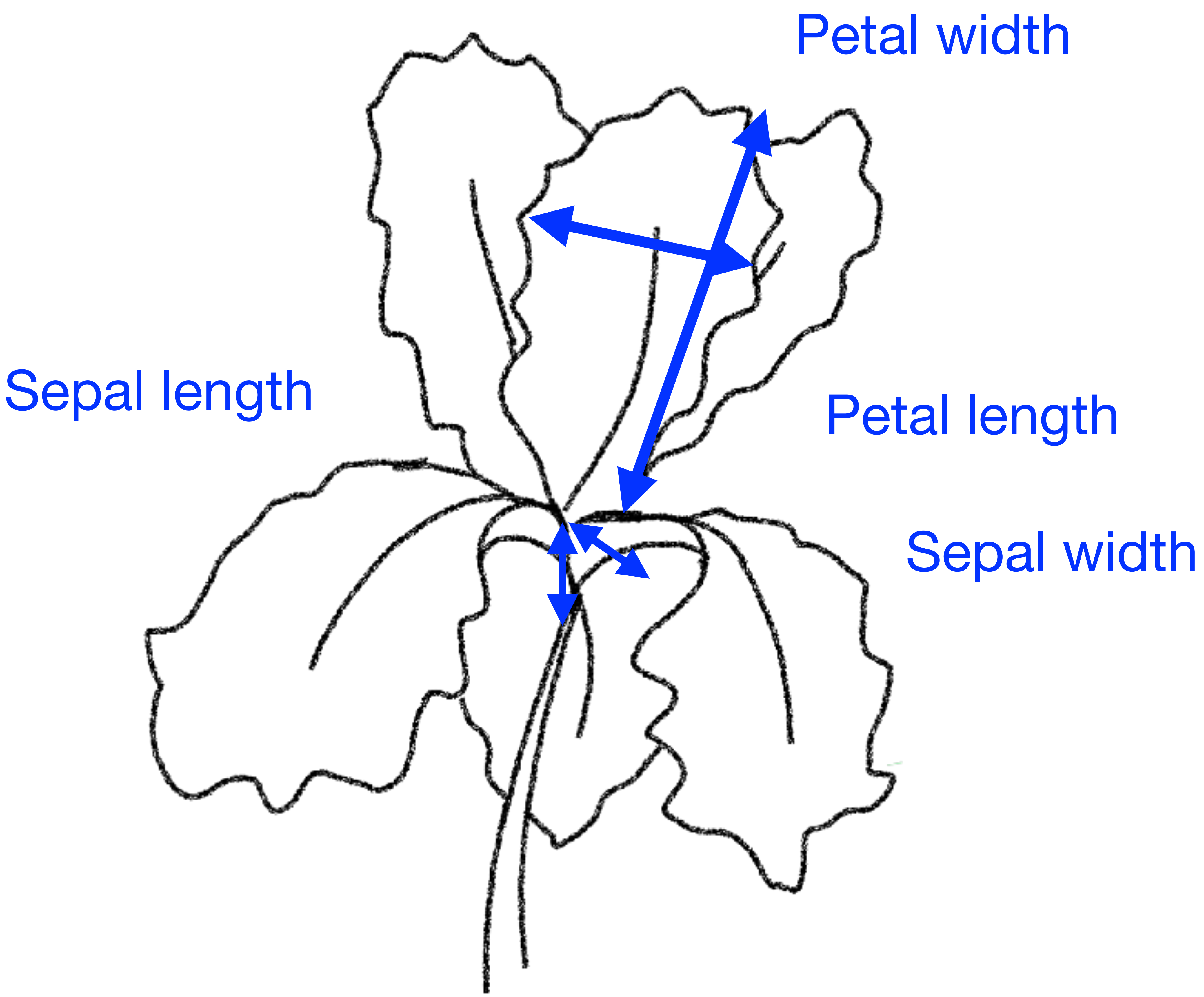
Nord Quantique

PASQAL

Our toy problem

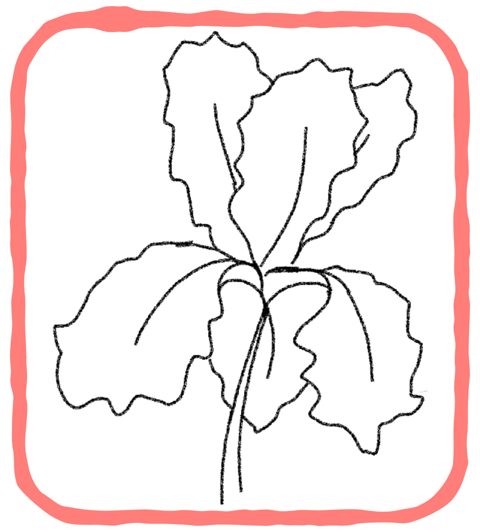
Iris

Our toy problem
Iris dataset

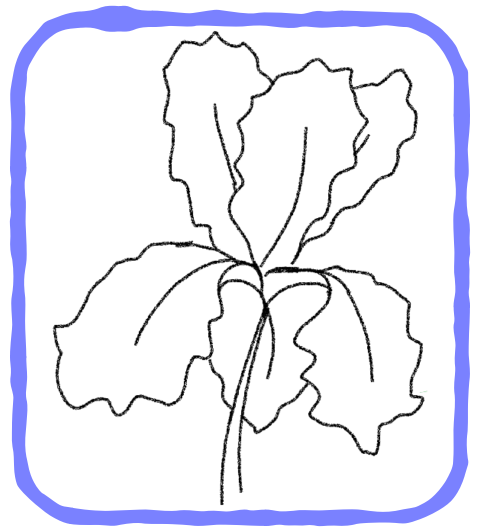


$$\vec{x} = [5.1, 3.5, 1.4, 0.2]$$

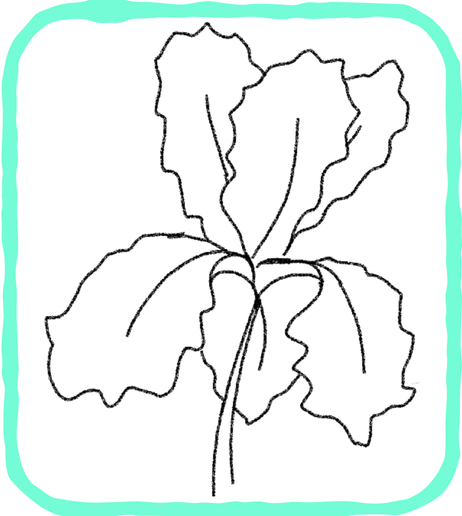
Three classes



Setosa



Versicolour

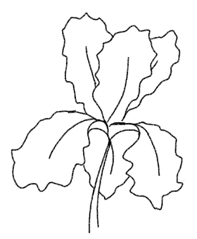


Virginica

Our toy problem

Dataset for binary classification

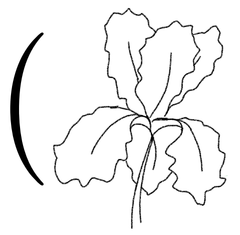
- Domain set

$\mathcal{X} = \{$  $\} = \{\vec{x}_i\}$ for example $\vec{x}_0 = [5.1, 3.5, 1.4, 0.2]$

- Label set

$$\mathcal{Y} = \{\square, \square\} = \{0, 1\}$$

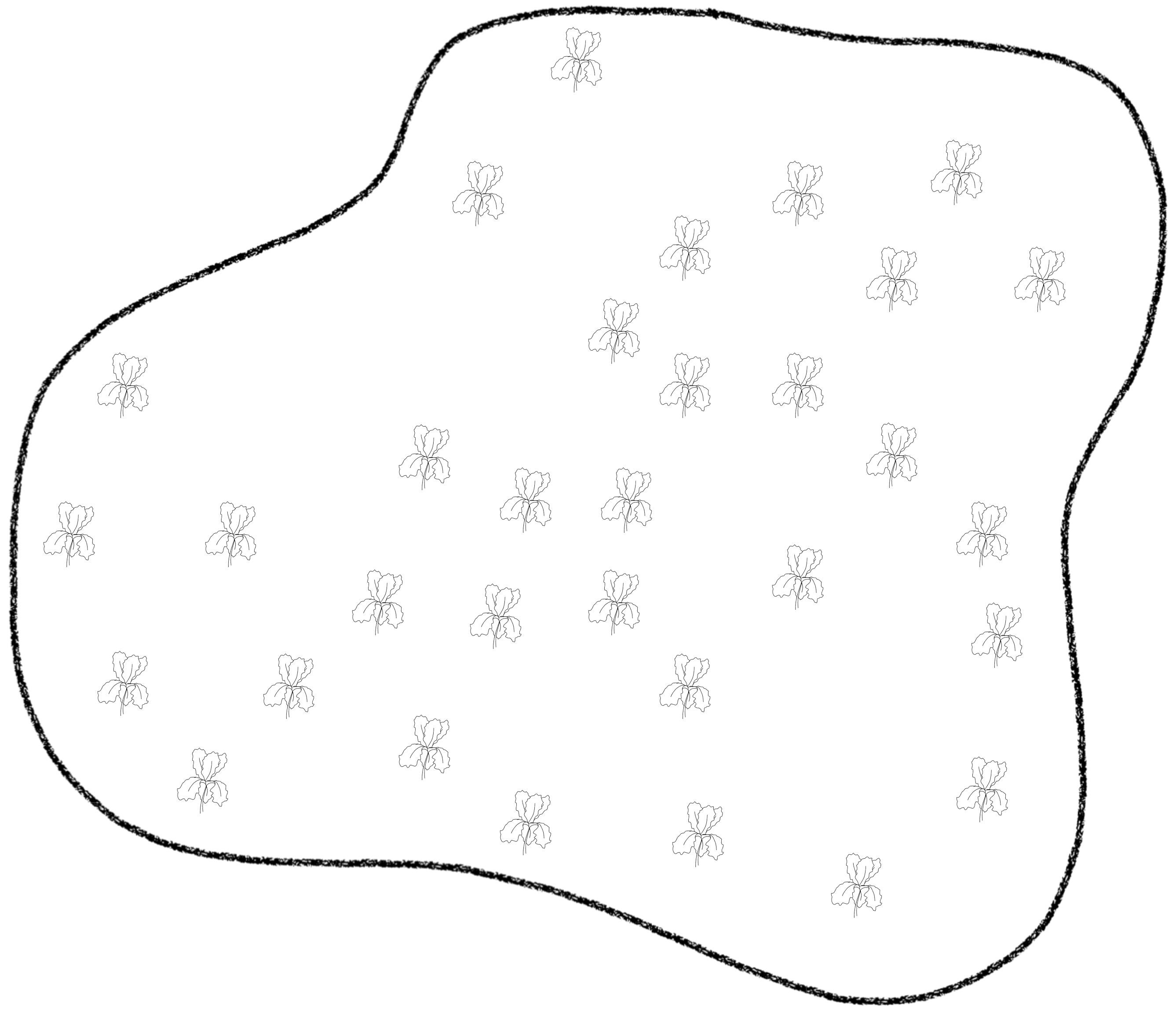
- Unknown, true labeling function

$f($  $) = \begin{cases} \square \\ \square \end{cases}$

Our goal is to learn the true labeling function f !

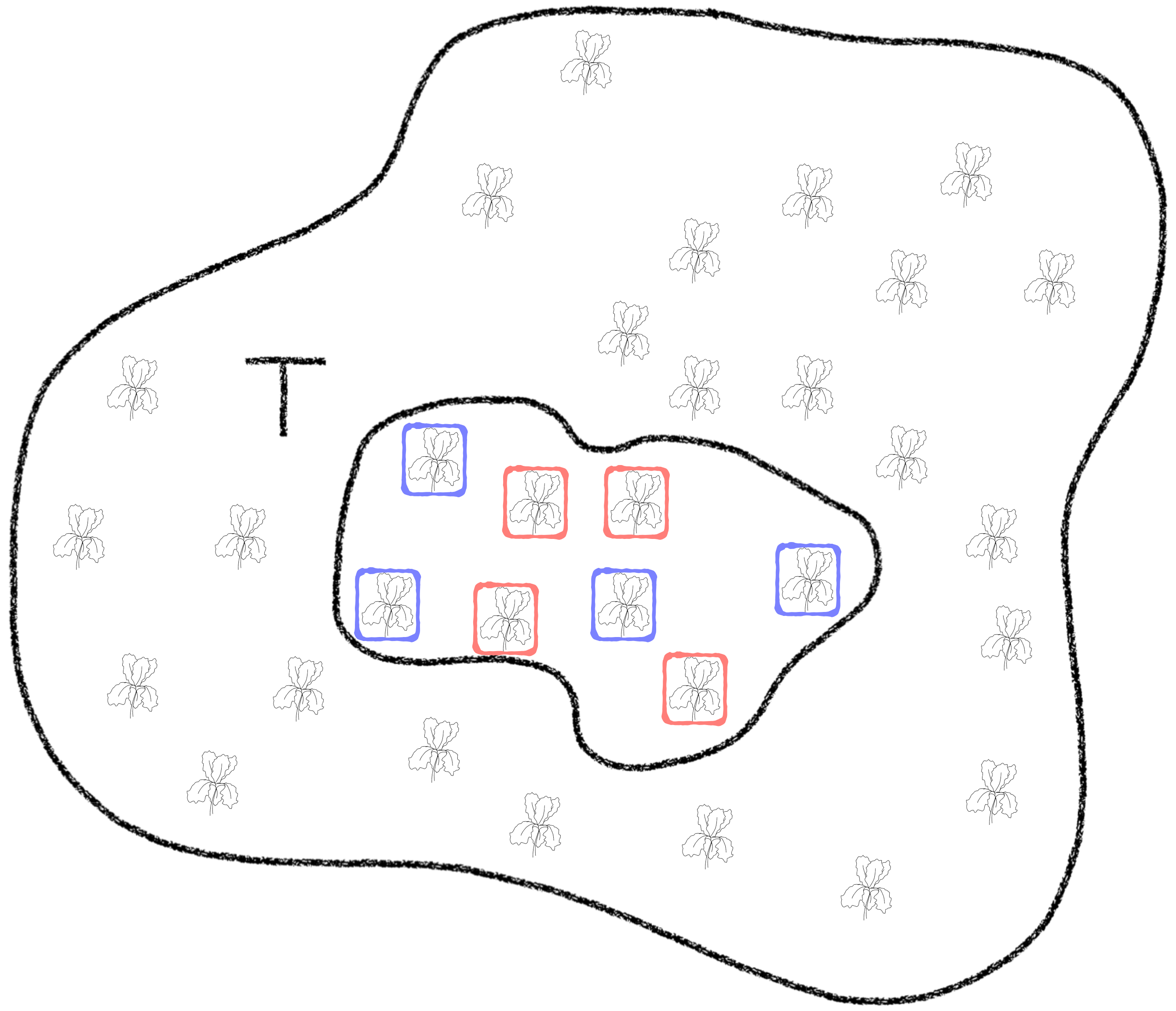
Our toy problem

Train and test set for supervised learning



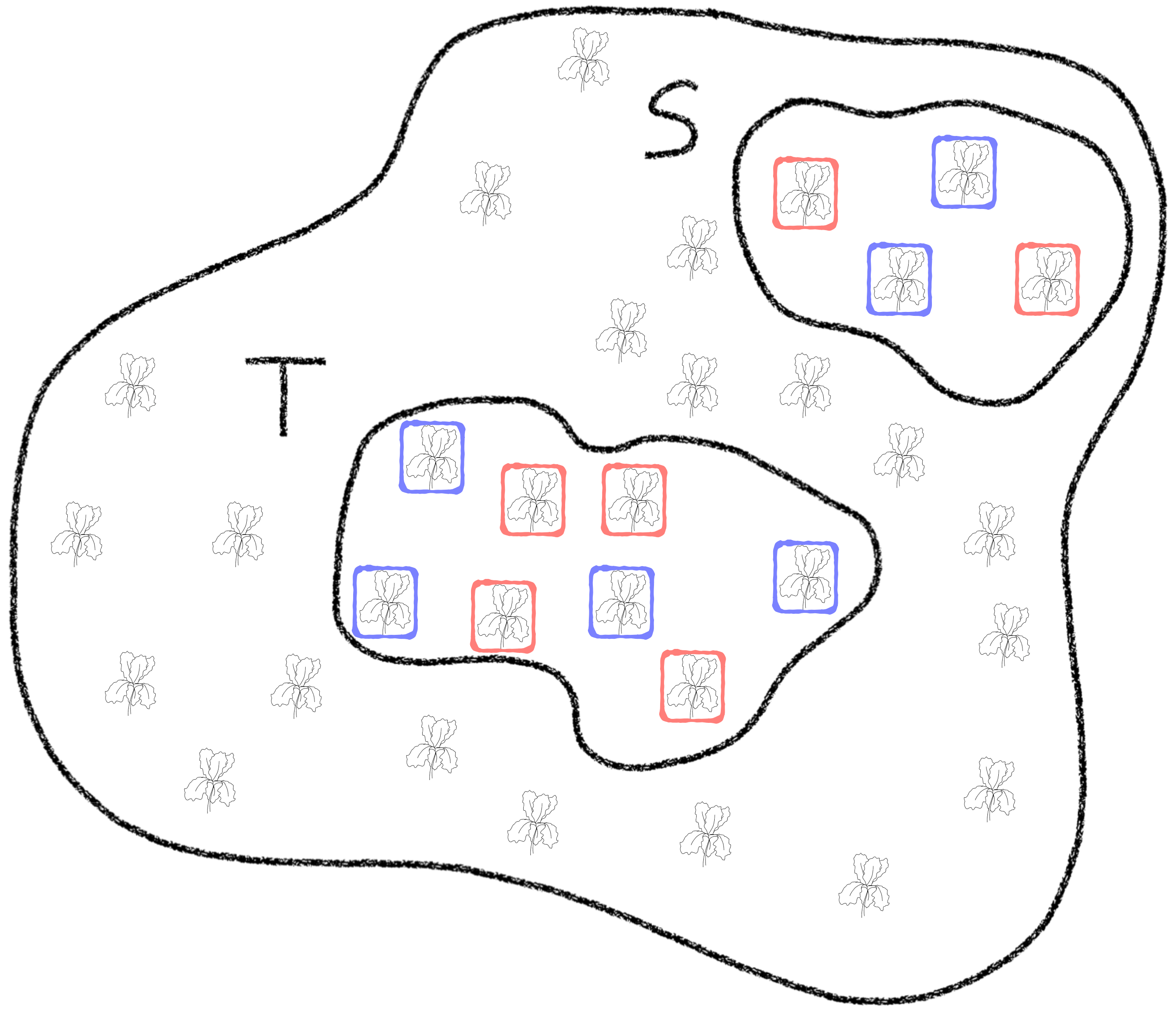
The set of all inputs and labels

$$\mathcal{Z} = \{(\vec{x}_i, y_i)\} = \{(\vec{x}_i, f(\vec{x}_i))\}$$



Train set is a finite sequence of pairs

$$T = ((\vec{x}_1, y_1), (\vec{x}_2, y_2), \ldots, (\vec{x}_m, y_m))$$



Test set is used to measure how well the classifier performs on unseen data

$$S = ((\vec{x}_1, y_1), (\vec{x}_2, y_2), \ldots, (\vec{x}_r, y_r))$$

Classification in machine learning

Classifier and the learning algorithm

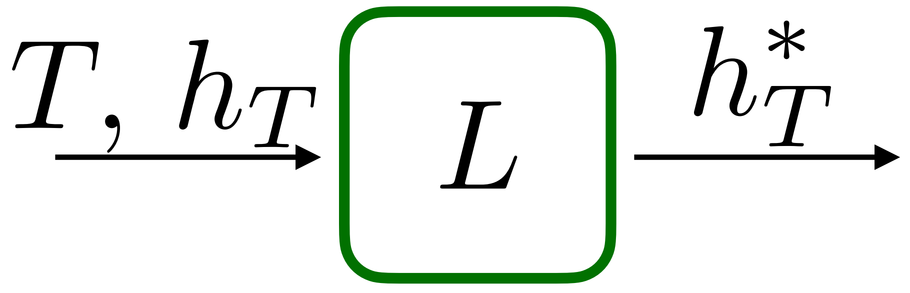
- **Classifier** h_T
  - input: instance of the domain set $\vec{x} \in \mathcal{X}$
  - output: a label $\tilde{y} \in \{0, 1\}$

Classification in machine learning

Classifier and the learning algorithm

- **Classifier** h_T
  - input: instance of the domain set $\vec{x} \in \mathcal{X}$
  - output: a label $\tilde{y} \in \{0, 1\}$

- **Learner** (learning algorithm) L
  - inputs: training set $T = \{(\vec{x}_i, y_i), i = 1, \ldots, m\}$
    classifier h_T
  - output: optimized classifier h_T^*



Classification in machine learning

Classifier and the learning algorithm

- **Classifier** h_T
  - input: instance of the domain set $\vec{x} \in \mathcal{X}$
  - output: a label $\tilde{y} \in \{0,1\}$

$$\xrightarrow{\vec{x}} \boxed{h_T} \xrightarrow{\tilde{y}}$$

- **Learner** (learning algorithm) L
  - inputs: training set $T = \{(\vec{x}_i, y_i), i = 1, \ldots, m\}$
    classifier h_T
  - output: optimized classifier h_T^*

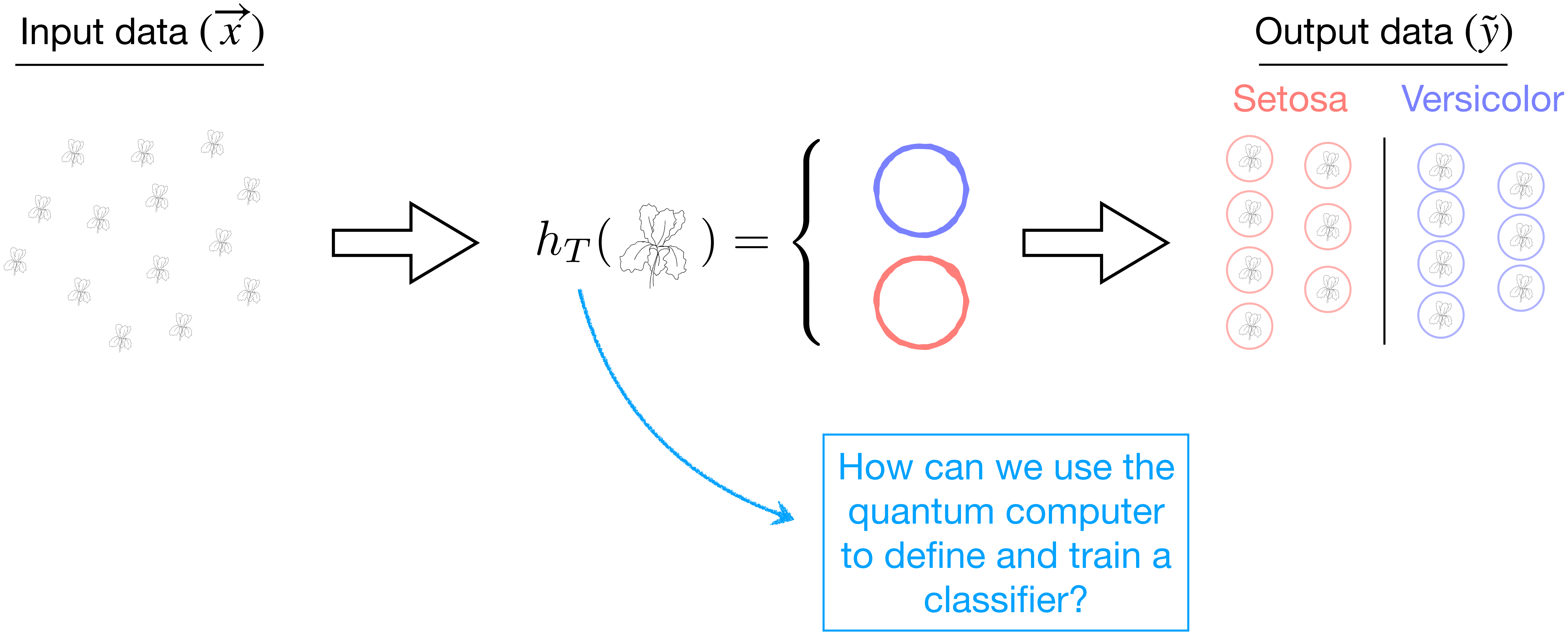
$$\xrightarrow{T,\, h_T} \boxed{L} \xrightarrow{h_T^*}$$

- **Error** of a classifier is the probability that it does not predict the correct label on a random data point

- Goal of the algorithm is to find h_T that **minimizes the error**.

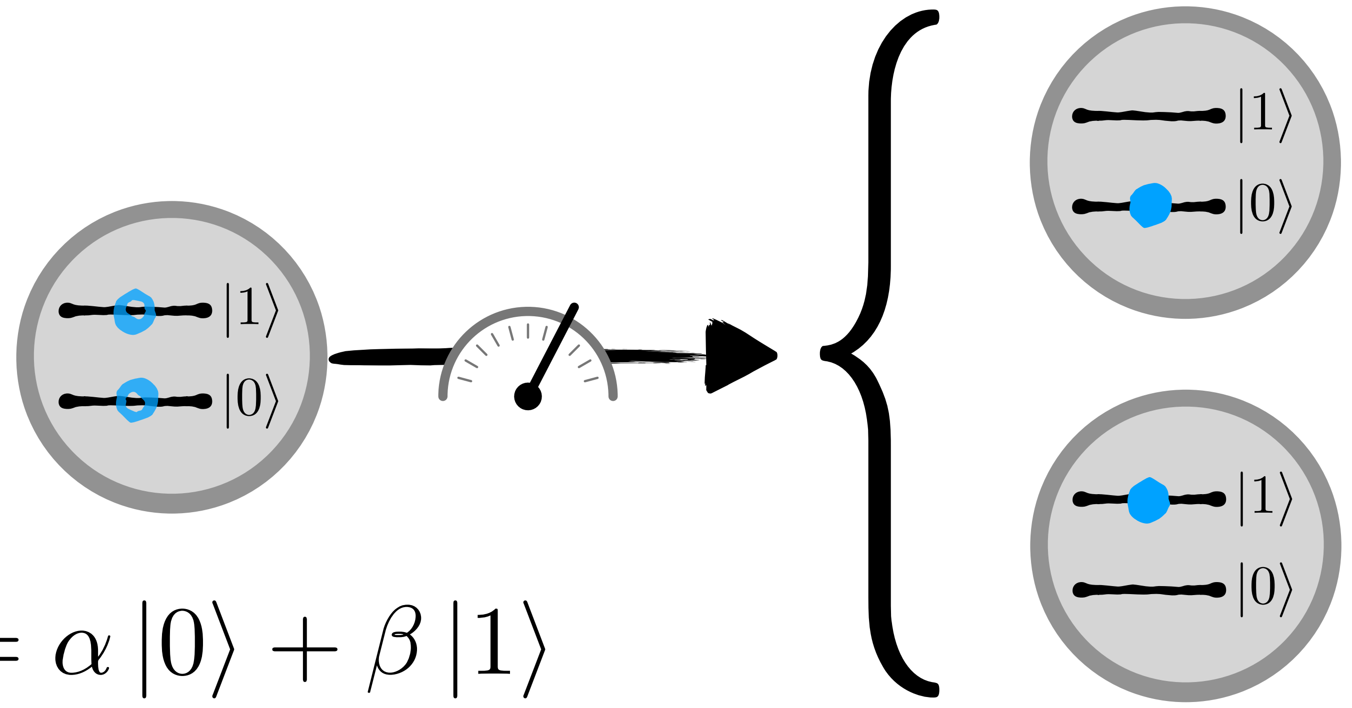
Classification in machine learning

The objective of this activity



Quantum refresher

1-qubit state



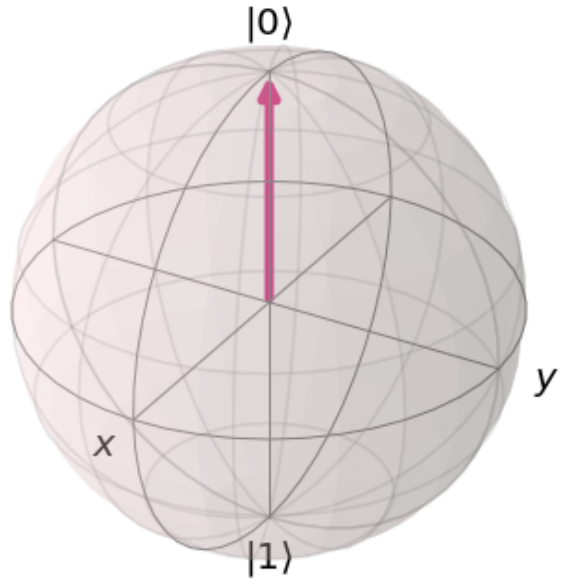
$$|\psi\rangle = \underbrace{\alpha\,|0\rangle + \beta\,|1\rangle}_{\text{superposition}}$$

$$\mathcal{P}_{|0\rangle} = |\alpha|^2$$

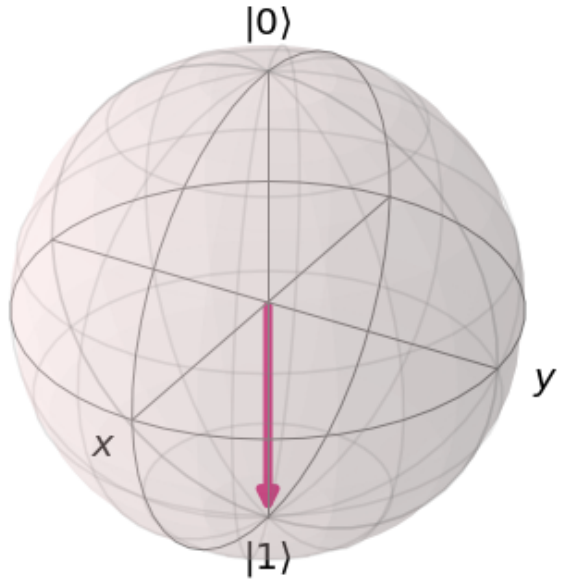
$$\mathcal{P}_{|1\rangle} = |\beta|^2$$

$$|\alpha|^2 + |\beta|^2 = 1$$

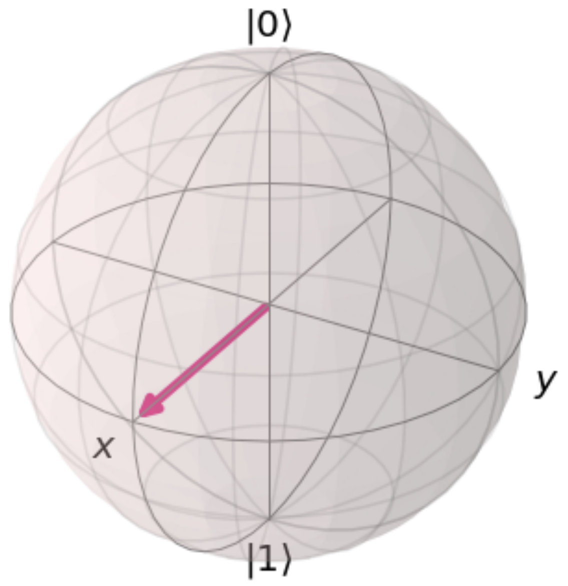
Bloch sphere



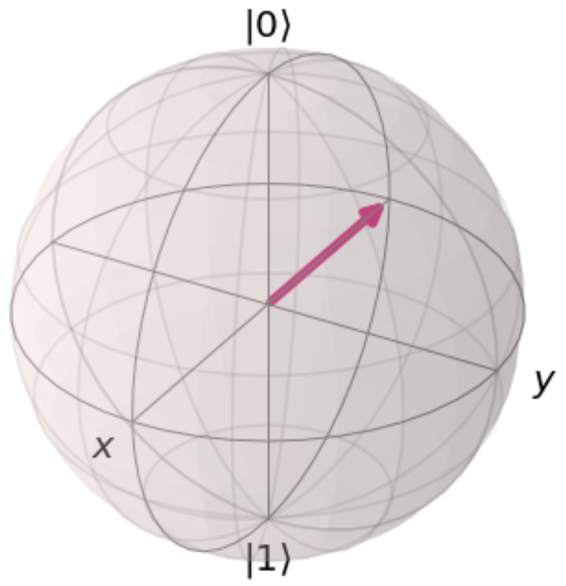
$$|\psi\rangle = |0\rangle$$



$$|\psi\rangle = |1\rangle$$



$$|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

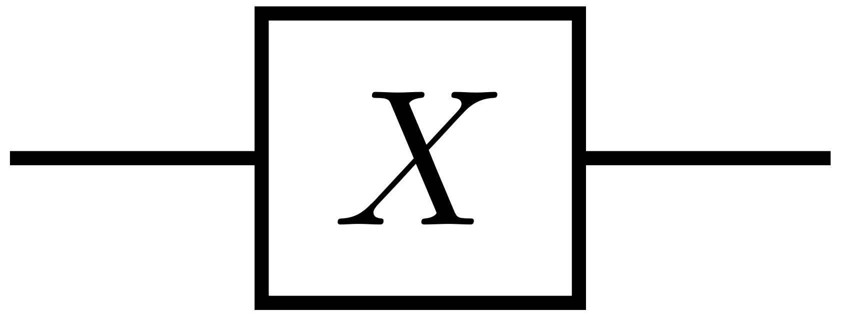


$$|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

Quantum refresher

1-qubit gates

X gate (NOT)



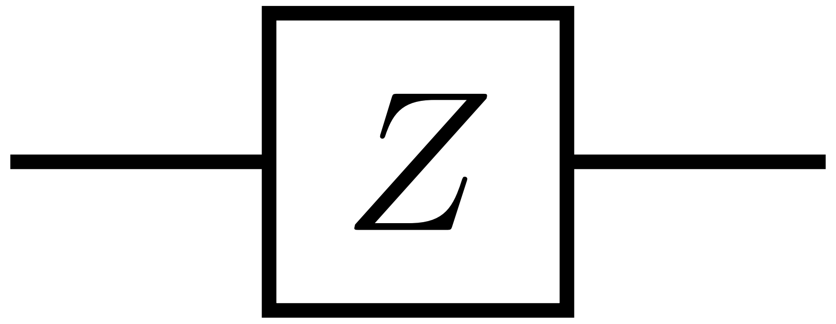
$|0\rangle \leftrightarrow |1\rangle$

Hadamard's gate

$|0\rangle \leftrightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

$|1\rangle \leftrightarrow \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

Z gate



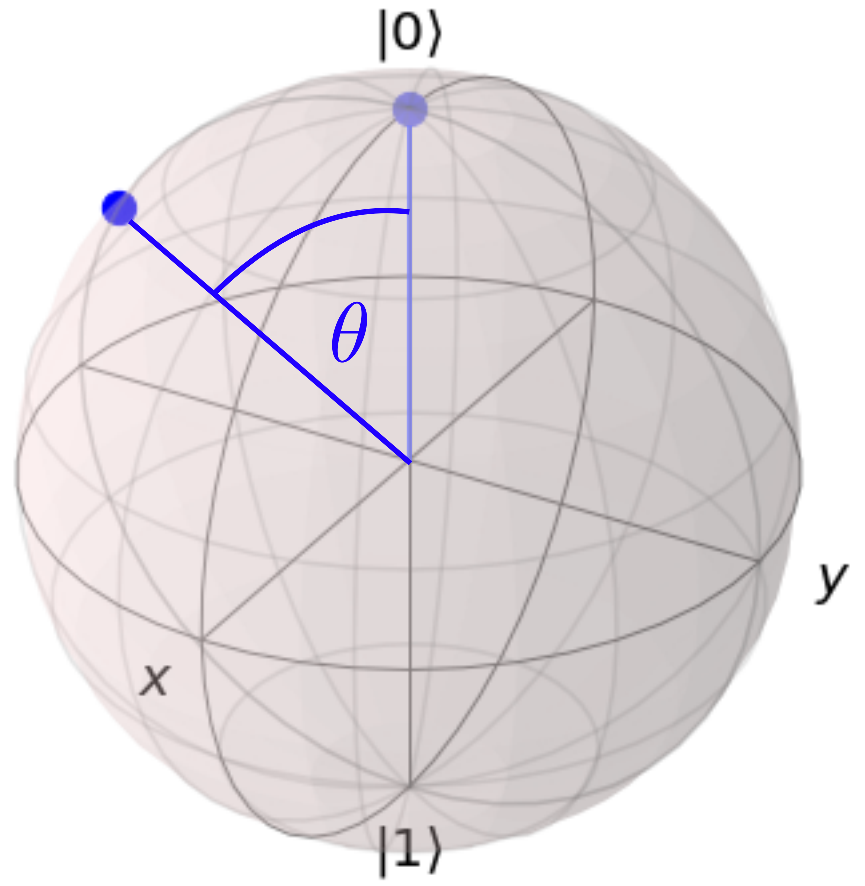
$|0\rangle \leftrightarrow |0\rangle$

$|1\rangle \leftrightarrow -|1\rangle$

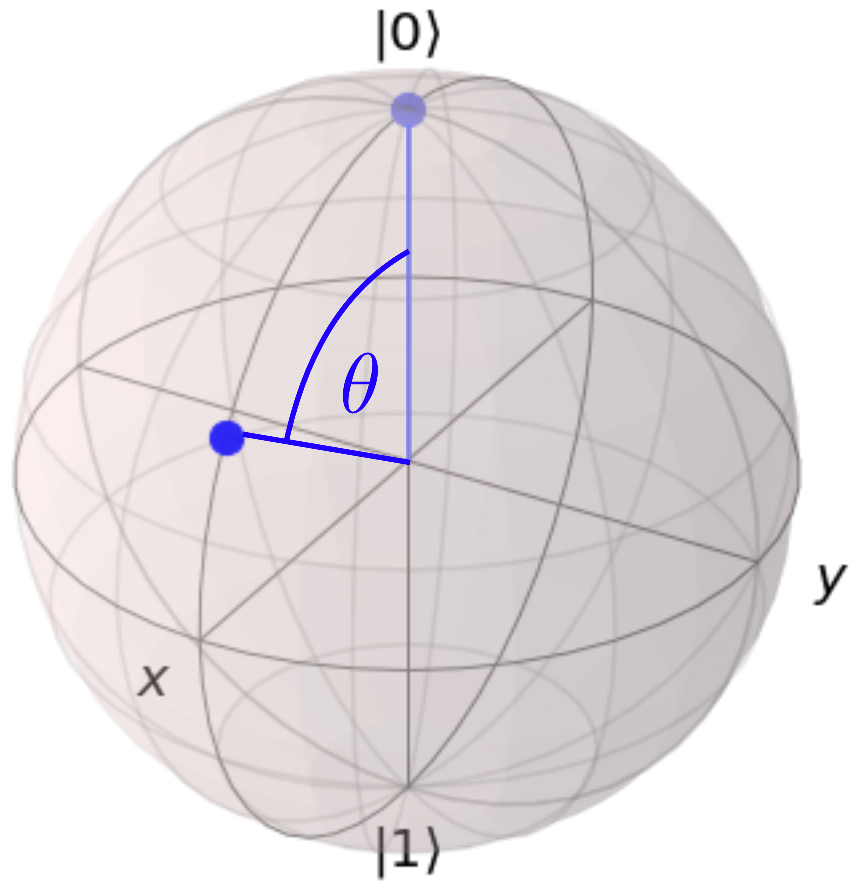
Quantum refresher

1-qubit parametrized gates

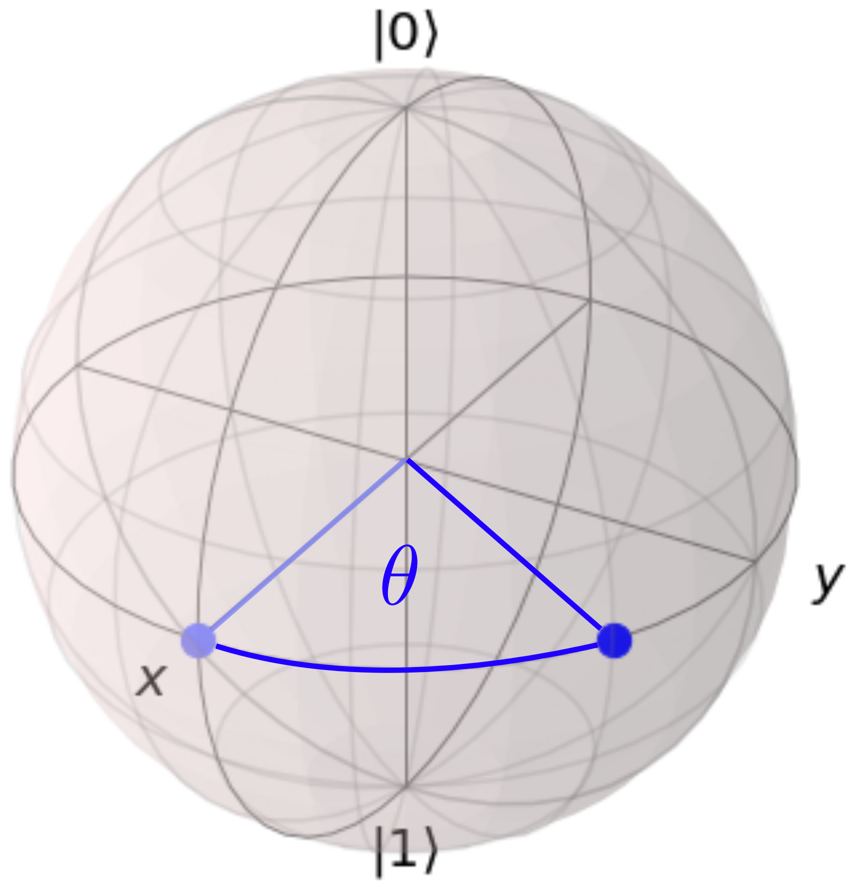
The state of a qubit can be rotated by an arbitrary angle θ along the *X*, *Y* and *Z* axis



$R_x(\pi/3)$



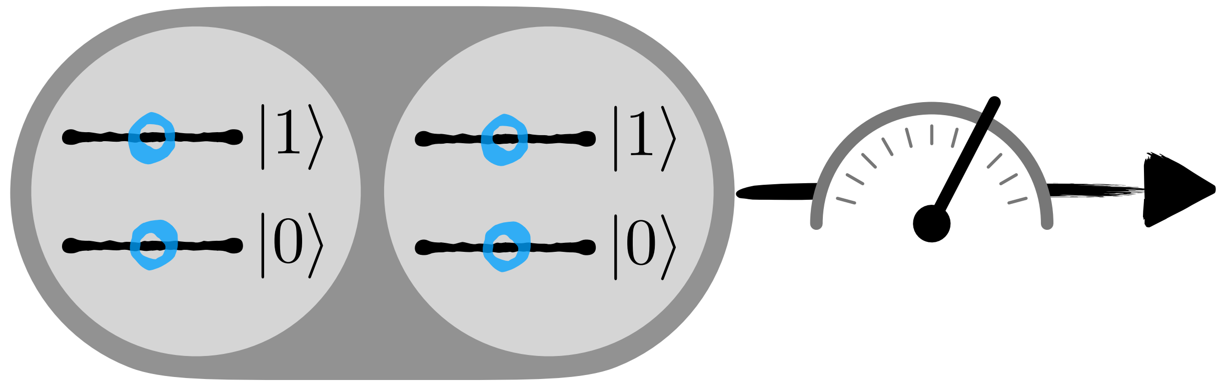
$R_y(\pi/3)$



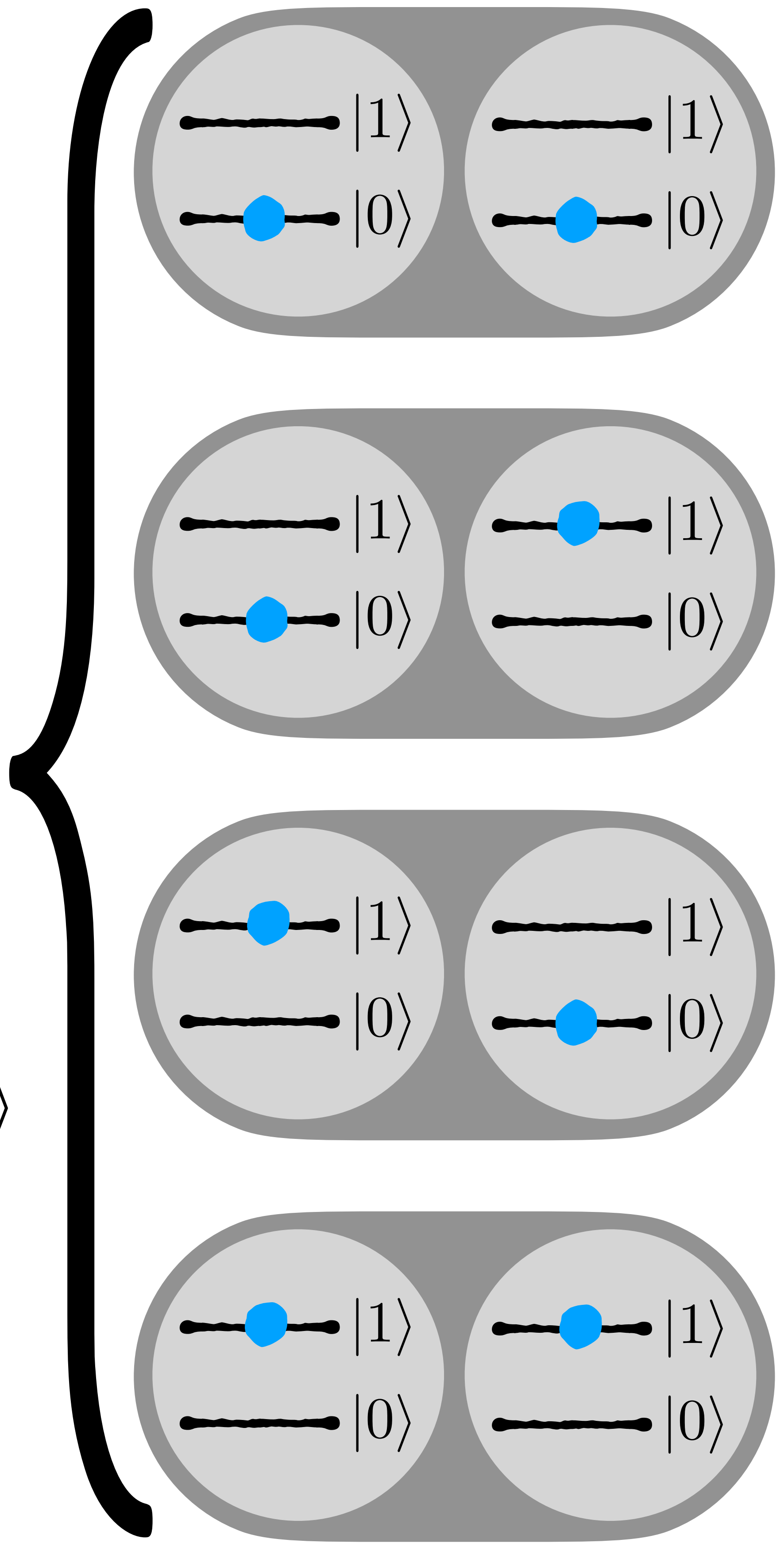
$R_z(\pi/3)$

Quantum refresher

2-qubits state



$$|\psi\rangle = \alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \delta\,|11\rangle$$



$$\mathcal{P}_{|00\rangle} = |\alpha|^2$$

$$\mathcal{P}_{|01\rangle} = |\beta|^2$$

$$\mathcal{P}_{|10\rangle} = |\gamma|^2$$

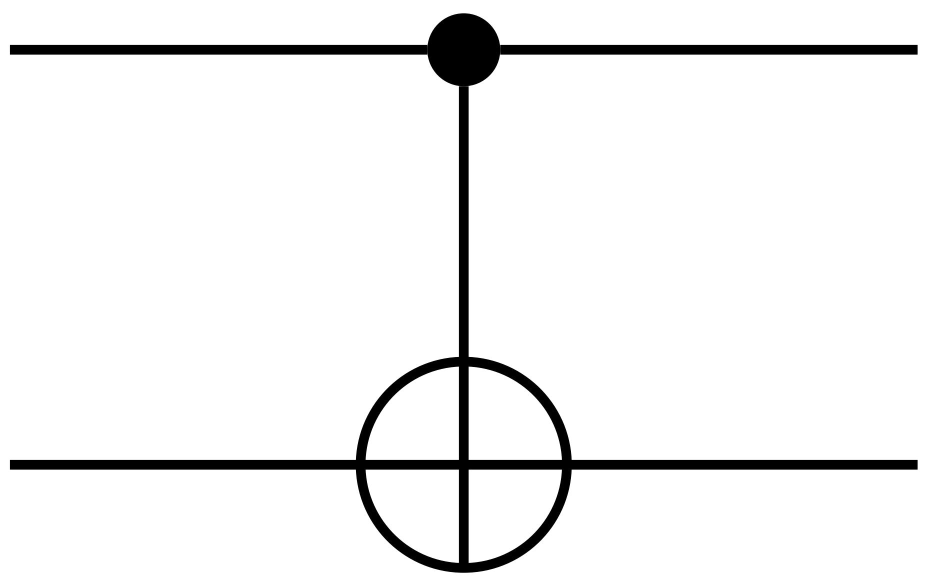
$$\mathcal{P}_{|11\rangle} = |\delta|^2$$

$$|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$$

Quantum refresher

2-qubit gates

Controlled X gate (*CNOT*)



CNOT gate creates **entanglement** between qubits!

$|00\rangle$

Target Control

$$|00\rangle \leftrightarrow |00\rangle$$

$$|01\rangle \leftrightarrow |11\rangle$$

$$|10\rangle \leftrightarrow |10\rangle$$

$$|11\rangle \leftrightarrow |01\rangle$$

Quantum refresher

Lab 1

Objectives

- Introduce the Iris dataset
- Visualize the application of gates on the Bloch sphere
- Build a quantum circuit with Qiskit
- Run an experiment on a backend and observe results

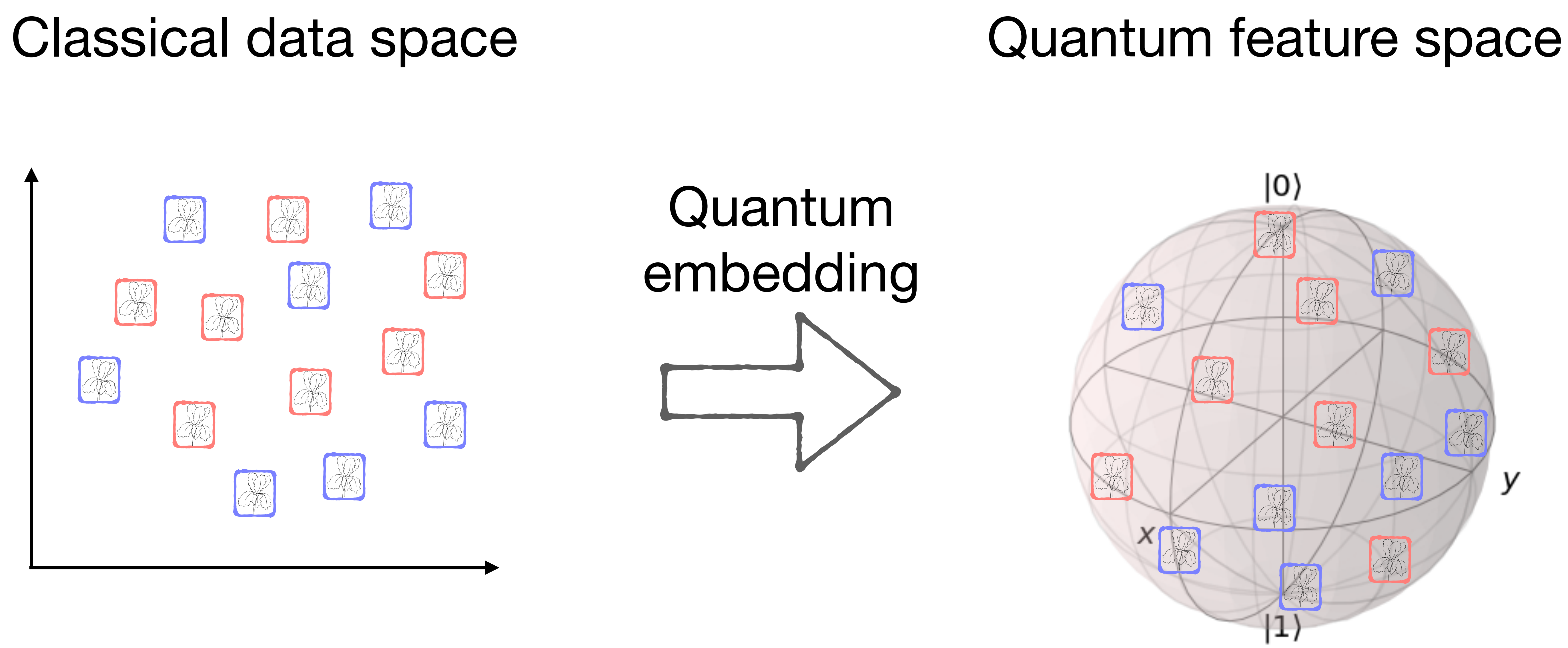
Exercice

- Build a given quantum circuit

Colab link Lab1

Data embedding (feature map)

Projection into quantum feature space

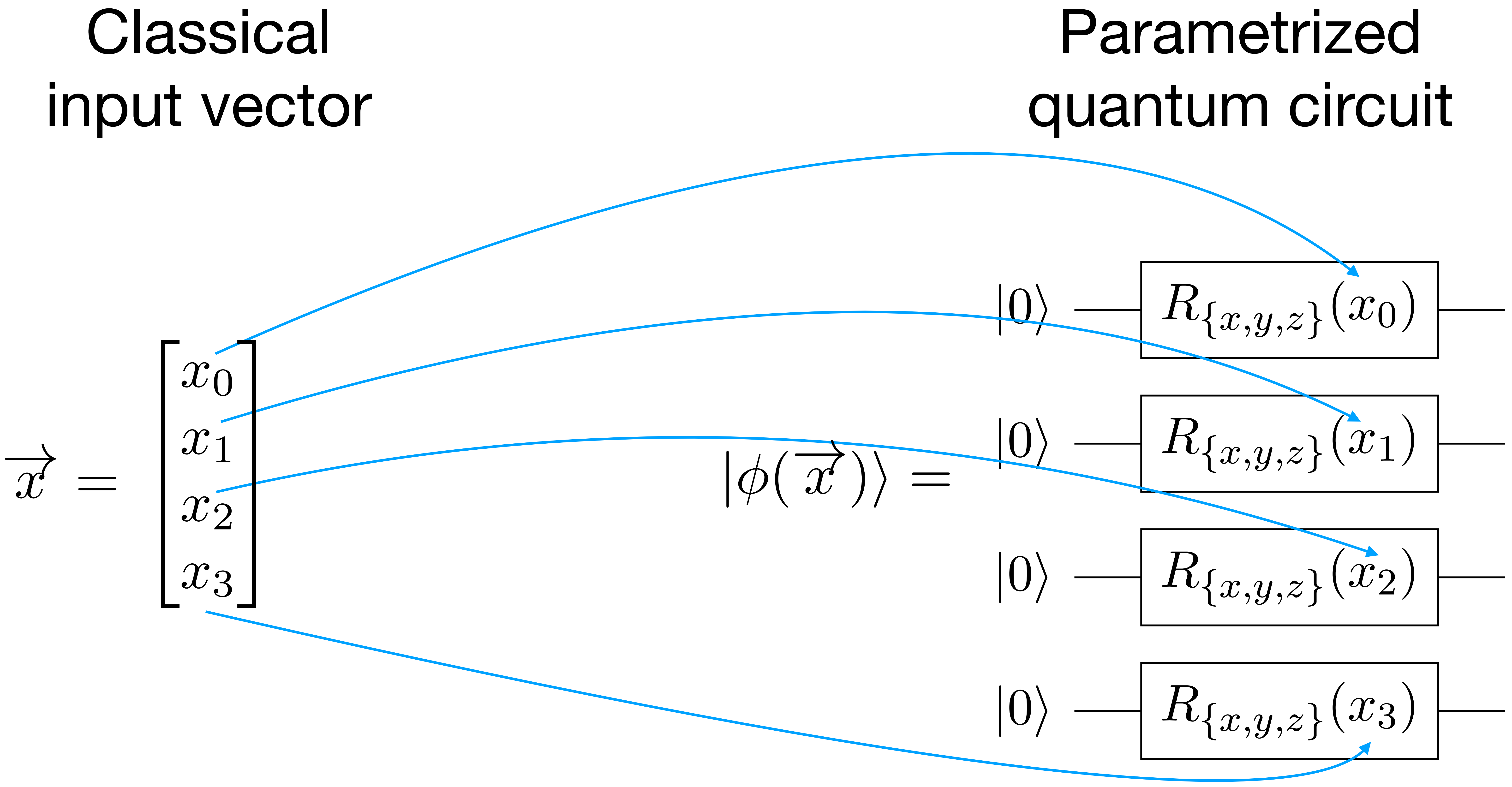


Two approaches:

- Angle embedding
- Amplitude embedding

Data embedding

Angle embedding



N features can be embedded in the rotation angles of *n* qubits

Data embedding

Amplitude embedding

Classical input vector

Quantum state

$$\vec{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = x_0 \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + x_1 \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} + x_3 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$|\phi(\vec{x})\rangle = x_0|00\rangle + x_1|01\rangle + x_2|10\rangle + x_3|11\rangle$$

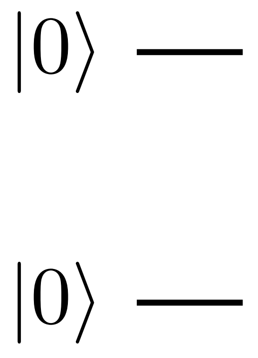
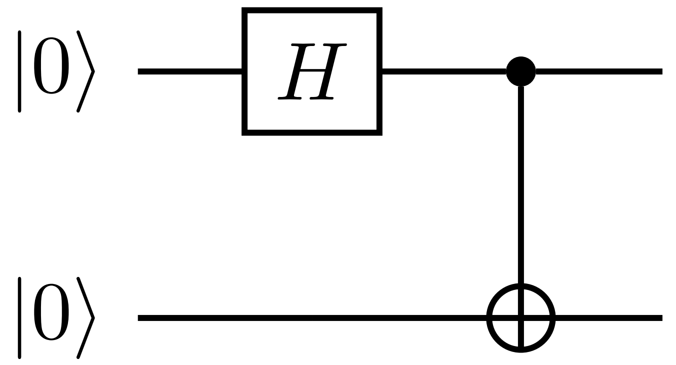
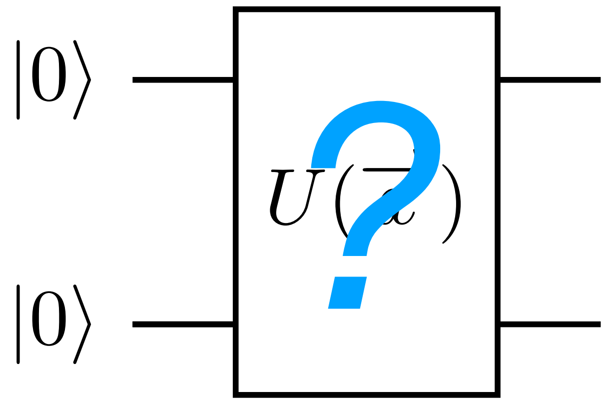
normalized

$$||\vec{x}||^2 = |x_0|^2 + |x_1|^2 + |x_2|^2 + |x_3|^2 = 1$$

N features can be embedded in the amplitudes of $log(N)$ qubits

Data embedding

Amplitude embedding - examples

Classical data	Normalized data	Quantum state	Quantum circuit
$\vec{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\vec{\tilde{x}} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\lvert\phi(\vec{x})\rangle = \lvert 00\rangle$	
$\vec{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	$\vec{\tilde{x}} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	$\lvert\phi(\vec{x})\rangle = \frac{1}{\sqrt{2}}(\lvert 00\rangle + \lvert 11\rangle)$	
$\vec{x} = \begin{bmatrix} 1 \\ 3.5 \\ -2 \\ 0 \end{bmatrix}$	$\vec{\tilde{x}} = \frac{1}{\sqrt{17.25}} \begin{bmatrix} 1 \\ 3.5 \\ -2 \\ 0 \end{bmatrix}$	$\lvert\phi(\vec{x})\rangle = \frac{1}{\sqrt{17.25}}(\lvert 00\rangle + 3.5\lvert 01\rangle - 2\lvert 10\rangle)$	

Data embedding

Lab 2

Objectives

- Map 1D data in quantum feature space with angle embedding
- Angle and amplitude embedding with the Iris dataset
- Qiskit quantum feature maps as embedding methods

Exercice

- Implement a quantum feature map

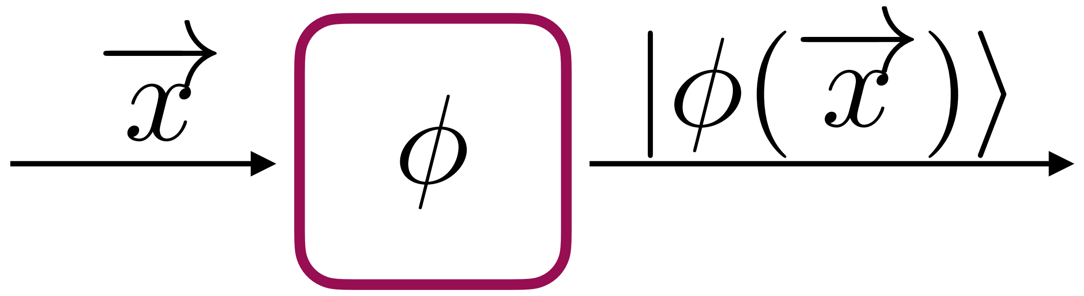
[Colab link Lab2]

Quantum classifier

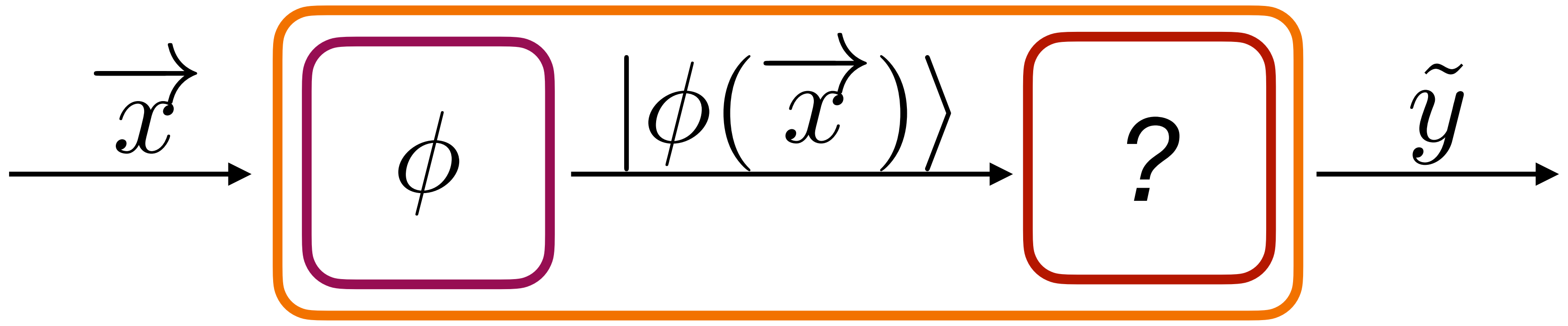
From data embedding to classifier

- Classifier

- Quantum embedding

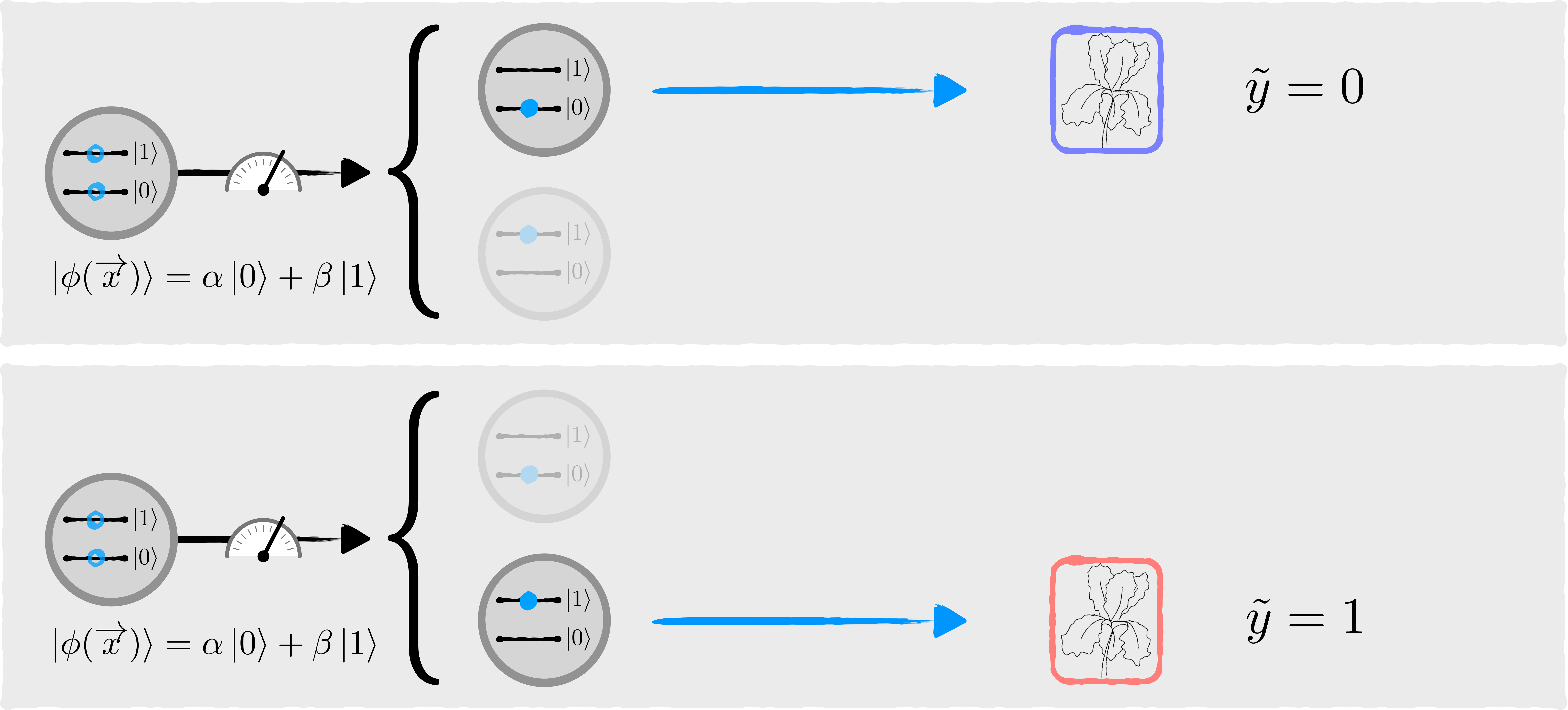


- A piece is missing!



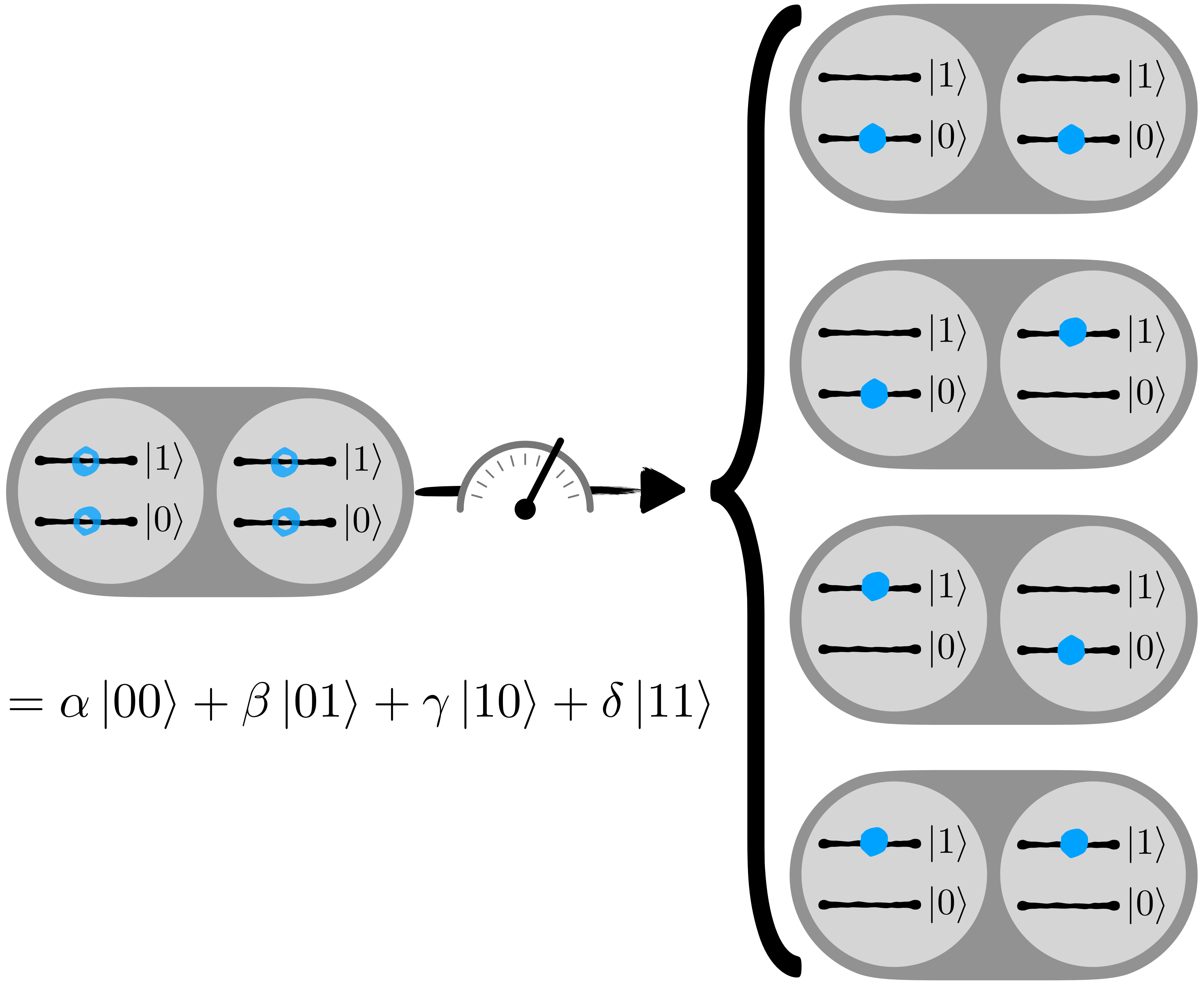
Quantum classifier

Measurement and interpretation function for 1 qubit



Quantum classifier

Measurement and interpretation function for 2 qubits



$$|\psi\rangle = \alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \delta\,|11\rangle$$

The interpretation function takes as input two bits

00
01
10
11

and should return 0 or 1.

We have a few options!

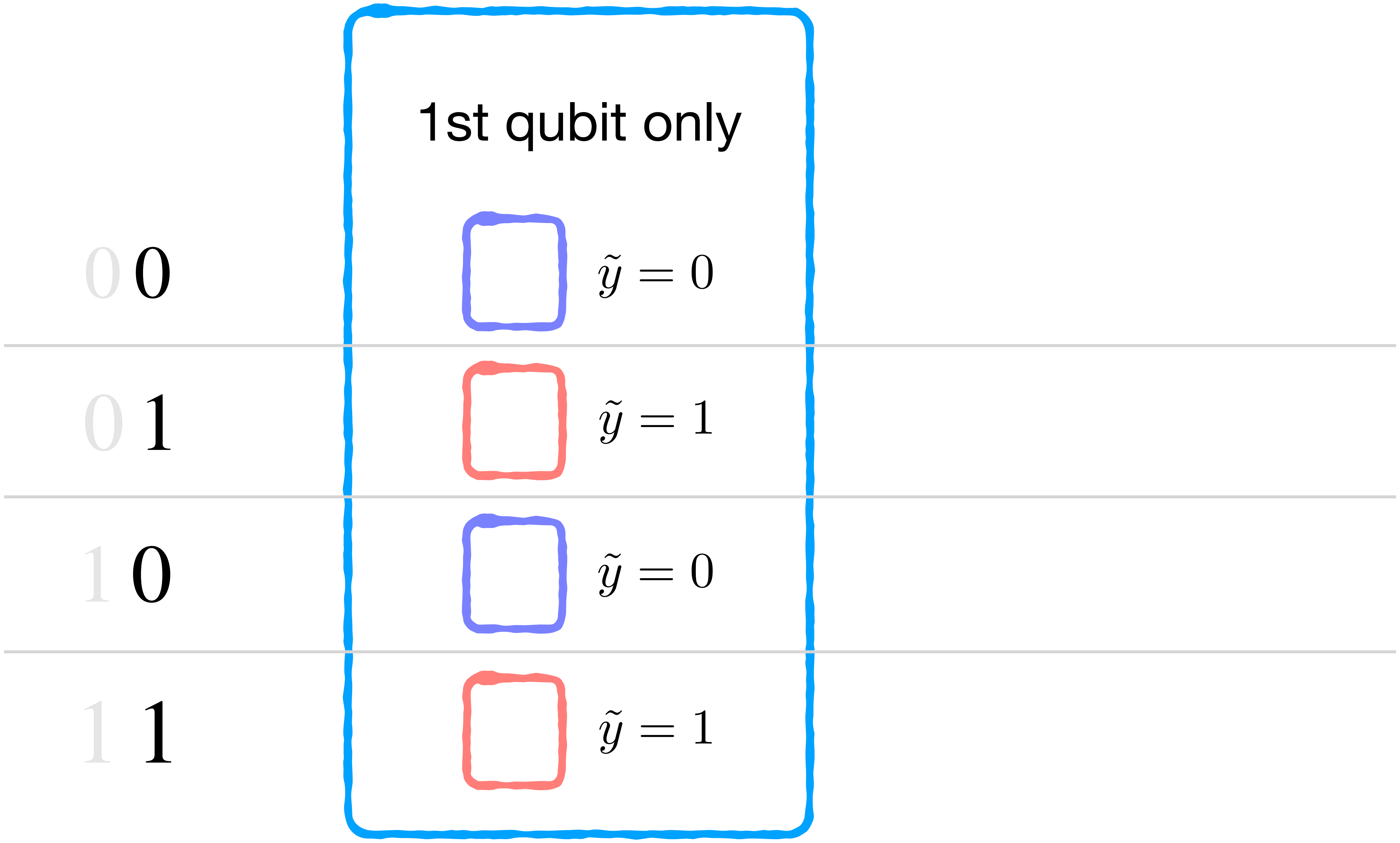
Quantum classifier

Measurement and interpretation function for 2 qubits

0 0
0 1
1 0
1 1

Quantum classifier

Measurement and interpretation function for 2 qubits



Quantum classifier

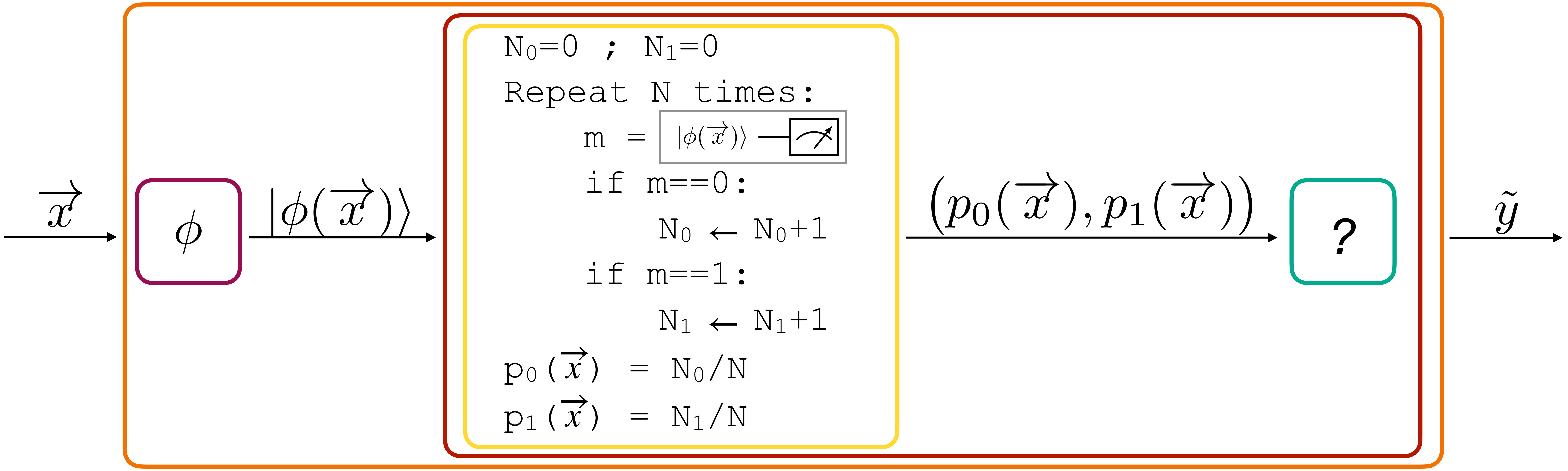
Measurement and interpretation function for 2 qubits

	1st qubit only	parity
0 0	$\tilde{y} = 0$	$\tilde{y} = 0$
0 1	$\tilde{y} = 1$	$\tilde{y} = 1$
1 0	$\tilde{y} = 0$	$\tilde{y} = 1$
1 1	$\tilde{y} = 1$	$\tilde{y} = 0$

Quantum classifier

Expectation value of the interpretation function

- The outcome of the measurement of a quantum state is **probabilistic.**
- We have to repeat the state preparation and the measurement **many times**.
- Result is a **probability** that input $\vec{x}$ belongs to the 0/1 class.



Quantum classifier

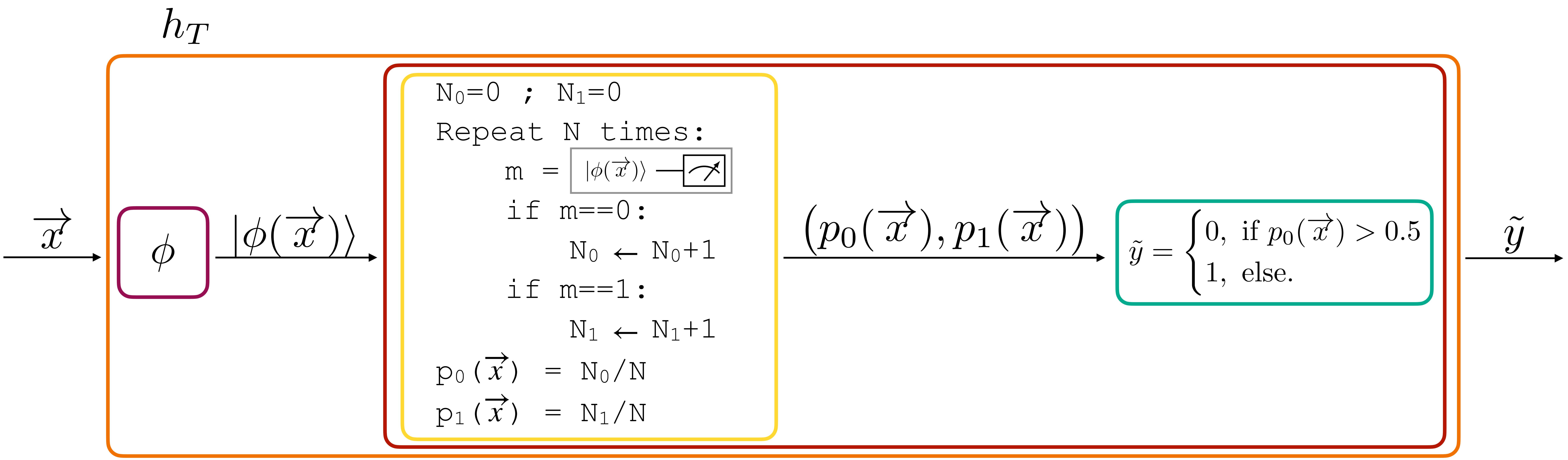
Decision rule

We assign the label based on the following decision rule

$$\xrightarrow{(p_0(\overrightarrow{x}), p_1(\overrightarrow{x}))} \boxed{\tilde{y} = \begin{cases} 0, & \text{if } p_0(\overrightarrow{x}) > 0.5 \\ 1, & \text{else.} \end{cases}} \xrightarrow{\tilde{y}}$$

Quantum classifier

A (first) definition of a quantum classifier



Quantum classifier

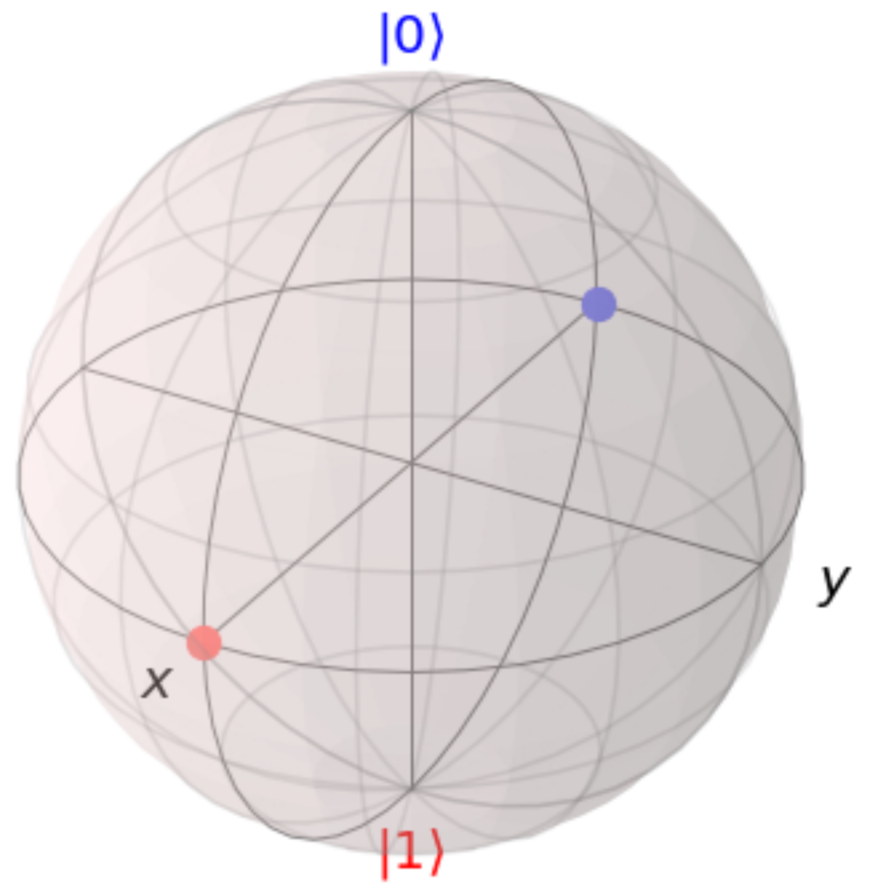
Example of what can go wrong

- Dataset composed of two points

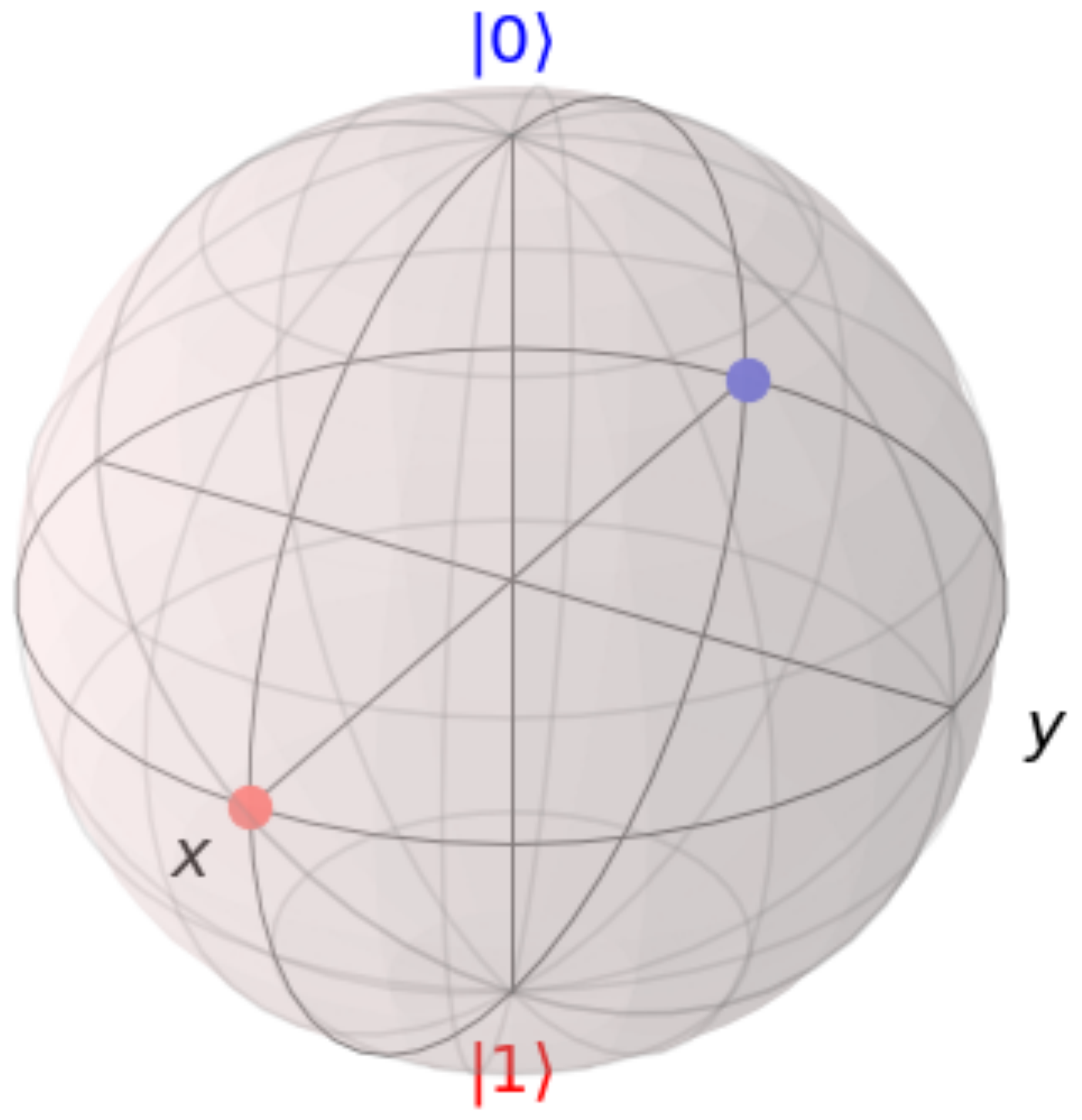
$$\mathcal{D} = \Big\{(x_0, y_0), (x_1, y_1)\Big\} = \Big\{(+1, 0), (-1, 1)\Big\}$$

- Angle embedding

$$\phi(x) = \mathcal{R}_y(\frac{x\pi}{2}) \quad \begin{cases} |\phi(+1)\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ |\phi(-1)\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \end{cases}$$



Quantum classifier

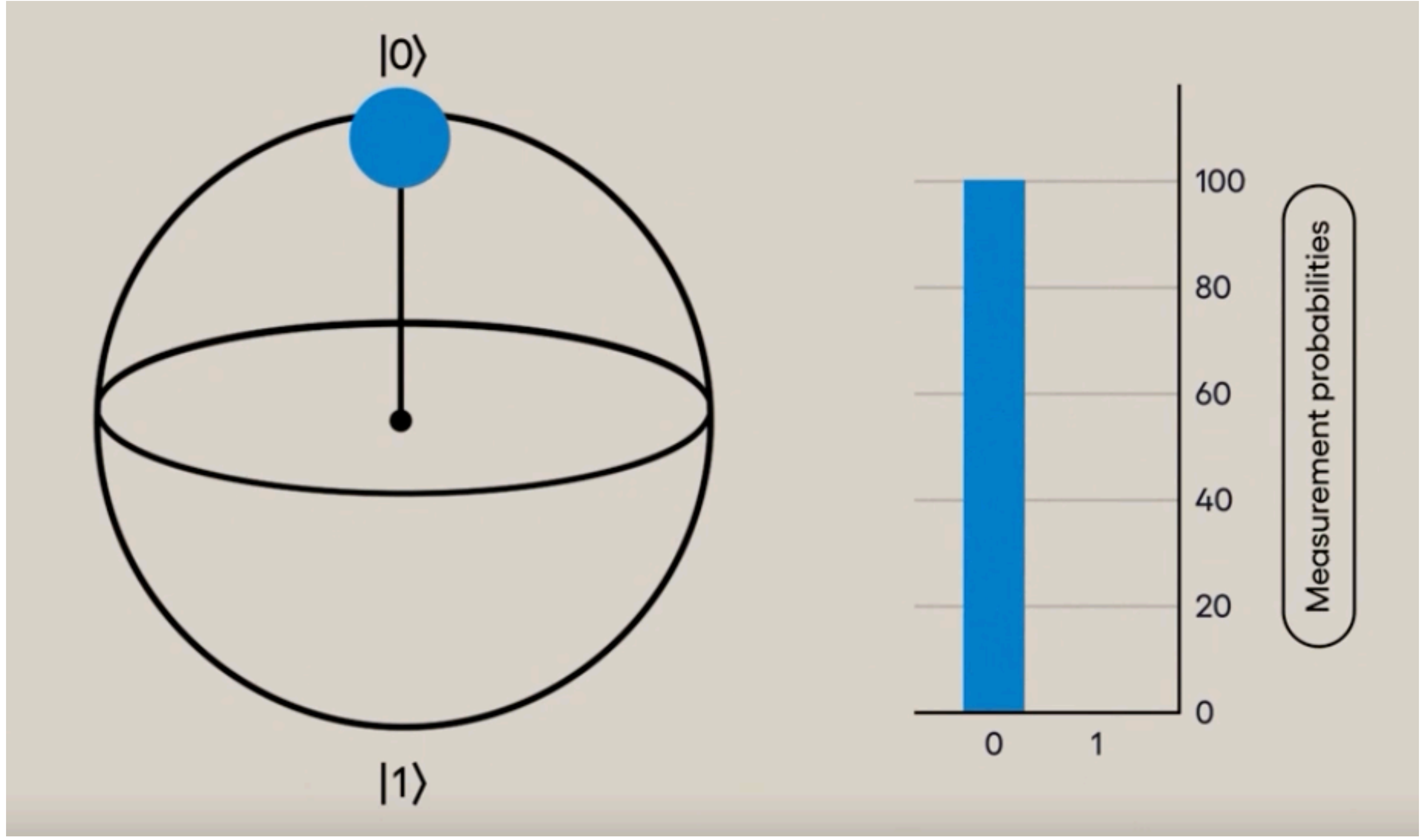


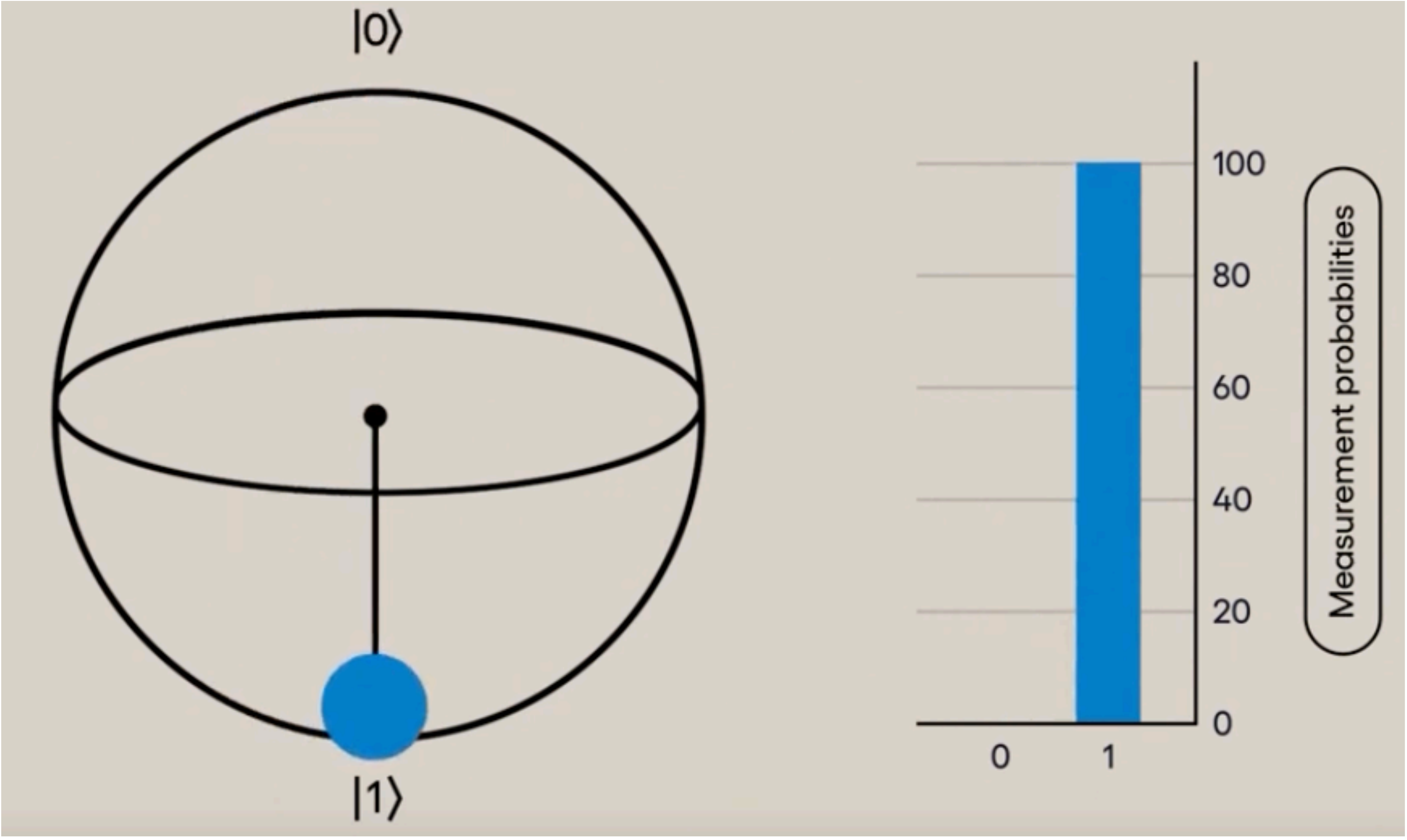
What does it really mean to measure?

$\hat{Z}$

Measuring along Z measures the hight of the qubit in the Bloch sphere

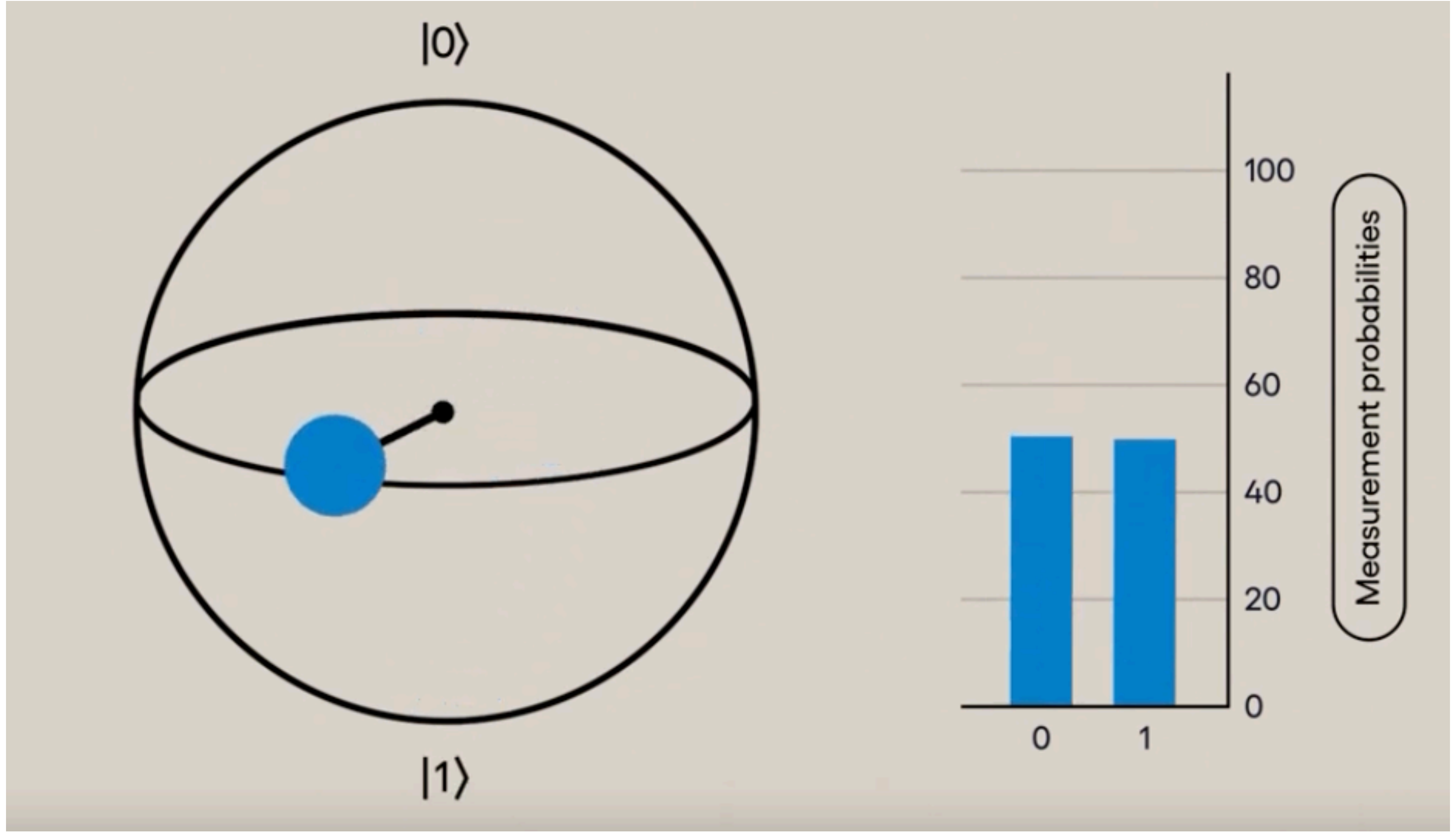
Quantum classifier

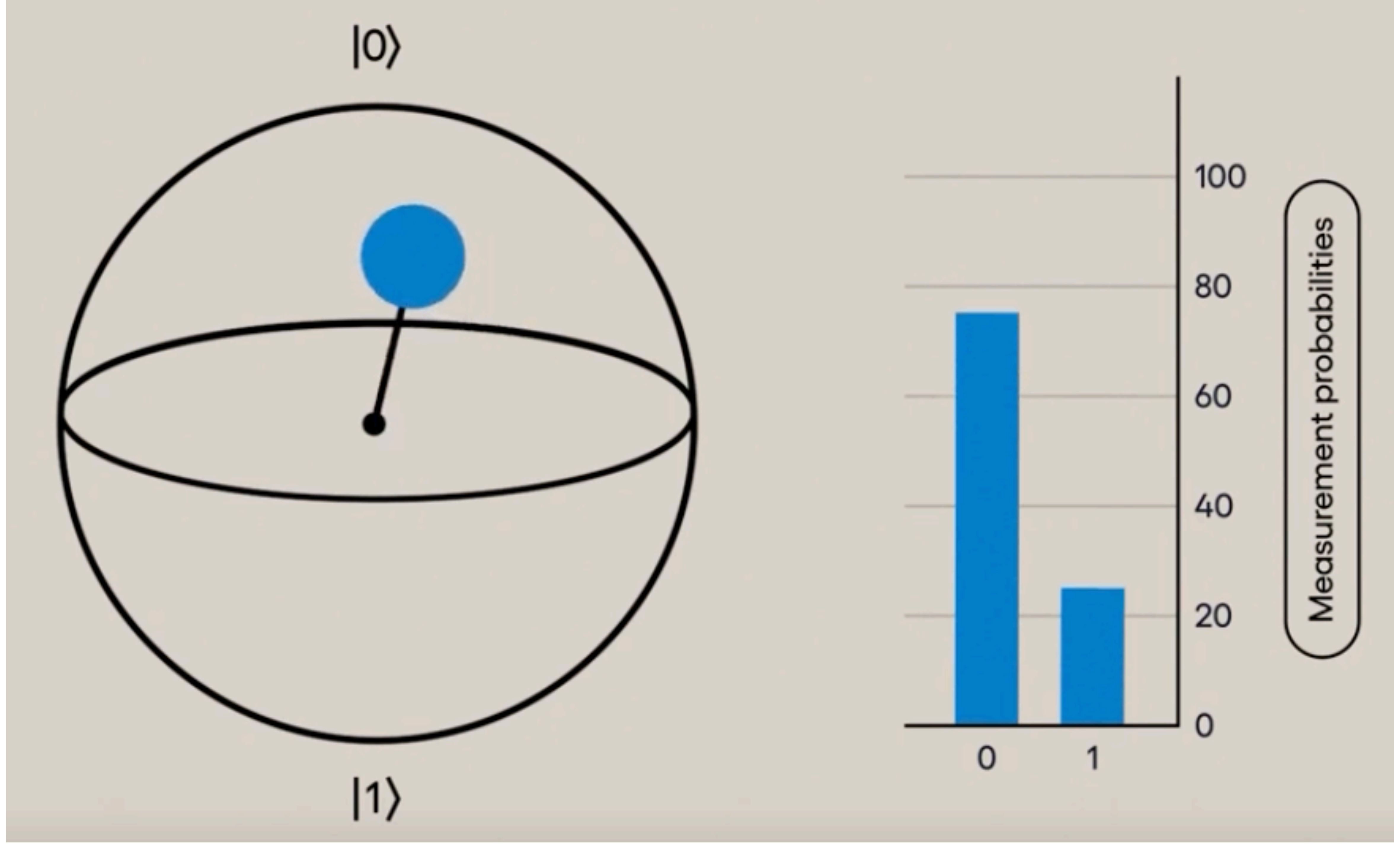




$\hat{Z}$

Quantum classifier

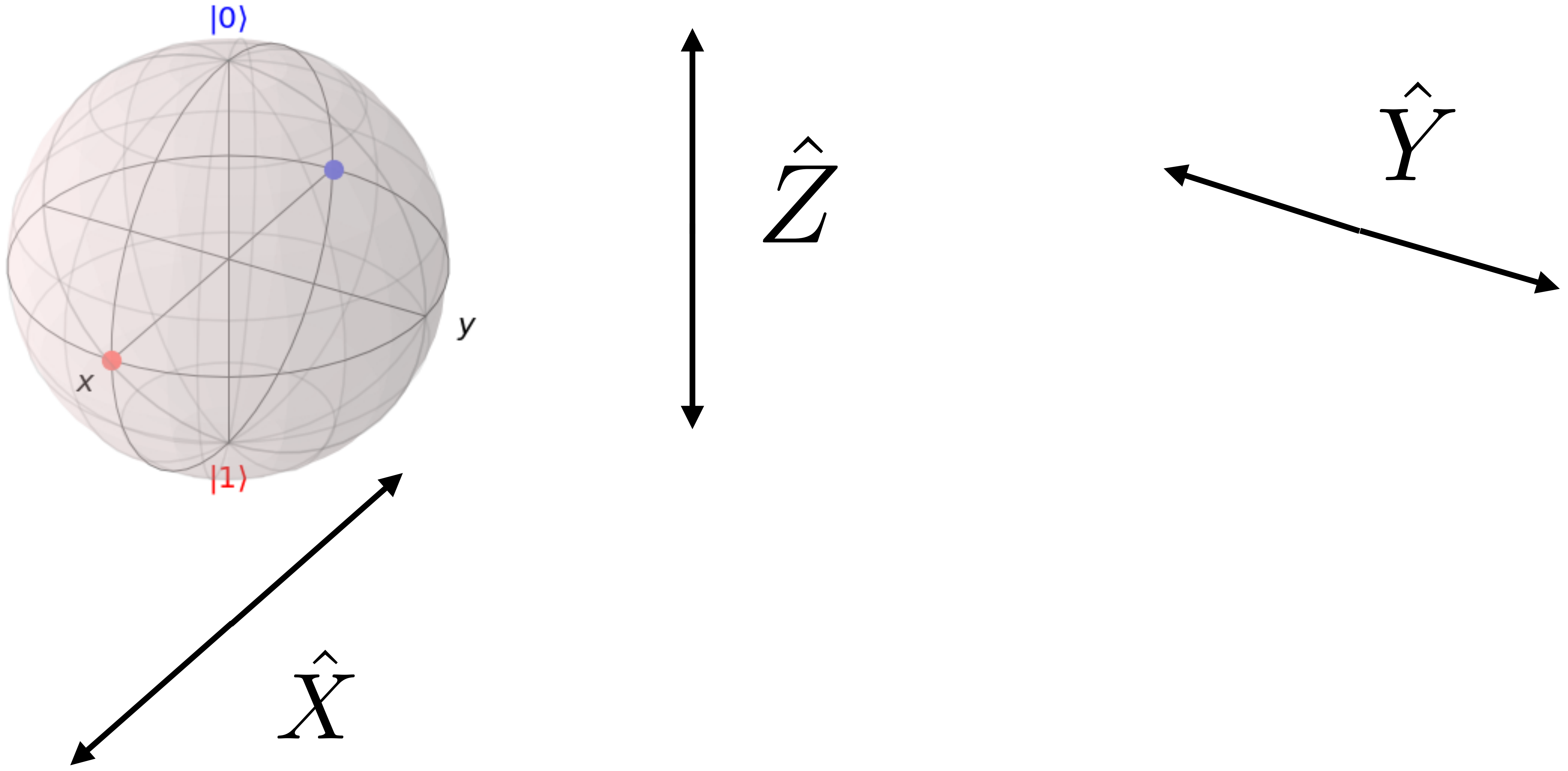




$\hat{Z}$

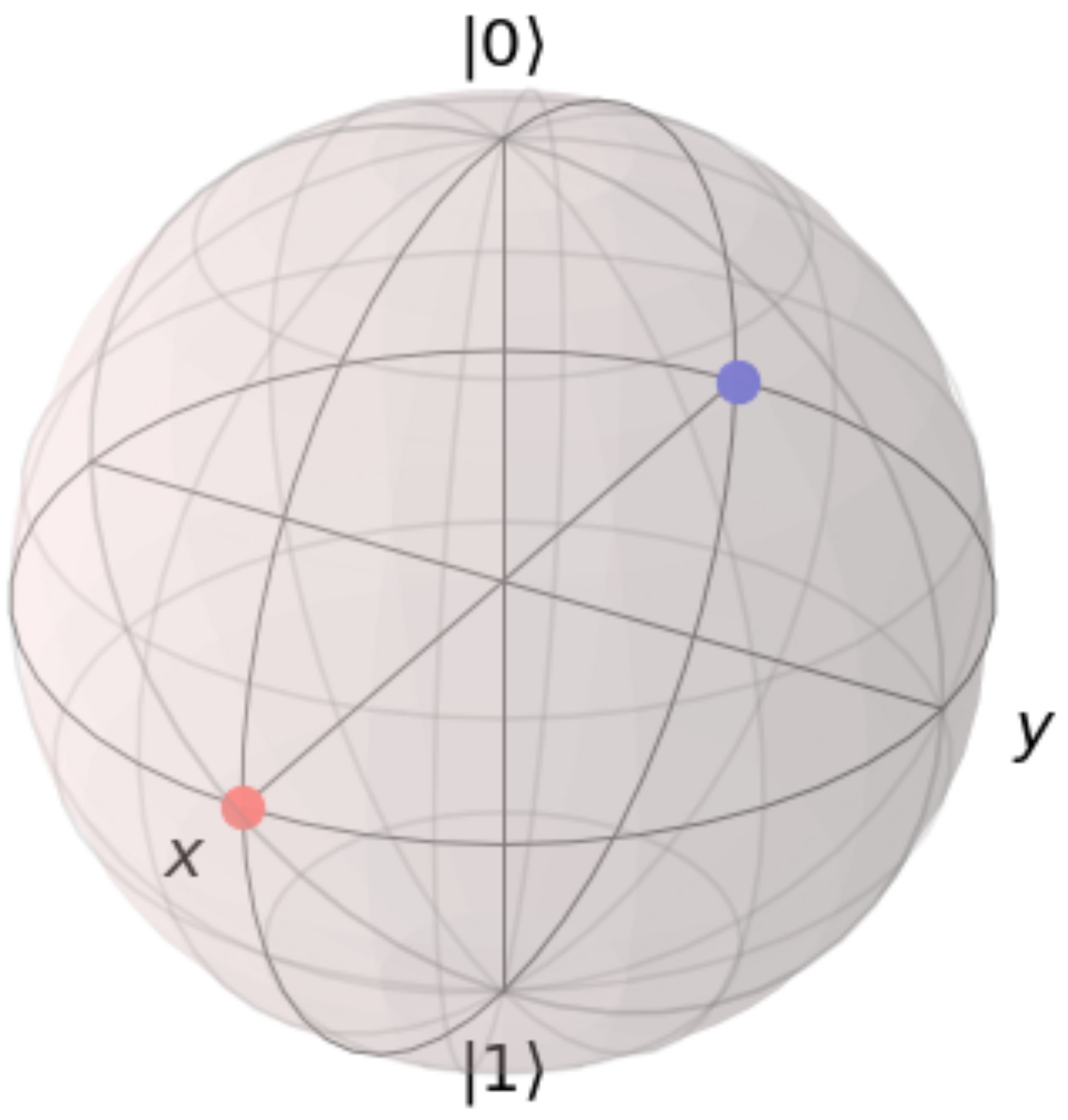
Quantum classifier

What does it really mean to measure?



Quantum classifier

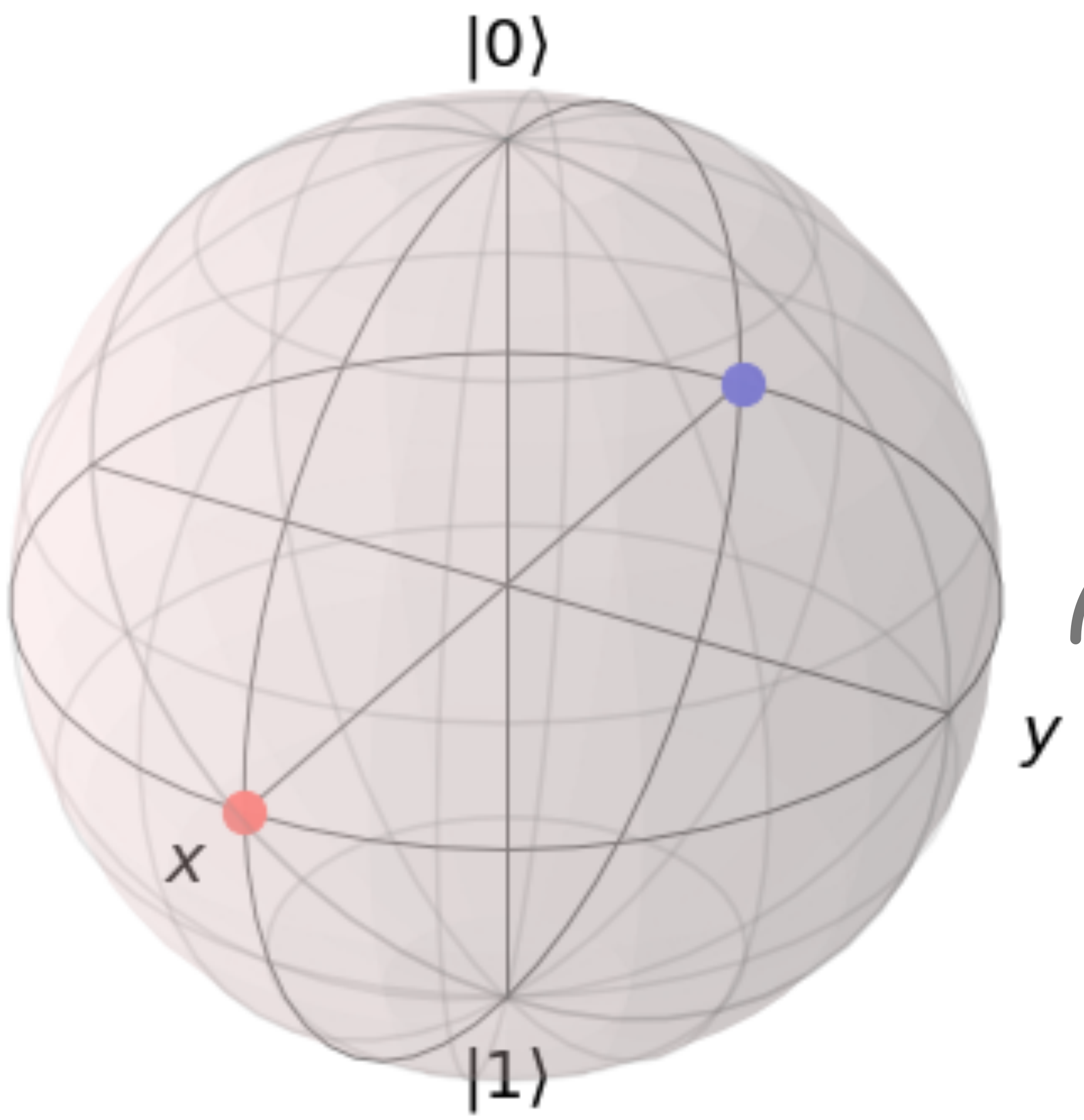
Measure along a different axis

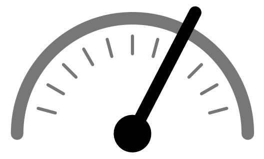


- Computational basis in which measurements are made corresponds to the Z basis.

Quantum classifier

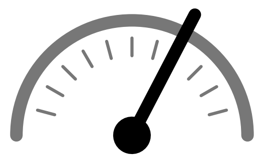
Measure along a different axis





$$\begin{cases} {\color{red}0}, & 50\% \text{ probability} \\ {\color{blue}1}, & 50\% \text{ probability} \end{cases}$$

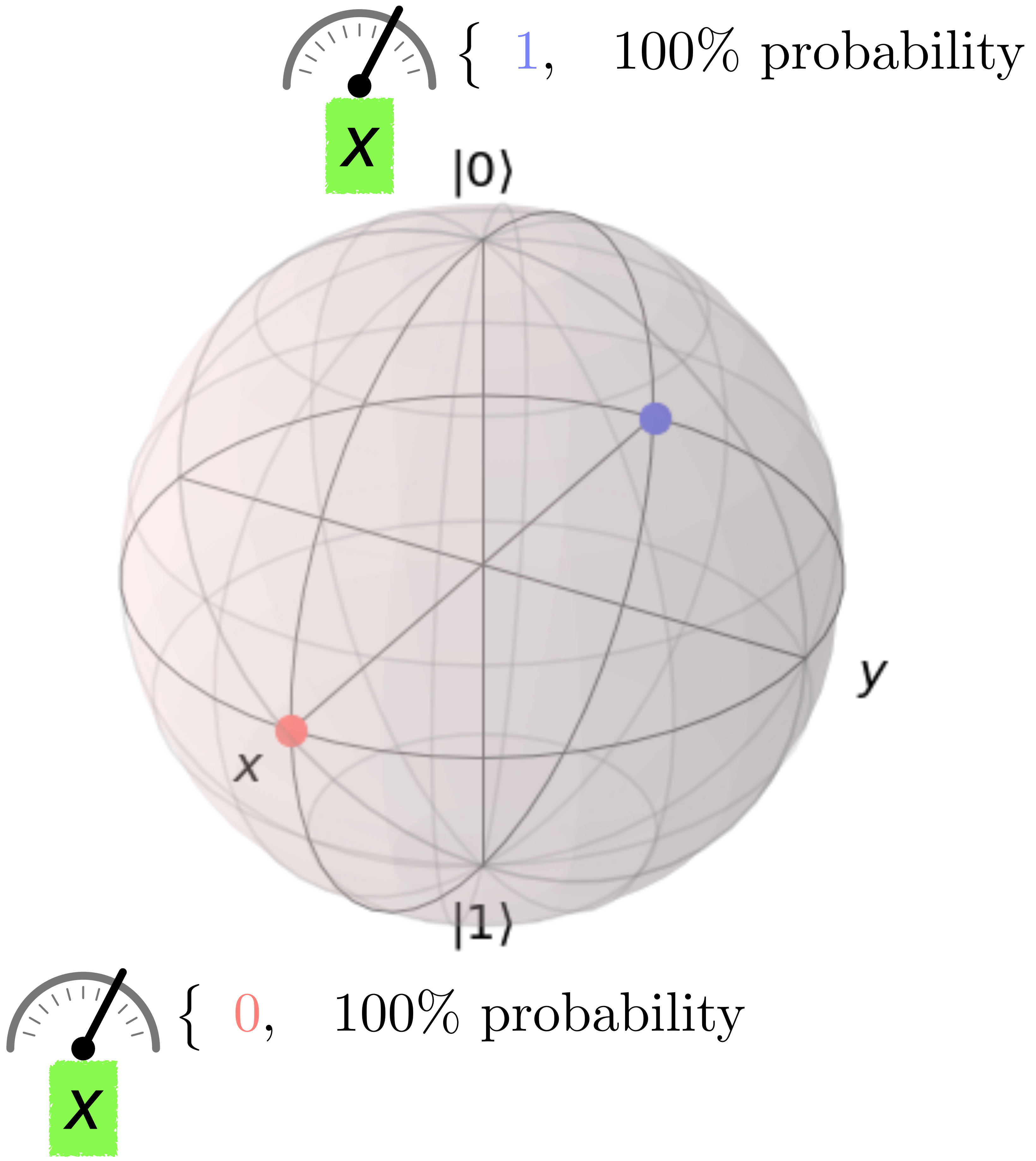
- Computational basis in which measurements are made corresponds to the *Z* basis.
- Measuring along the *Z* axis does not allow to distinguish between classes 0 and 1.



$$\begin{cases} {\color{red}0}, & 50\% \text{ probability} \\ {\color{blue}1}, & 50\% \text{ probability} \end{cases}$$

Quantum classifier

Measure along a different axis

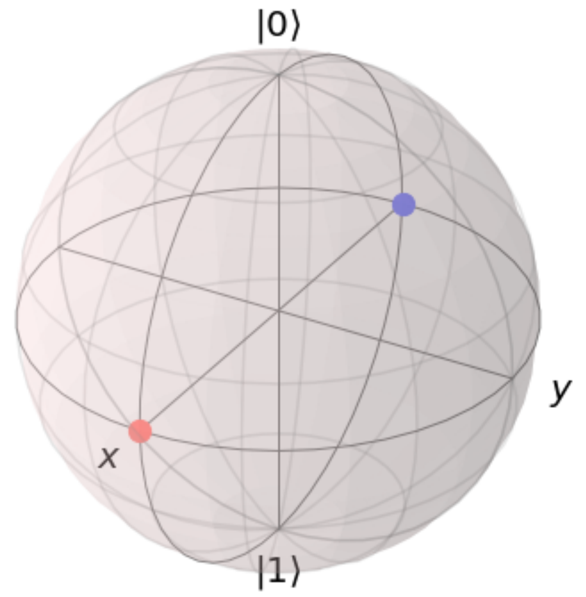


- Computational basis in which measurements are made corresponds to the Z basis.
- Measuring along the Z axis does not allow to distinguish between classes 0 and 1.
- If somehow we could measure along the ***X* axis**, both classes would be easily separable…

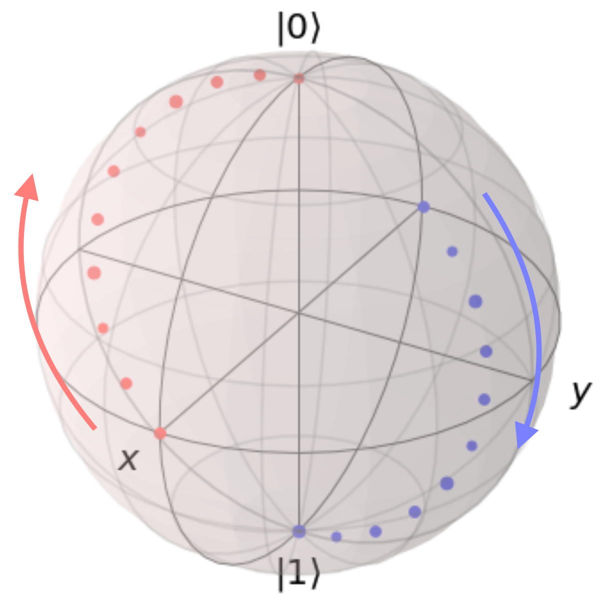
Quantum classifier

Measure along a different axis

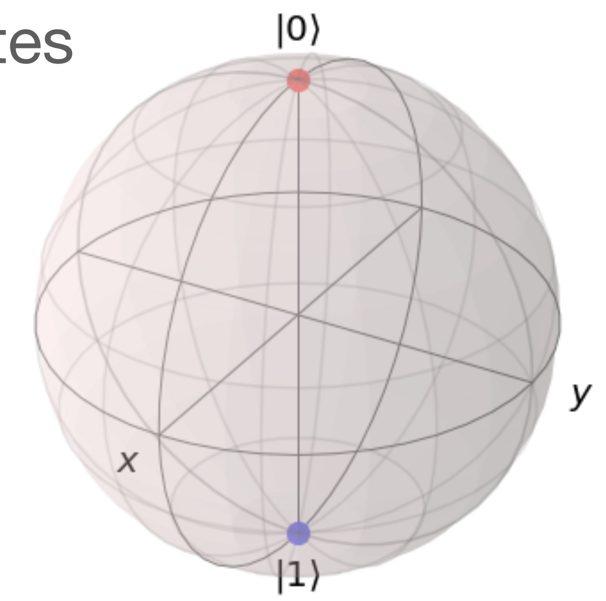
Initial states



Rotated states



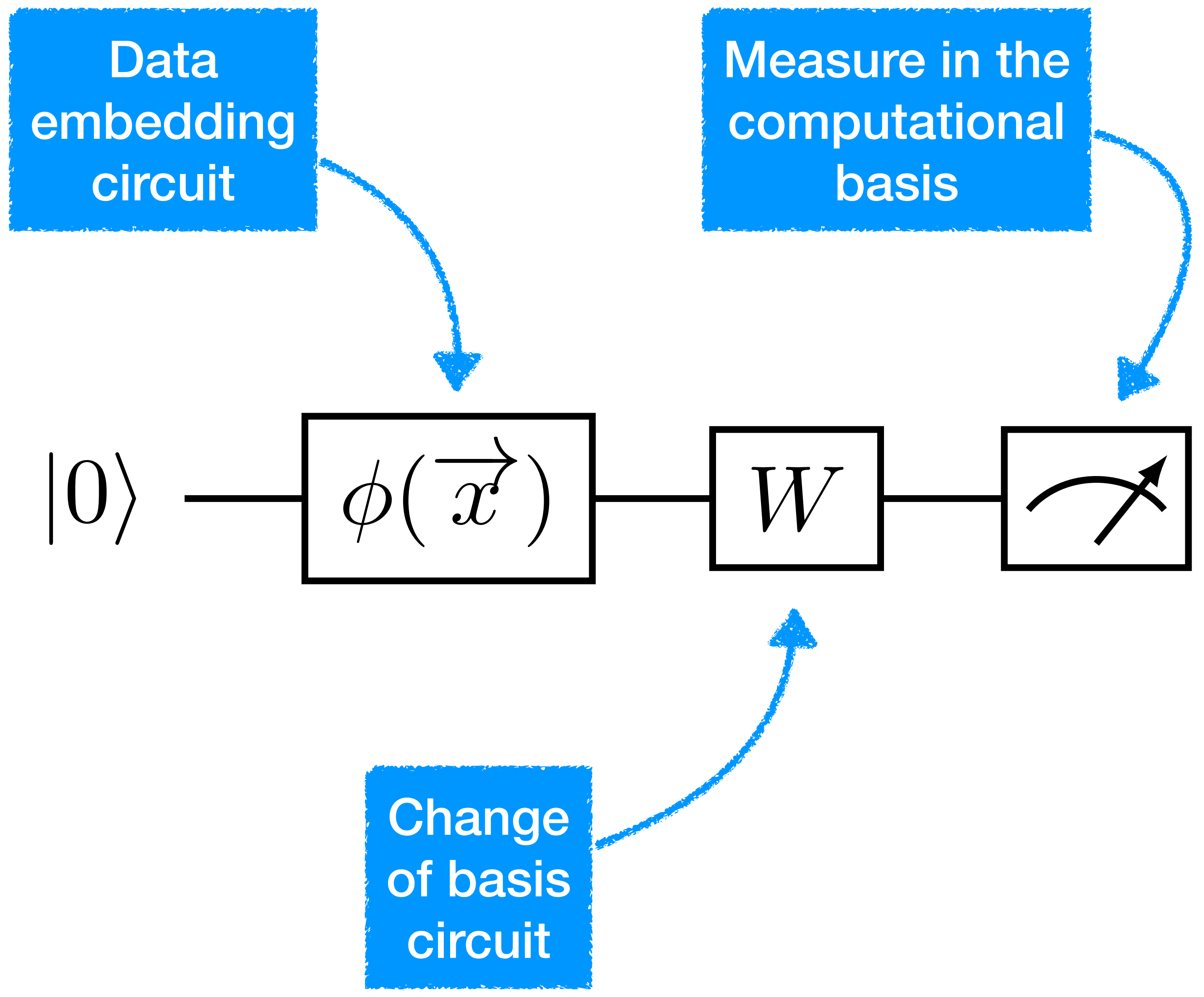
Distinguishable states



- Computational basis in which measurements are made corresponds to the Z basis.
- Measuring along the Z axis does not allow to distinguish between classes 0 and 1.
- If somehow we could measure along the ***X* axis**, both classes would be easily separable…
- Or rotate the state of the qubit before making the measurement!

Quantum classifier

Measure along a different axis



- Computational basis in which measurements are made corresponds to the Z basis.
- Measuring along the Z axis does not allow to distinguish between classes 0 and 1.
- If somehow we could measure along the ***X* axis**, both classes would be easily distinguishable…
- Or rotate the state of the qubit before making the measurement!
- Three components of a quantum classifier

Quantum classifier

Lab 3

Objectives

- Classify data in quantum feature space using 1 qubit
    - Measurement along X,Y,Z axis

Exercice

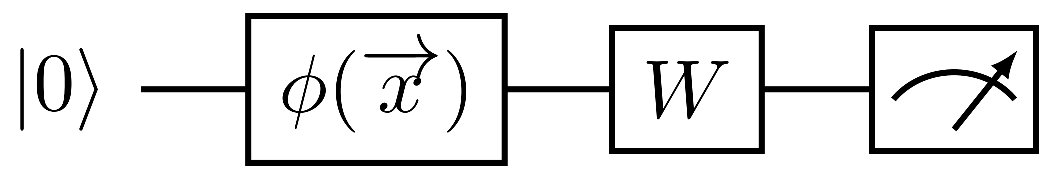
- Modify data embedding circuit and measurement axis

Colab link Lab3

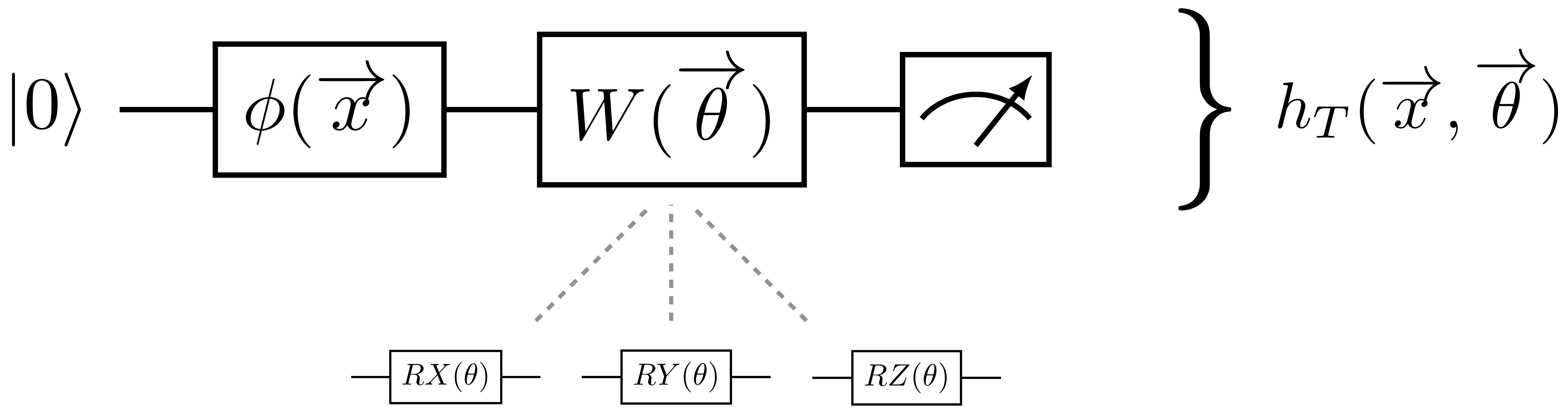
Training a quantum classifier

Learning the optimal rotation circuit

- In general, we don't know the form of the *W* circuit.



- We include rotation gates with free parameters to "learn" the optimal measurement.

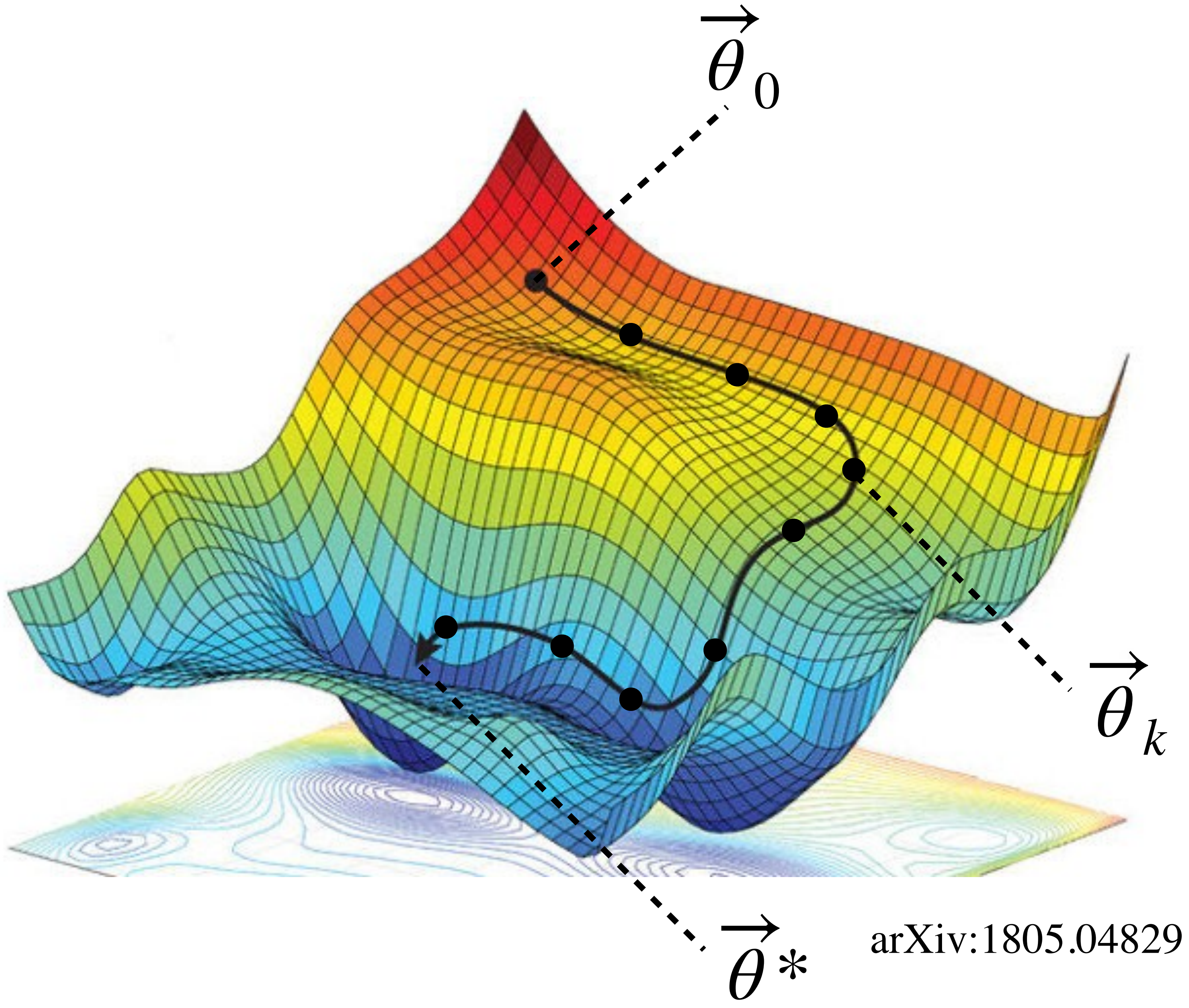


The *optimal measurement* is the one that minimizes the classification error.

Training a quantum classifier

Variational methods

- Variational methods are a technique for the **approximation** of complicated probability distributions.
- Principle:
    1. Start with a "*trial rotation*", or *ansatz*, that depends on ajustable parameters $\vec{\theta}_0$
    2. Iteratively update the parameters to find $\vec{\theta}*$ that minimizes the classification error.



arXiv:1805.04829

Training a quantum classifier

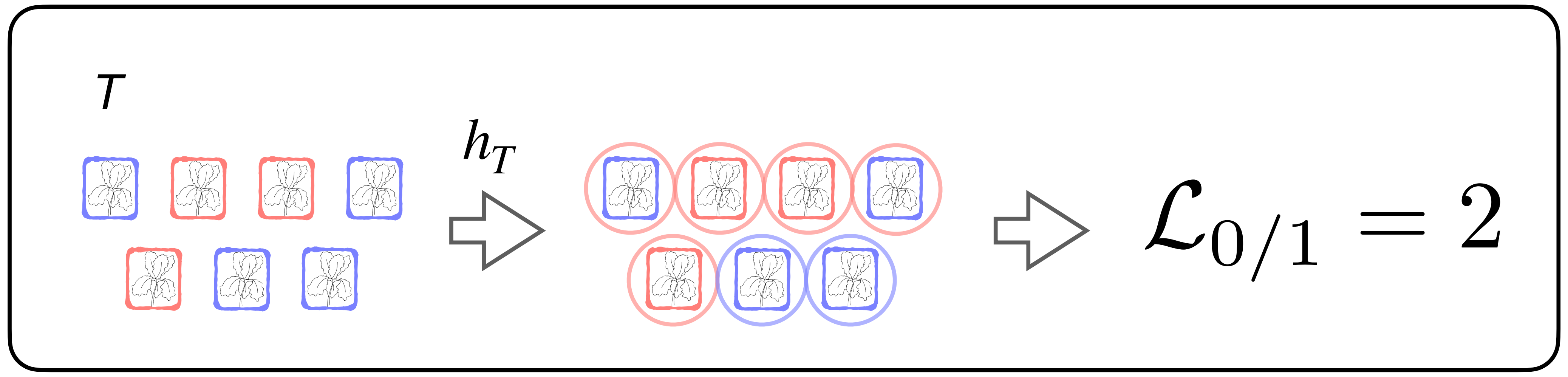
Loss function

- The **objective function** - or **loss function** - is a measure of the error between the output of our model and the target value

$$\mathcal{L}(\vec{x}, h_T(\vec{x}, \vec{\theta}), y)$$

- <u>Example</u>

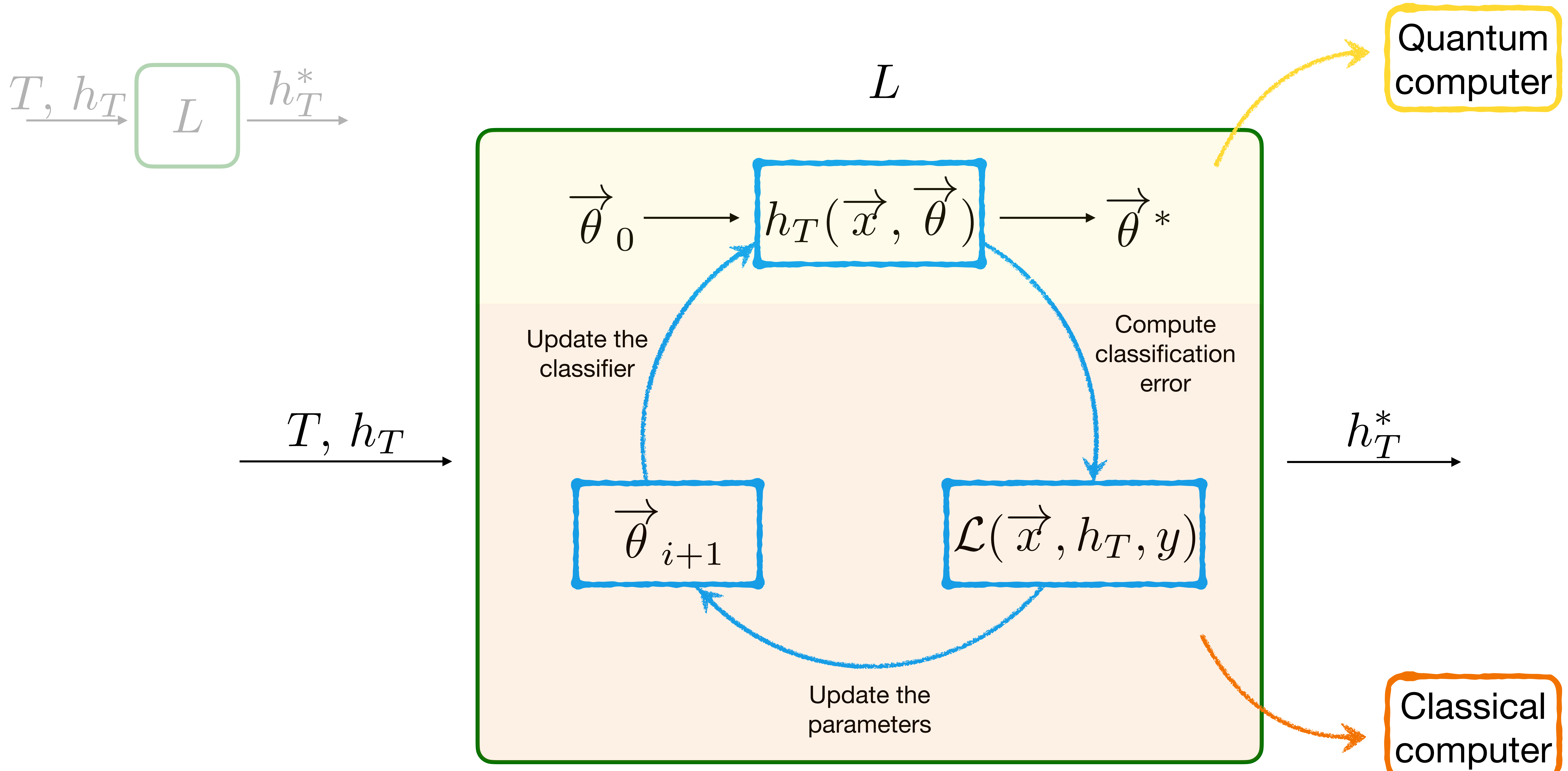
$$\mathcal{L}_{0/1} = \sum_{(\vec{x},y)\in T} 1_{h_T(\vec{x},\vec{\theta})\neq y}$$



- **Training** a classifier consists in a minimization of the loss function computed over the **training set**, in hope that the trained model will generalize well on the **test set**.

Training a quantum classifier

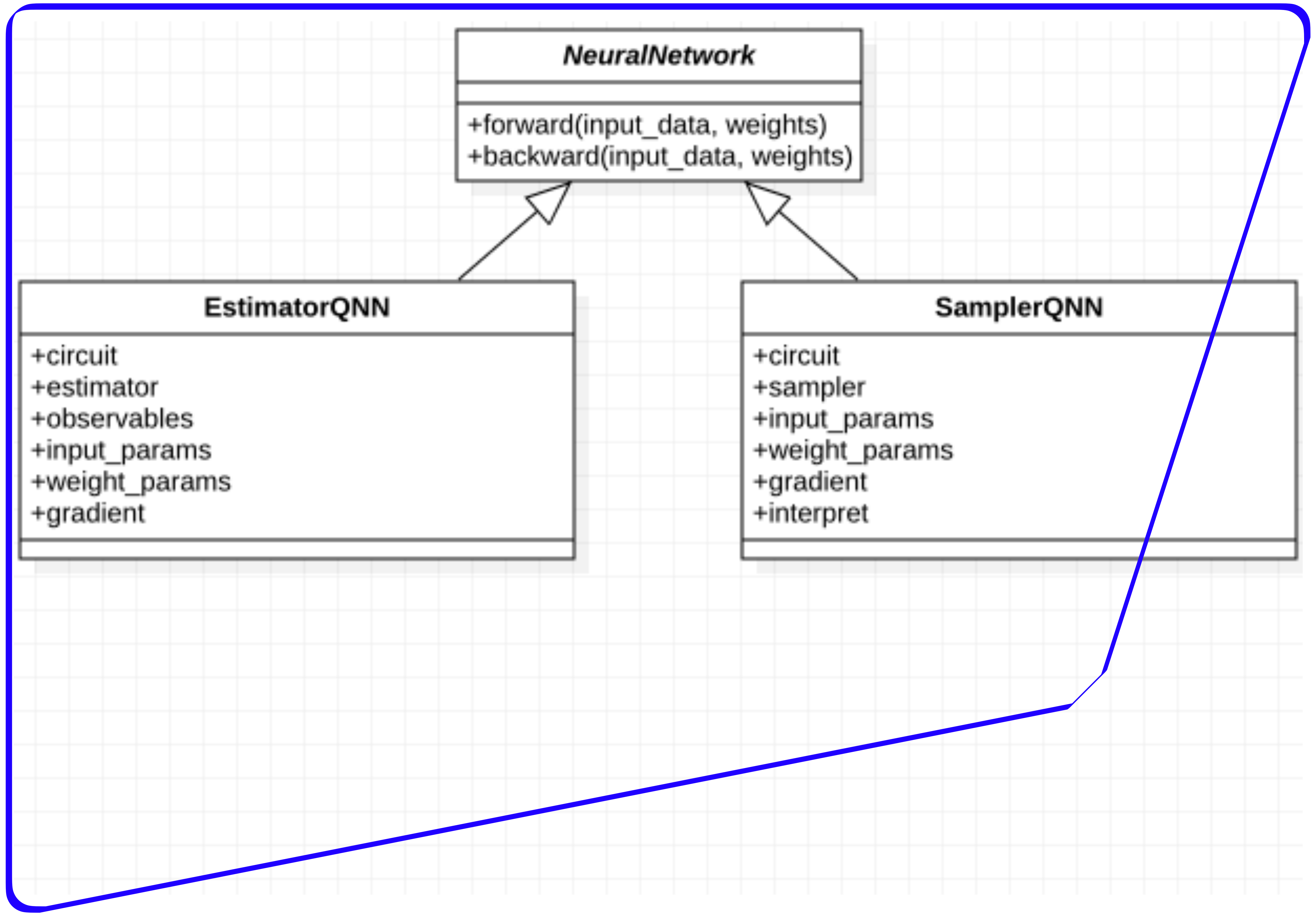
Learner (*learning algorithm*)



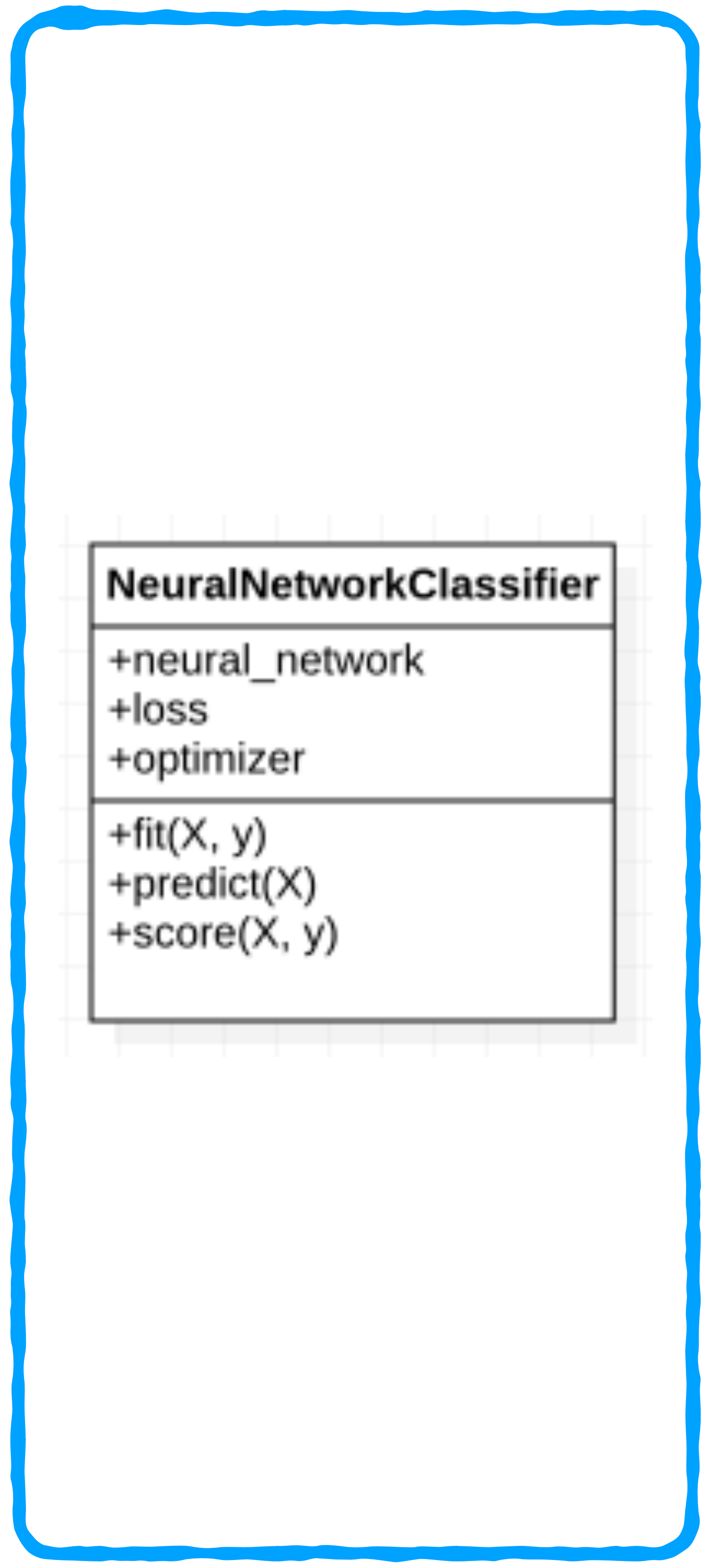
Training a quantum classifier

Qiskit machine learning library (version > 0.5)

Definition of a quantum model for classification



Training the quantum model



Training a quantum classifier

Lab 4

Objectives

- Building a parametrized circuit
    - Angle embedding
    - Variational quantum circuit (ansatz)
- Classification of the Iris dataset

Exercice

- Modify data embedding circuit

Colab link Lab4

Questions?

Olivier Nahman-Lévesque
olivier.nahman-
levesque@usherbrooke.ca

Test

CLICK HERE **to access the test**
You have 24 hours to do it