

量子機械学習

- 量子古典ハイブリッドによる機械学習の実践 -

次世代システム研究室

過去の研究発表

- ・ 量子アニーリングによる組み合わせ最適化問題

GMO INTERNET

Clip slide

量子コンピュータ時代へ：
現在の量子コンピュータで何ができるか

日本を代表する総合インターネットグループへ GMO

This is a slide thumbnail with a white background. At the top left is the 'GMO INTERNET' logo. At the top right is a blue button with a white icon and the text 'Clip slide'. The main text is centered and reads '量子コンピュータ時代へ：' followed by '現在の量子コンピュータで何ができるか'. At the bottom, there is a blue bar with the text '日本を代表する総合インターネットグループへ GMO'.

GMO INTERNET

量子コンピューターを用いて
金融分野の最適化課題の解決

日本を代表する総合インターネットグループへ GMO

This is a slide thumbnail with a white background. At the top left is the 'GMO INTERNET' logo. The main text is centered and reads '量子コンピューターを用いて' followed by '金融分野の最適化課題の解決'. At the bottom, there is a blue bar with the text '日本を代表する総合インターネットグループへ GMO'.

- ・ ゲート式量子コンピュータによる量子計算

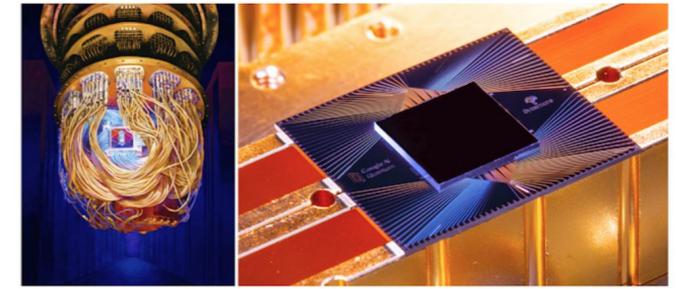
GMO INTERNET

実践量子コンピュータ
～IBM Qiskitを使って、量子計算してみよう～

次世代システム研究室

日本を代表する総合インターネットグループへ GMO

This is a slide thumbnail with a white background. At the top left is the 'GMO INTERNET' logo. The main text is centered and reads '実践量子コンピュータ' followed by '～IBM Qiskitを使って、量子計算してみよう～'. At the bottom right, it says '次世代システム研究室'. At the bottom, there is a blue bar with the text '日本を代表する総合インターネットグループへ GMO'.



量子超越性の実証

Quantum Supremacy Using a Programmable Superconducting Processor

**スーパーコンピュータで
1万年の計算をわずか200秒で**

Wednesday, October 23, 2019

Posted by John Martinis, Chief Scientist Quantum Hardware and Sergio Boixo, Chief Scientist Quantum Computing Theory, Google AI Quantum

Google AI Blog **量子機械学習 library の発表**

The latest news from Google AI

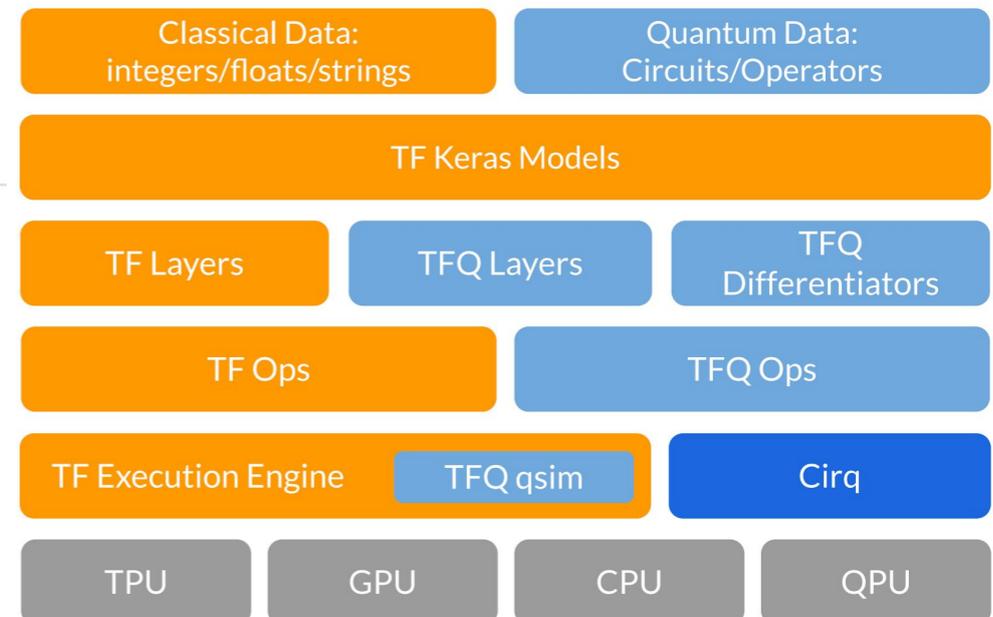


Announcing TensorFlow Quantum: An Open Source Library for Quantum Machine Learning

Monday, March 9, 2020

Posted by Alan Ho, Product Lead and Masoud Mohseni, Technical Lead, Google Research

"Nature isn't classical, damnit, so if you want to make a simulation of nature, you'd better make it quantum mechanical." – Physicist [Richard Feynman](#)



**機械学習+ 量子コンピュータの研究開発が加速している
⇒ 量子機械学習とは、どのような技術なのか紹介したい**

アジェンダ

- 量子コンピュータとは？
- ゲート式量子コンピュータをさわってみる
- 実践 1
 - 量子アルゴリズムを実装・量子テレポーテーション
- 実践 2
 - 量子機械学習・量子回路で分類問題を解く

量子コンピュータとは？

古典的粒子と量子

我々の日常 = “古典的”

量子の世界

粒子は“そこ”に存在する



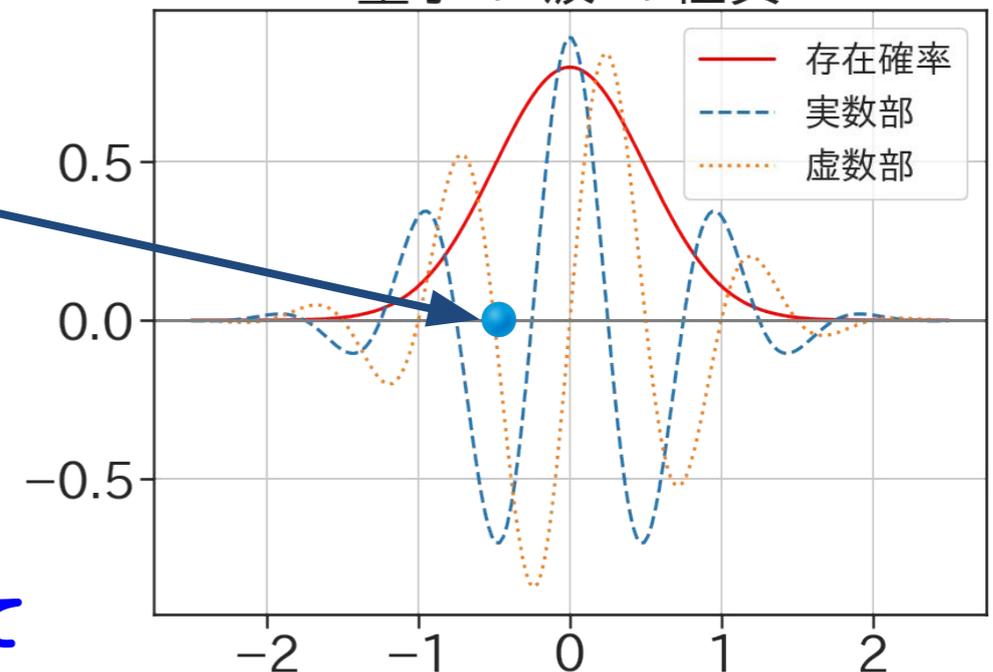
粒子は
複素数の波(=波動関数)

どこに存在するかは**未確定**
波の強さが“そこ”に存在する確率

観測者 測定して初めて
“位置”が確定



量子の“波”の性質



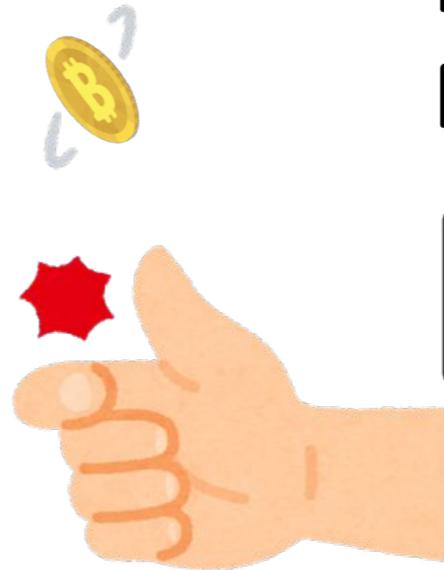
ミクロの世界はこのような不思議な性質を持っていて
これらの量子の性質を調べる理論を“量子力学”といいます

量子重ね合わせ(superposition)

量子は複数の状態を同時に取れる

例えるなら

「回転中のコイン」

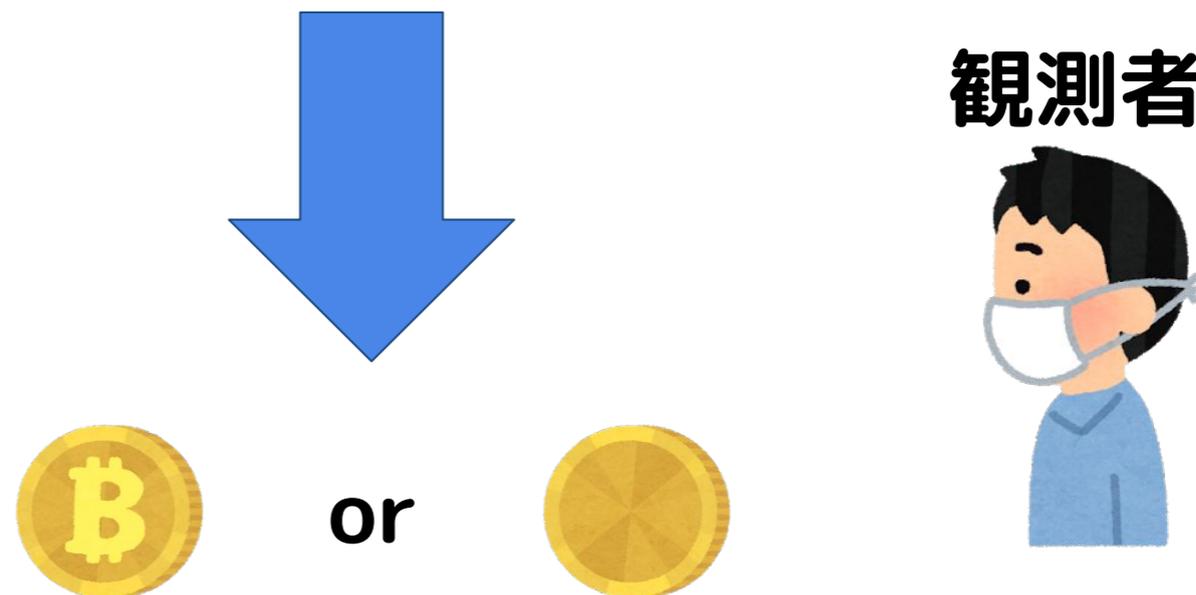


「表」でもあり
「裏」でもある状態

$$|\text{表}\rangle + |\text{裏}\rangle$$

1つの状態のみが
ランダムに観測され
重ね合わせ状態は
破壊される

⇒ 波動関数の収縮



観測者

量子エンタングルメント

例) 2枚のコインの**表同士**・**裏同士**で“もつれた状態”

$$|\text{表}\text{表}\rangle + |\text{裏}\text{裏}\rangle$$

α

コインA

コインB

観測者A

「表」

Aが測定された瞬間、
Bの測定結果も確定

観測者B

「表」

β

コインA

コインB

観測者A

「裏」

Aが測定された瞬間、
Bの測定結果も確定

観測者B

「裏」

なぜ量子コンピュータが、注目をあつめているか

理化学研究所プレスリリースより

スーパーコンピュータ「富岳」Graph500において世界第1位を獲得
 –ビッグデータの処理で重要となるグラフ解析で最高レベルの評価–

理化学研究所（理研）、九州大学、株式会社フィックスターズ、富士通株式会社による共同研究グループは、スーパーコンピュータ「富岳」¹⁾による測定結果で、大規模グラフ解析に関するスーパーコンピュータの国際的な性能ランキングである「Graph500」において、世界第1位を獲得しました。

このランキングは、HPC（ハイパフォーマンス・コンピューティング：高性能計算技術）に関する国際会議「ISC2020」と同時期にGraph500 Committeeから6月22日（日本時間6月22日）に発表されました。

大規模グラフ解析の性能は、大規模かつ複雑なデータ処理が求められるビッグデータの解析において重要な指標となるもので、「富岳」は開発・整備中ながら2015年6月から9期連続第1位獲得の実績を持つスーパーコンピュータ「京」よりも2倍以上の能力を有することが実証されました。



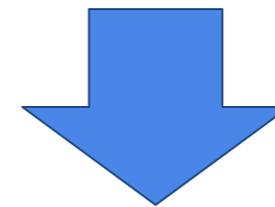
スーパーコンピュータ「富岳」（開発・整備中）

エクサスケールの時代へ

- ・ 毎秒100京の演算

しかし、従来の古典的コンピュータでは

- ・ 素子の微細化によるムーアの法則の限界
- ・ 発熱・電力消費量の問題



新たな技術革新が必要とされている

量子コンピュータの提唱

1981 R. Feynman, "Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy."

- ・ Shor のアルゴリズム(素因数分解による暗号解読への応用)など
 実用的な問題に対して古典コンピュータの効率を凌駕するアルゴリズムの発見

2種類の量子コンピューター

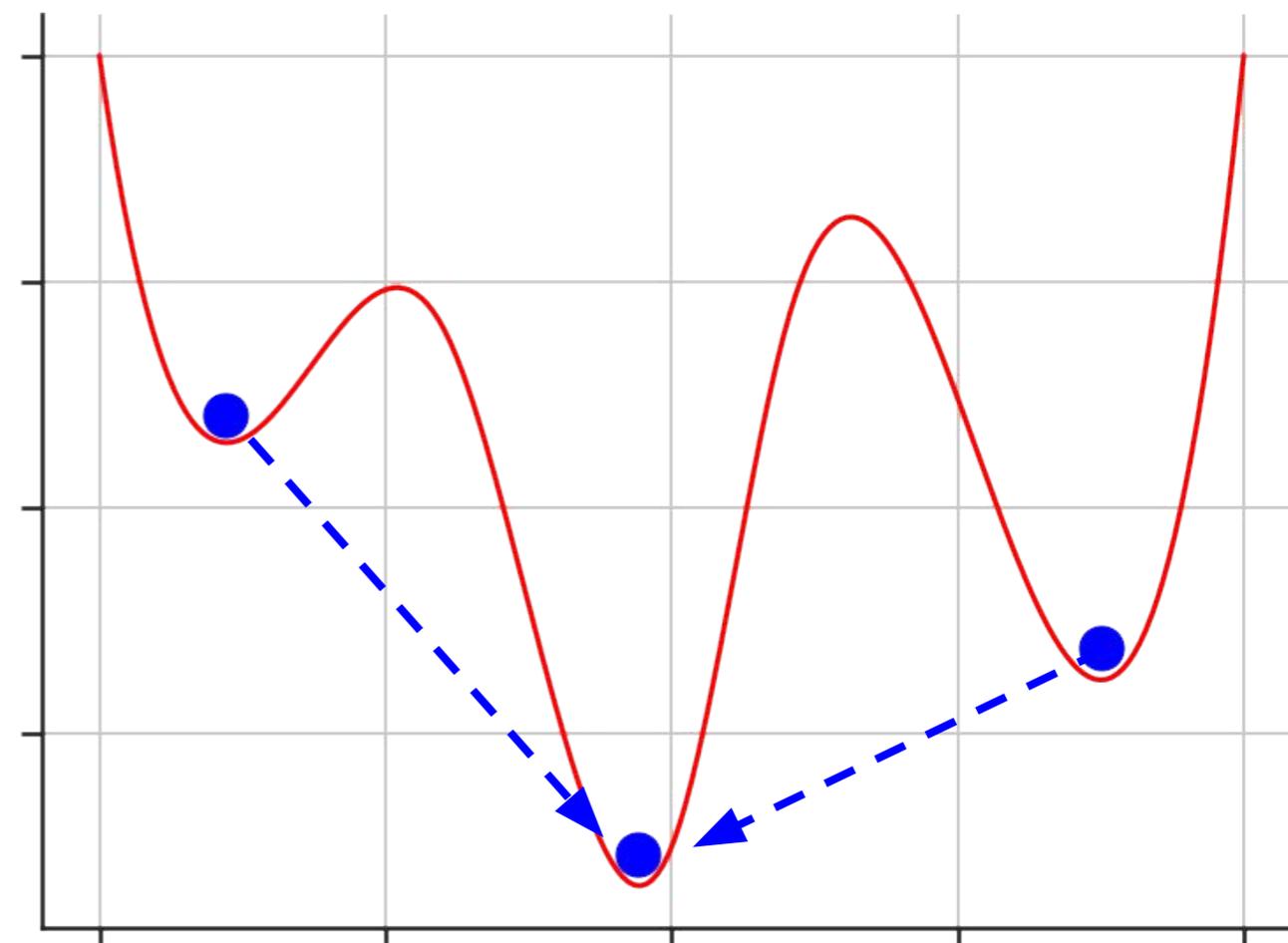


- アニール型
 - D-wave, NEC 他
 - 利点
 - 開発が(ゲート型と比較すると)簡単
 - 問題点
 - 解ける問題に制限がある
- ゲート型
 - IBM, Rigetti, Google 他
 - 利点
 - 汎用的な計算が出来る
 - 問題点
 - 開発が非常に難しい



アニーリング型

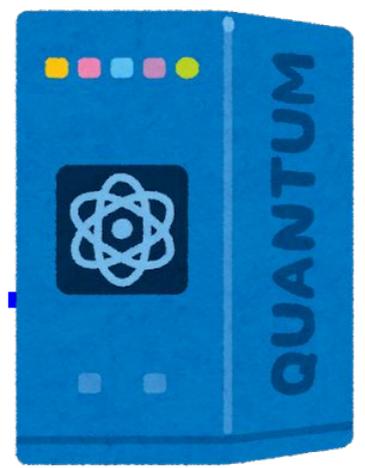
- 利点
 - 開発が(ゲート型と比較すると)簡単
- 問題点
 - 解ける問題に制限がある



量子のもつトンネル効果を利用して
最適な組み合わせ探索ができる

2種類の量子コンピューター

今回の話はこちら



- アニール型
 - D-wave, NEC 他
 - 利点
 - 開発が(ゲート型と比較すると)簡単
 - 問題点
 - 解ける問題に制限がある

- ゲート型
 - IBM, Rigetti, Google 他
 - 利点
 - 汎用的な計算が出来る
 - 問題点
 - 開発が非常に難しい



ゲート式量子計算って何？

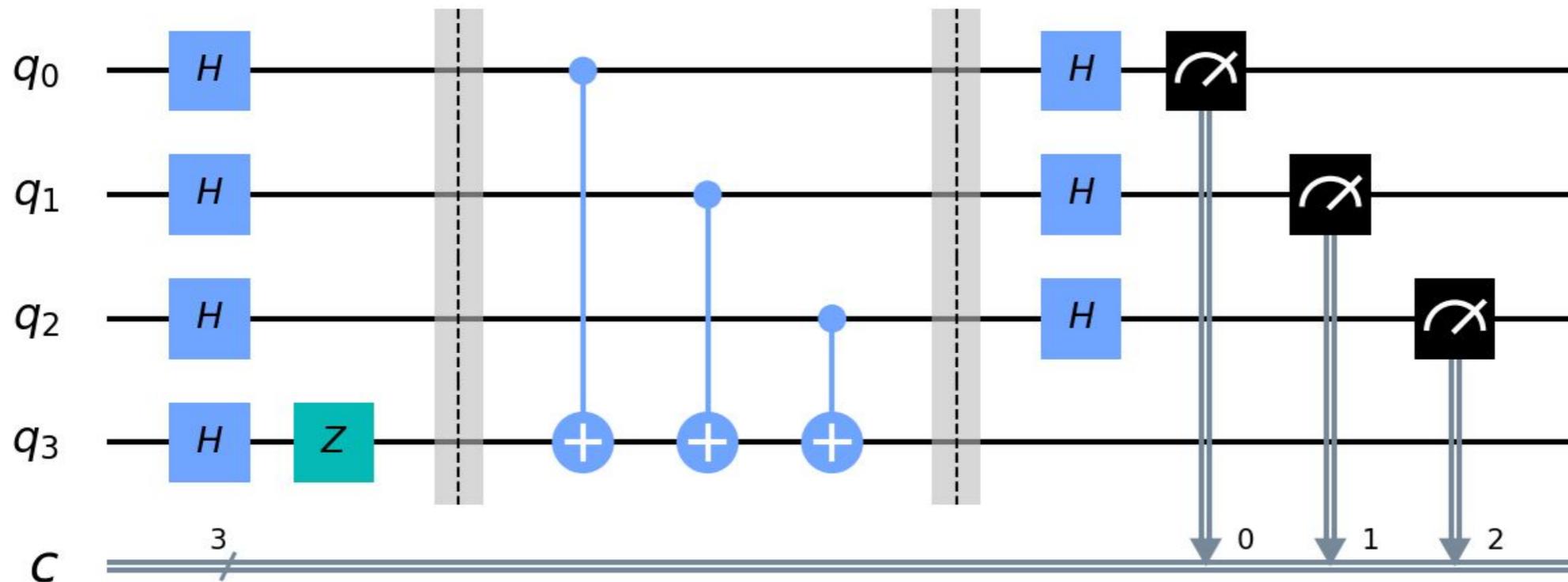
古典 bit

0 か **1** の一方の状態

量子 bit ⇒ **qubit**

0¹ の重ね合わせの状態

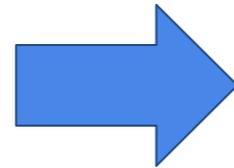
qubit をゲートで操作して計算



ゲート式量子コンピュータの作り方

“量子的に振る舞うモノ”

ならなんでもよい



- ・ 超伝導量子ビット
 - ・ Google/IBM/rigetti/Alibaba他
- ・ イオントラップ
 - ・ IonQ, Honeywell
- ・ 光量子ビット
 - ・ XANADU
- ・ 量子ドット
 - ・ intel
- ・ トポロジカル量子ビット
 - ・ Microsoft

など様々な実装が研究されており
それぞれに**長所・短所**がある

量子コンピュータは、なぜ強力なのか？

古典コンピュータ

n bit = 2^n 個の状態

0000...0

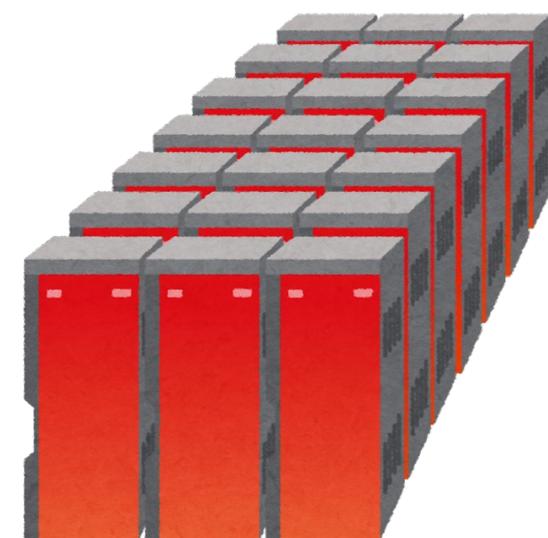
1000...0

0100...0

.

.

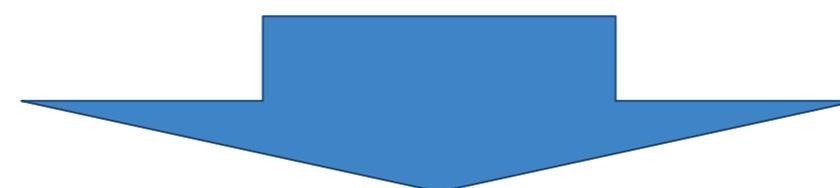
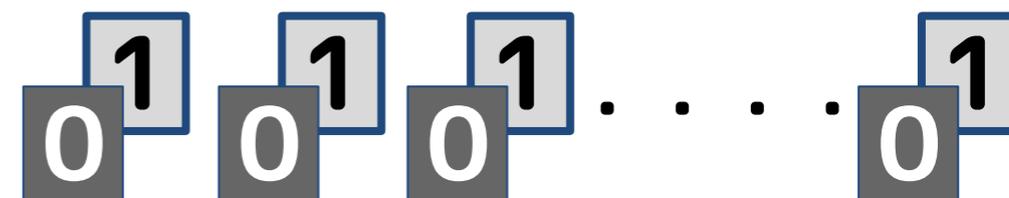
1111...1



逐次計算しないとイケない

量子コンピュータ

n qubit = 2^n 個の状態の重ね合わせ



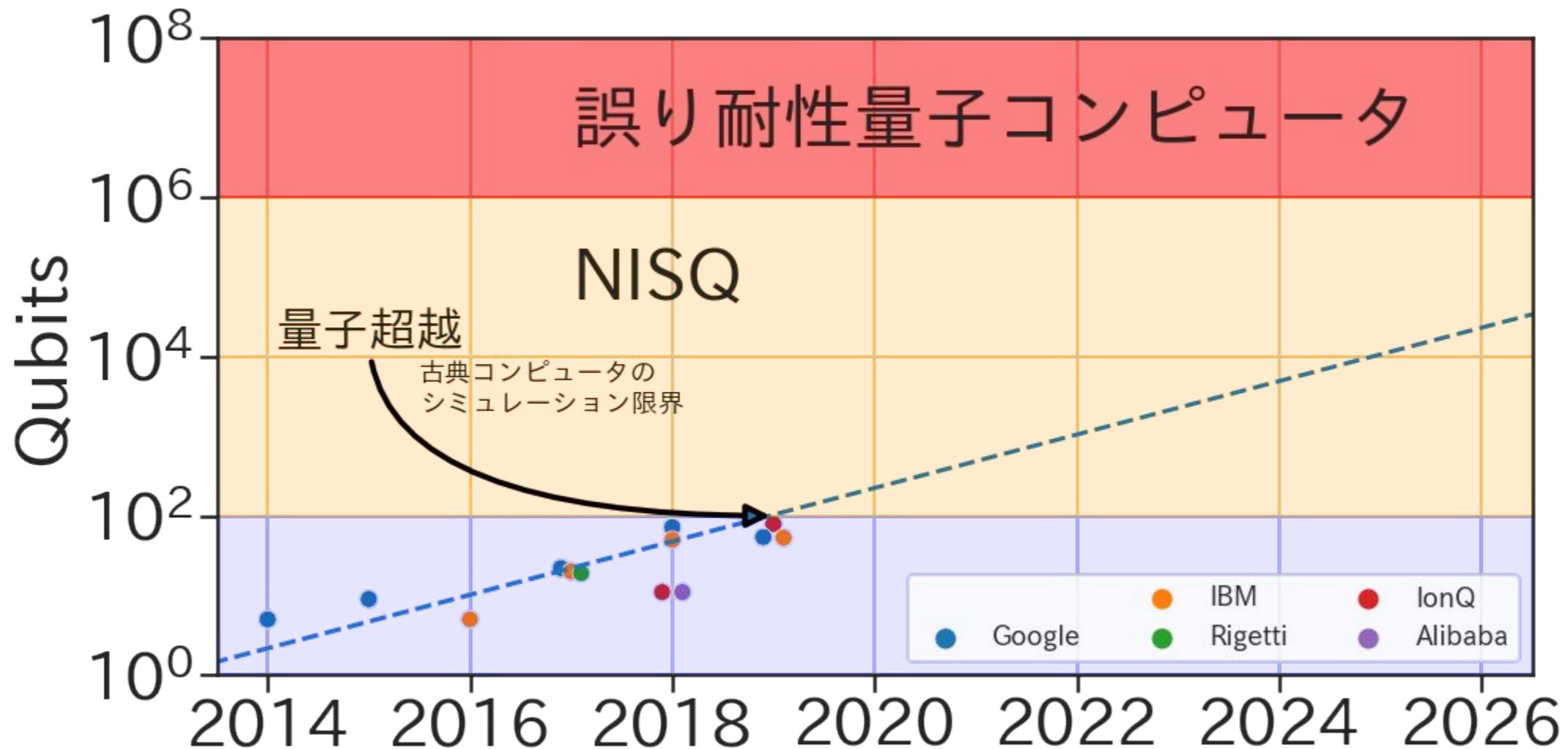
1度に同時に計算できる

注：正解を効率的に抽出する
アルゴリズムが必要

最悪の場合指数関数的に探索コストが増える
⇒ アルゴリズムの研究が不可欠

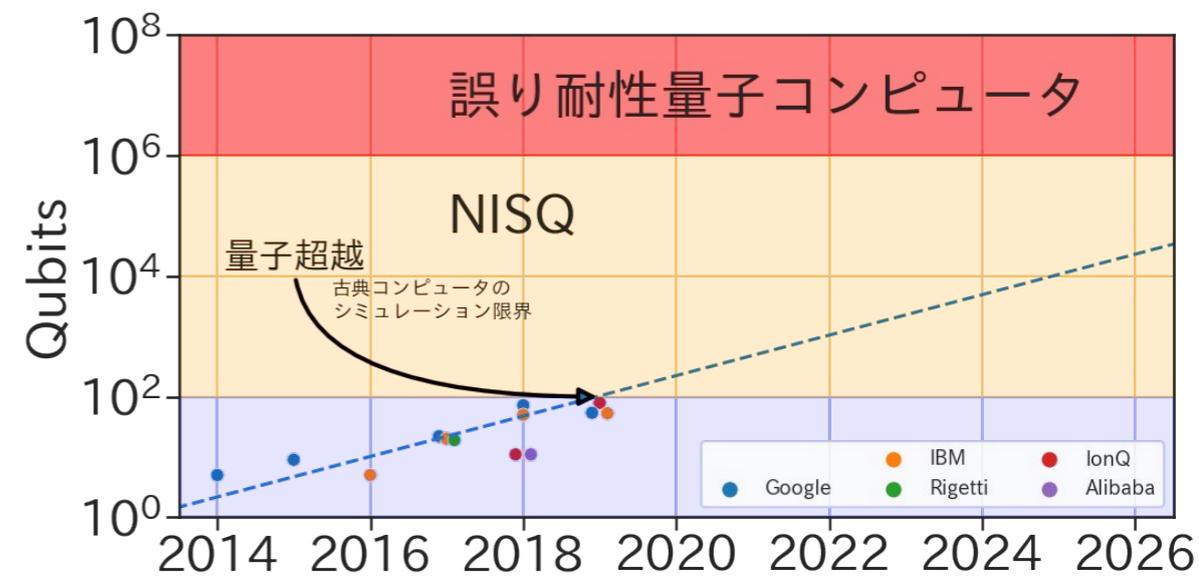
ゲート式量子コンピュータの発展

- **qubitの状態は非常に不安定** ⇒ 技術的に難しい
 - ようやく50~100 qubit ⇒ アニーリング式2048 qubit (D-wave 2000Q)
- NISQ (Noisy Intermediate Scale Quantum)
 - 特殊な問題について古典シミュレーション限界を突破
- 実用的なタスク ⇒ 素因数分解には1億 qubit は必要



NISQアルゴリズムの発展

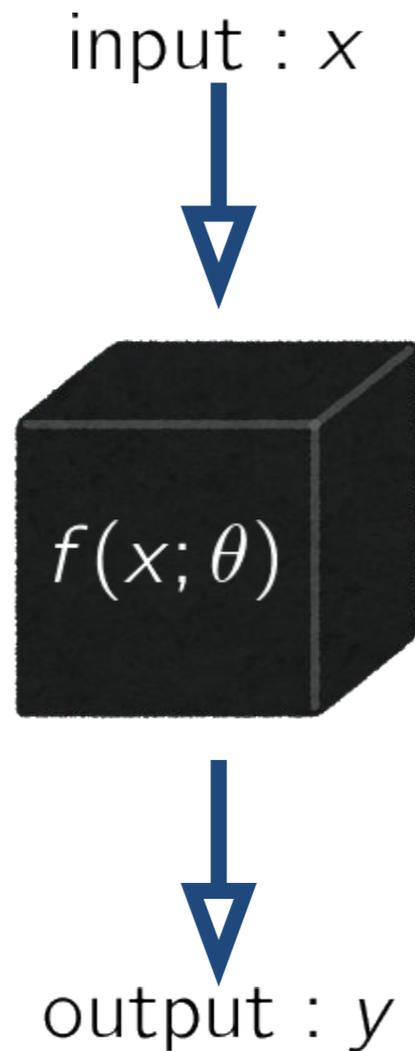
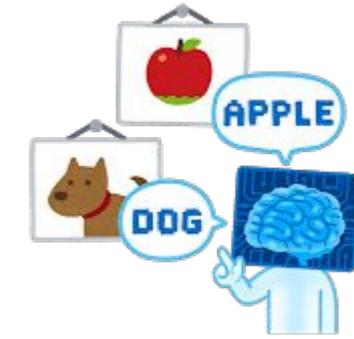
- 汎用量子コンピュータは **まだ無理**
- Noisy Intermediate Scale Quantum
⇒ **NISQでも利用できるアルゴリズム研究が進められている**
 - 短い回路・少ないqubitで何か出来ないか？



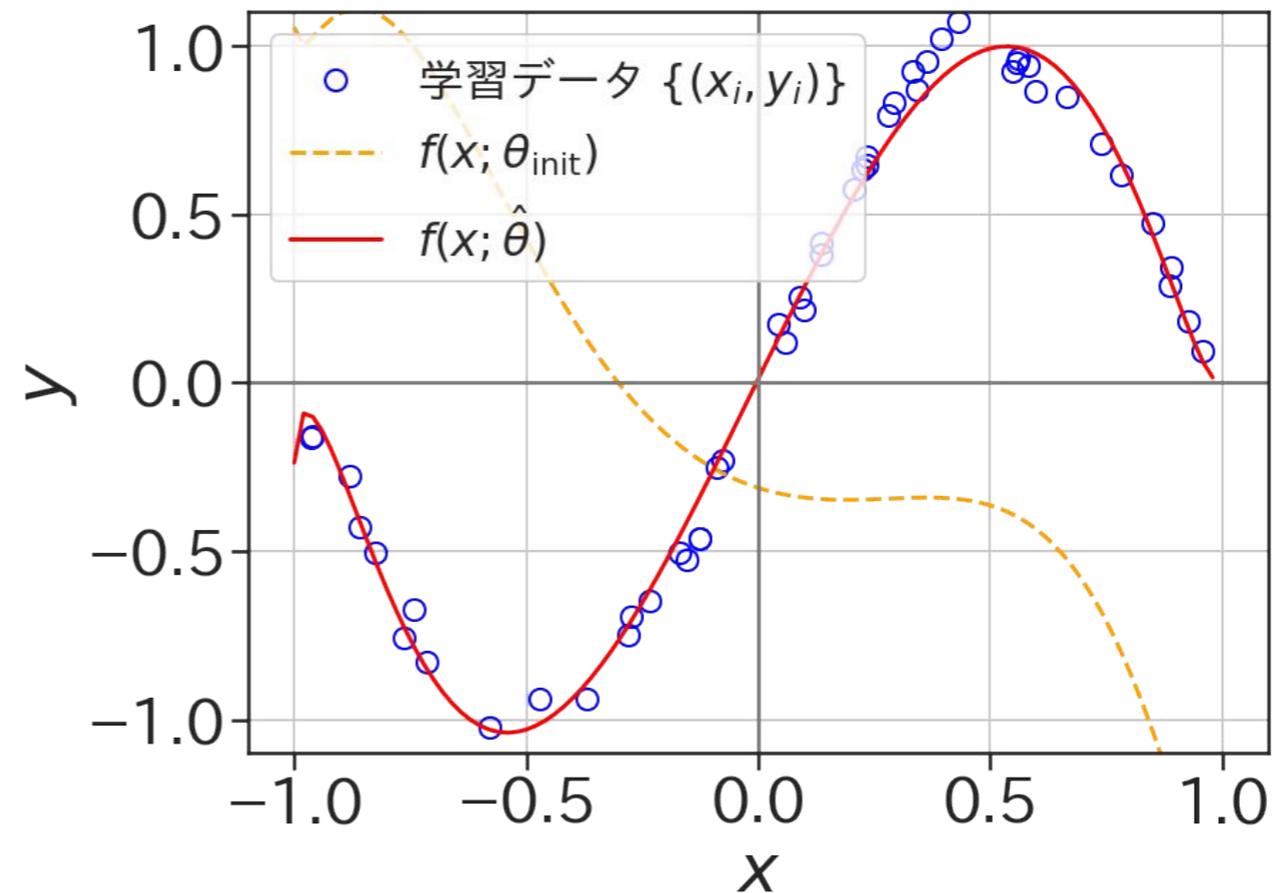
- Variational Quantum Eigensolver (VQE)
⇒ 量子化学シミュレーション
- Quantum Approximate Optimization Algorithm (QAOA)
⇒ 最適化問題
- Quantum Machine Learning (QML)
⇒ 量子機械学習

機械学習

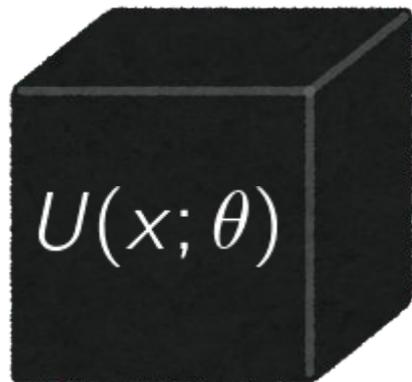
・ 教師あり学習



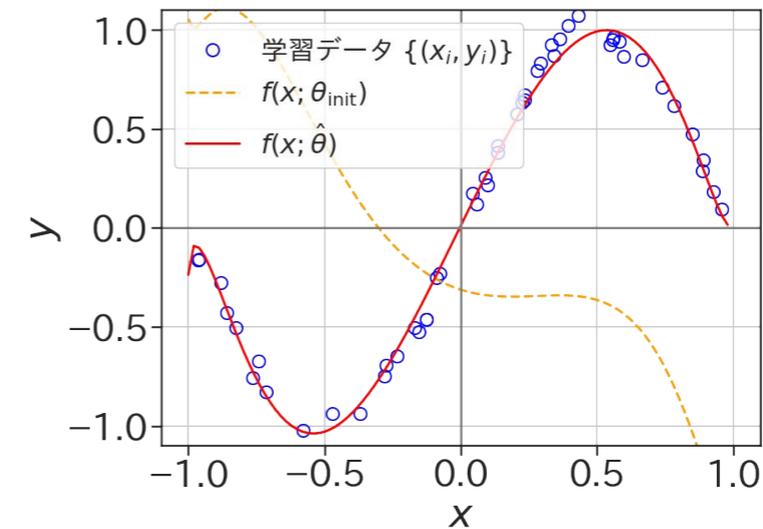
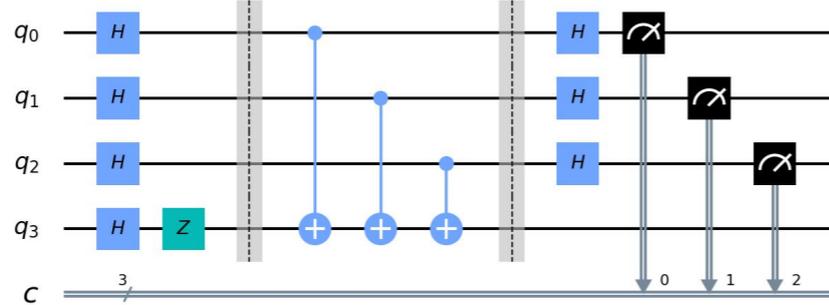
学習データを再現するように
数理モデル(f)のパラメータ(θ)を最適化



量子機械学習

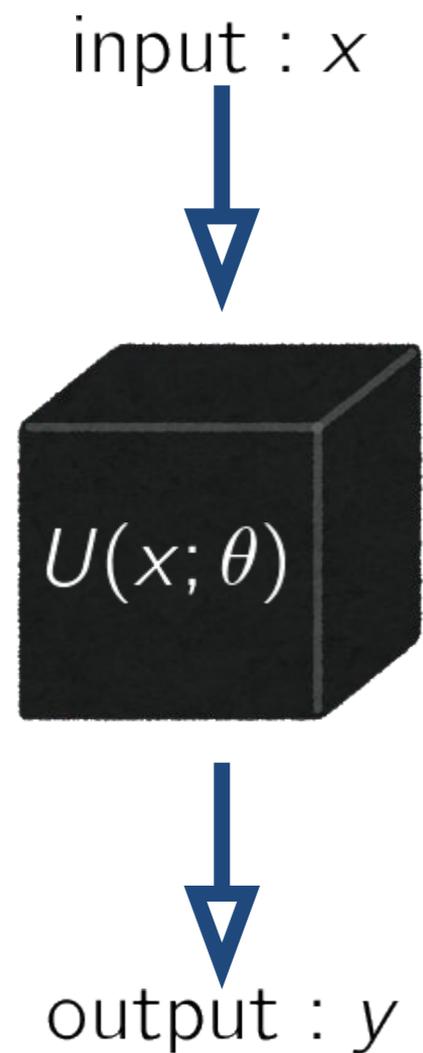
input : x output : y

学習データを再現するように
量子回路(U)のパラメータ(theta)を最適化



- ・ **量子回路 + 古典**のハイブリッド計算
- ・ qubitを用いることで、
指数的に高い特徴量空間を活用できる

量子機械学習・実装



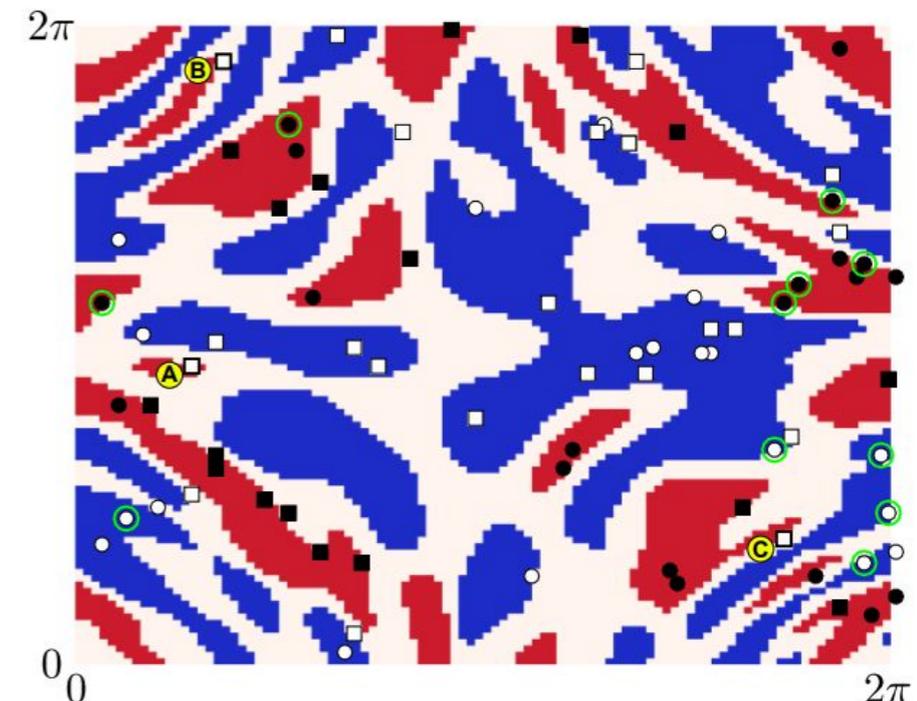
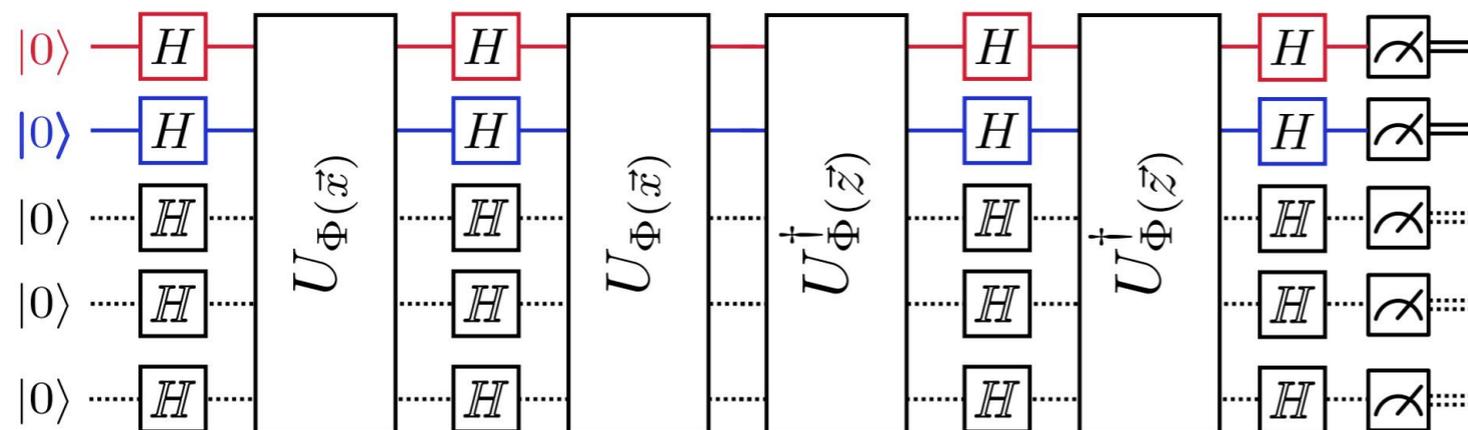
Letter | Published: 13 March 2019

Supervised learning with quantum-enhanced feature spaces

Vojtěch Havlíček, Antonio D. Córcoles , Kristan Temme , Aram W. Harrow, Abhinav Kandala, Jerry M. Chow & Jay M. Gambetta

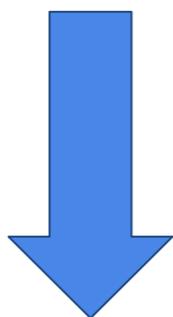
Nature **567**, 209–212(2019) | [Cite this article](#)

20k Accesses | **49** Citations | **255** Altmetric | [Metrics](#)

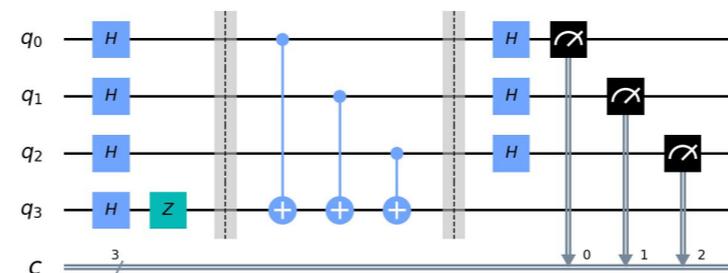


クラス分けタスクの結果

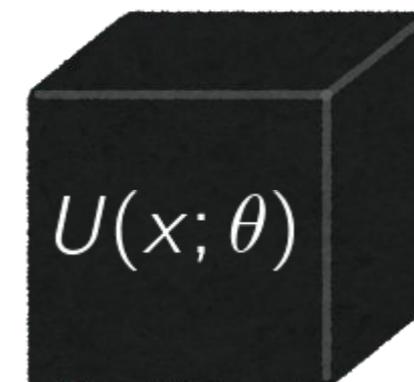
量子コンピュータは凄そうだけど、
結局、何やっているか分からない



まずは、
ゲート式量子コンピュータをさわってみる



input : x

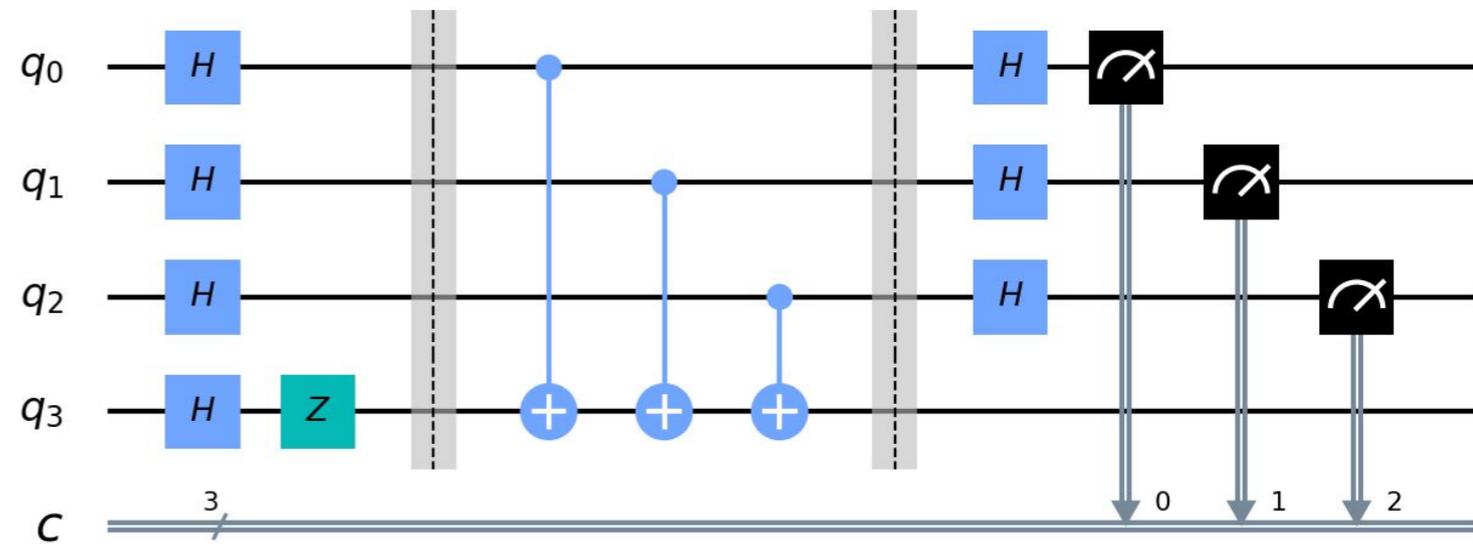


output : y

量子計算 Framework (Python)

ややこしいですが
 量子力学で動く集積回路
 ⇒ 古典コンピュータの計算
 ⇒ 量子回路をシミュレーションしています

- 量子計算をシミュレート
 & 実際の量子コンピュータで実行



Blueqat

Cirq

Qiskit

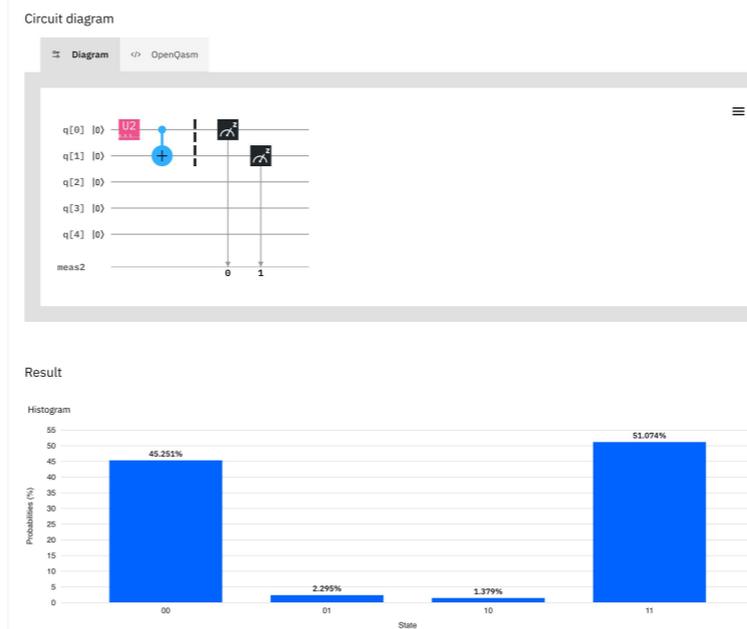
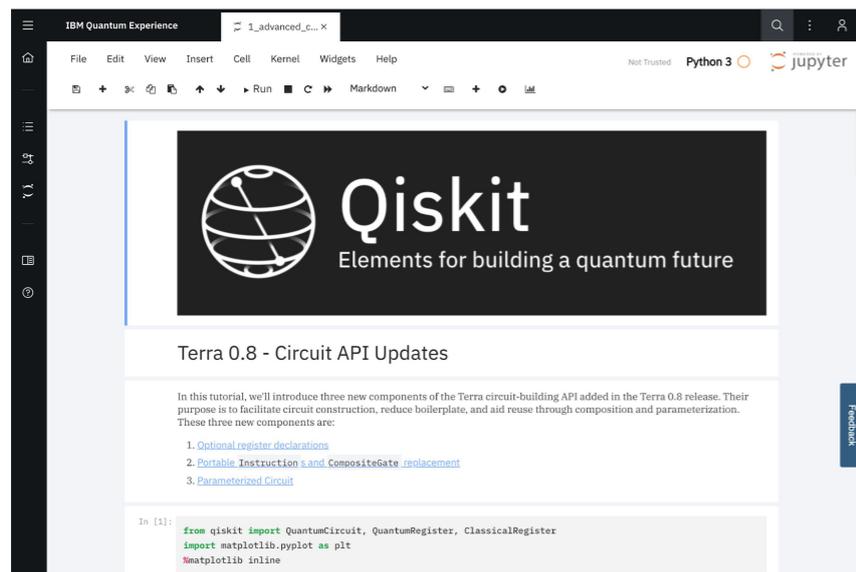
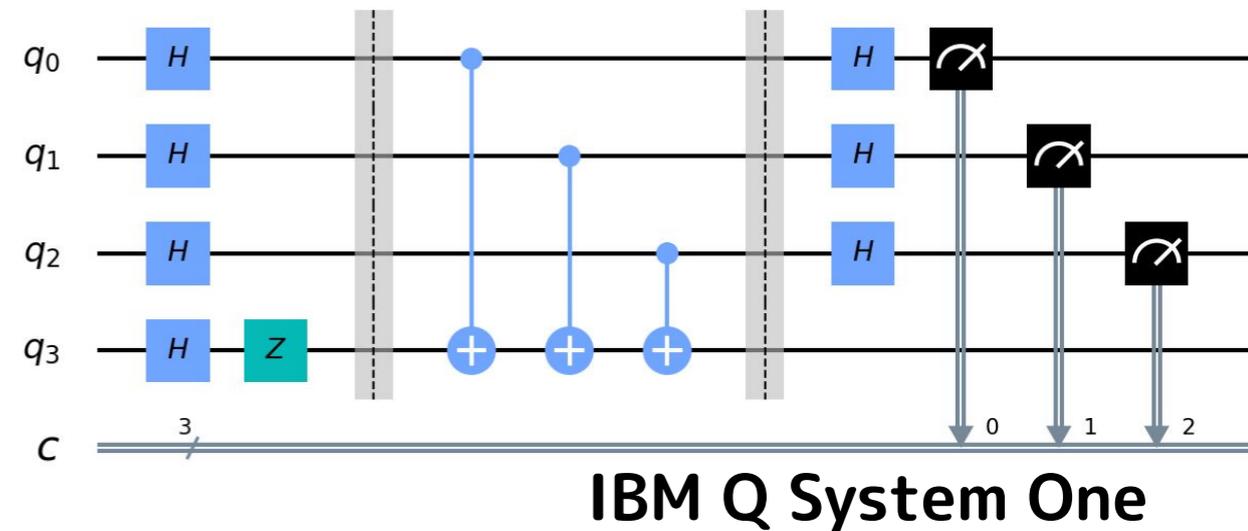
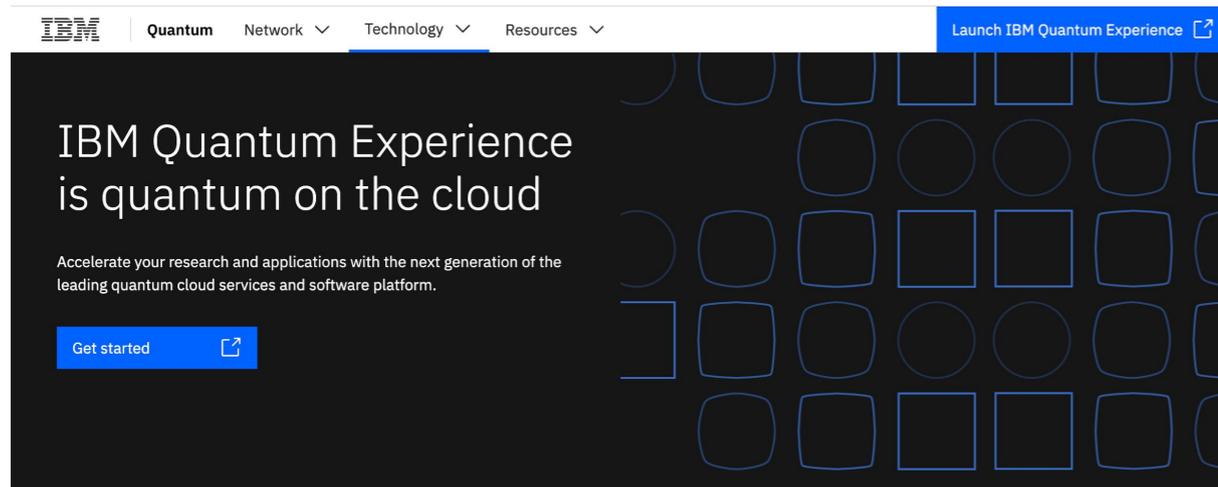
XANADU

PENNYLANE

IBM Quantum Experience

まずは、実際の量子コンピュータを動かしてみる

- 量子コンピュータ (IBM Q) で計算可能 (15 qubit まで)
- Cloud 上で実行可能
- Tutorial & Document が豊富



Qubit : 0 であり 1 でもある状態

$$|q\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|\alpha|^2 + |\beta|^2 = 1 \quad \alpha, \beta : \text{複素数}$$

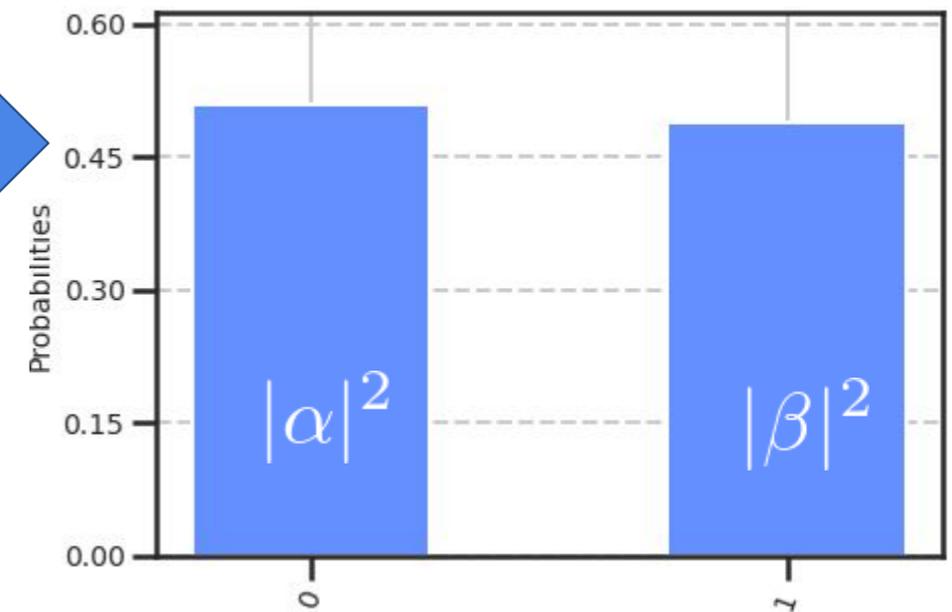
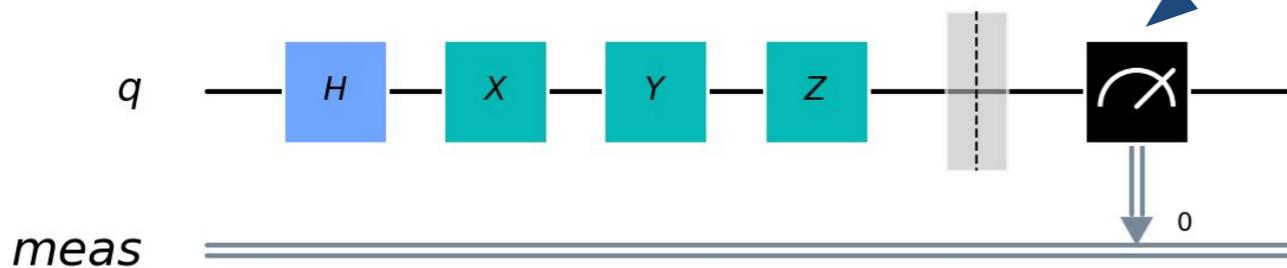
・ 測定 : 重ね合わせの破壊

0 の確率 $|\alpha|^2$

1 の確率 $|\beta|^2$

・ 量子ゲートの作用 (= 0, 1 の混合を操作)

$$U : |q\rangle \rightarrow \alpha'|0\rangle + \beta'|1\rangle$$



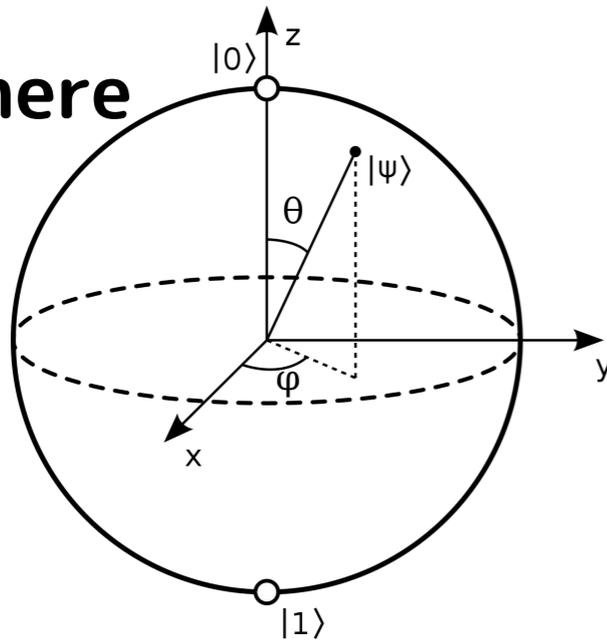
これらの計算はベクトル・行列で表現される

$|?\rangle$: ケット・ベクトル $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

$$|q\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad U|q\rangle = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

Qubit の表し方(その2)

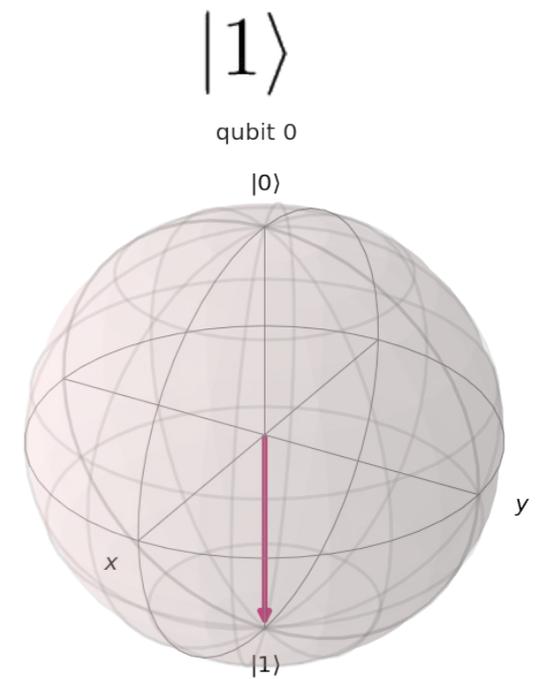
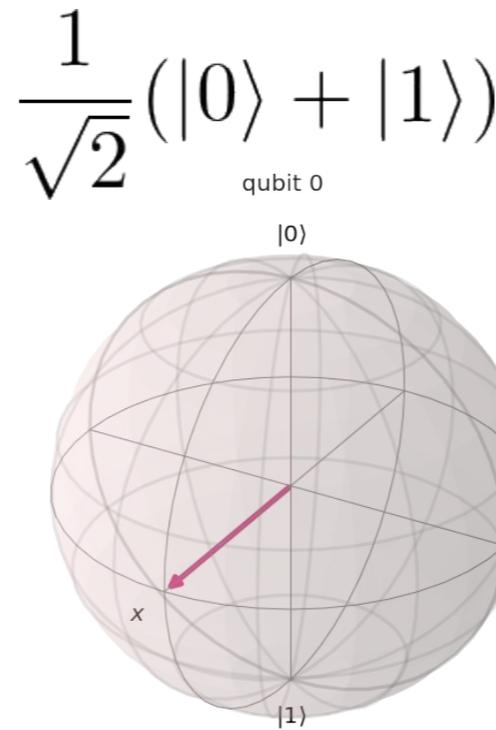
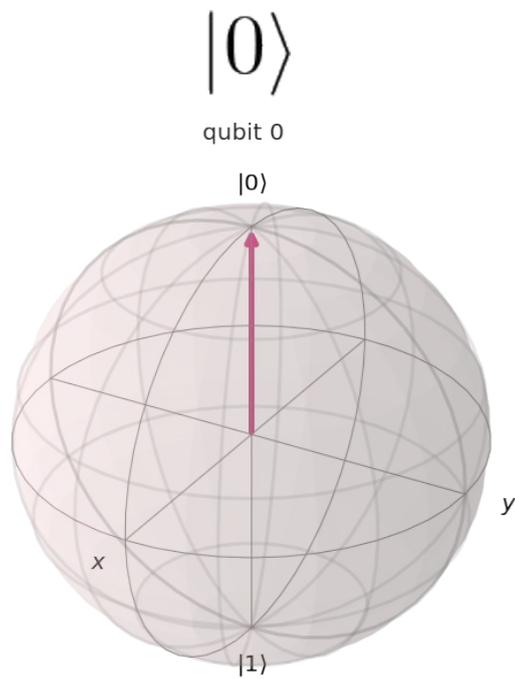
Bloch Sphere



$$|q\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad |\alpha|^2 + |\beta|^2 = 1$$

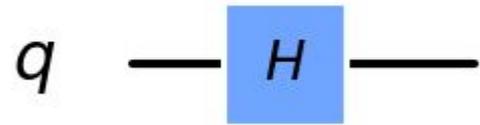
$$= \cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle$$

qubit は球面の1点としても表現できる



⇒ ゲート操作はこのベクトルの向きを回転させることに対応

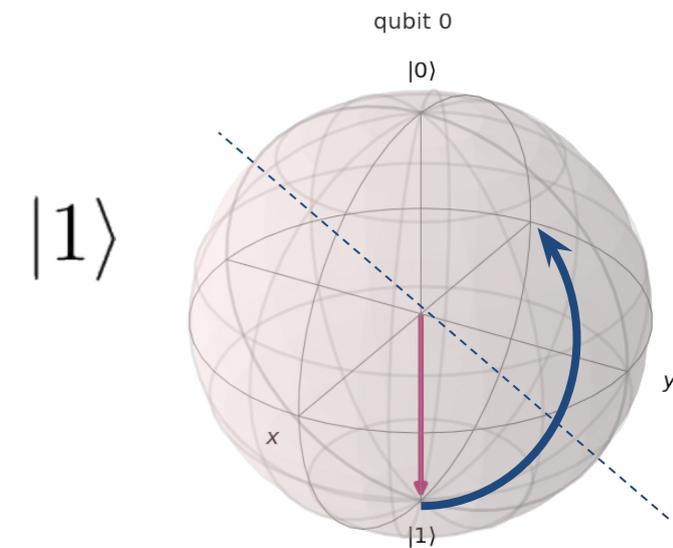
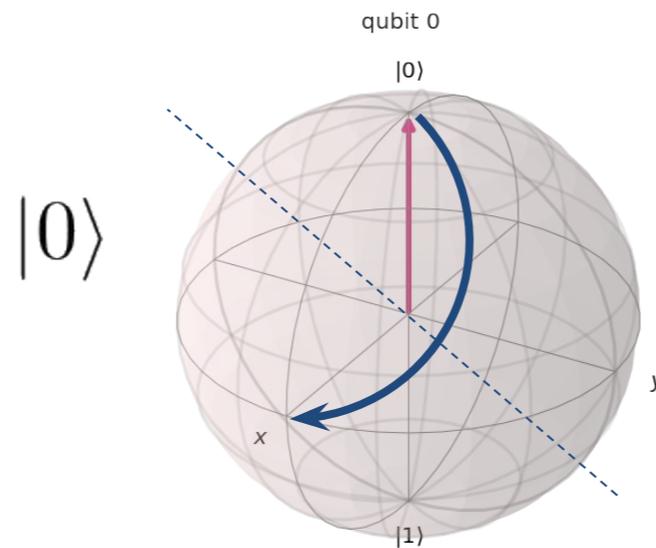
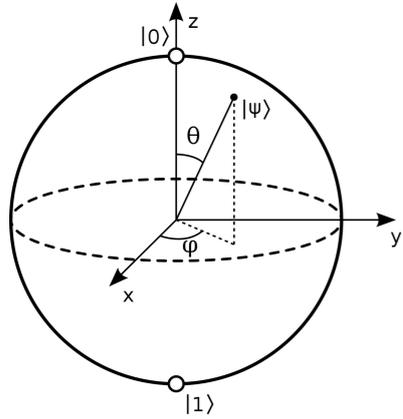
量子ゲートの例1：アマダール(Hamard)・ゲート



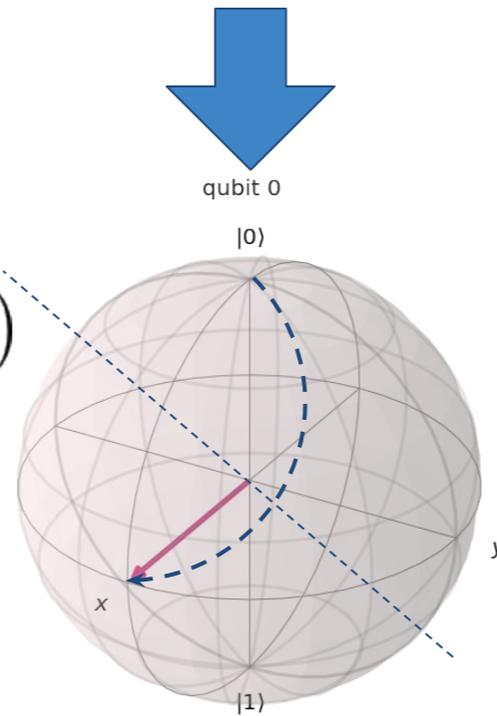
量子重ね合わせ状態を作る

xz軸回りに180度回転

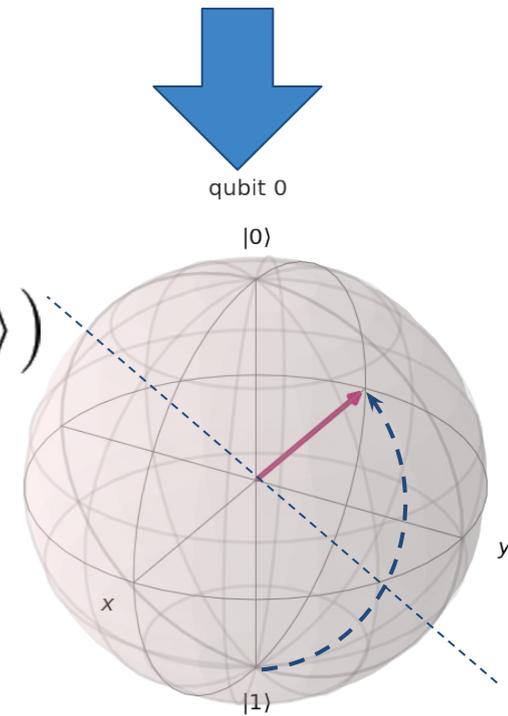
$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$



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

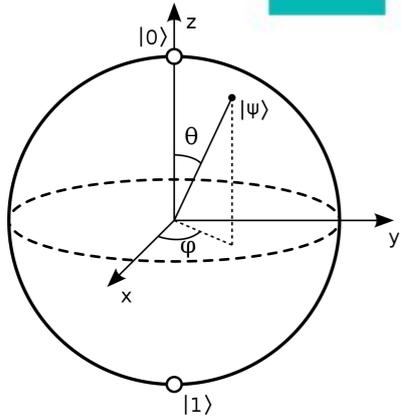
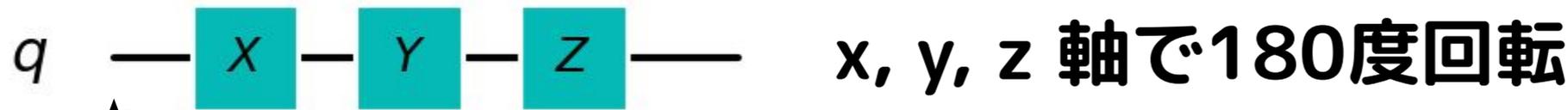


$$H|1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

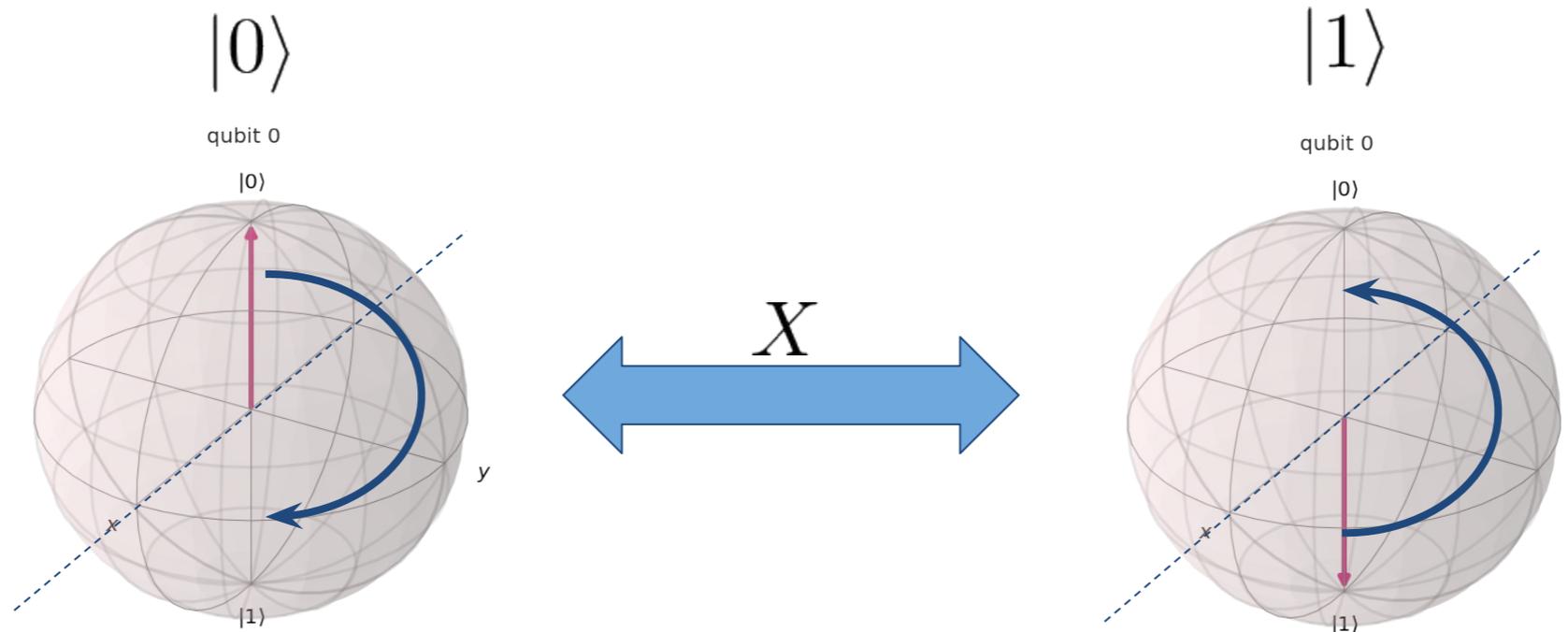


量子ゲートの例2：パウリ・ゲート

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$



特に X gate は NOT gate として利用される ($0 \leftrightarrow 1$)



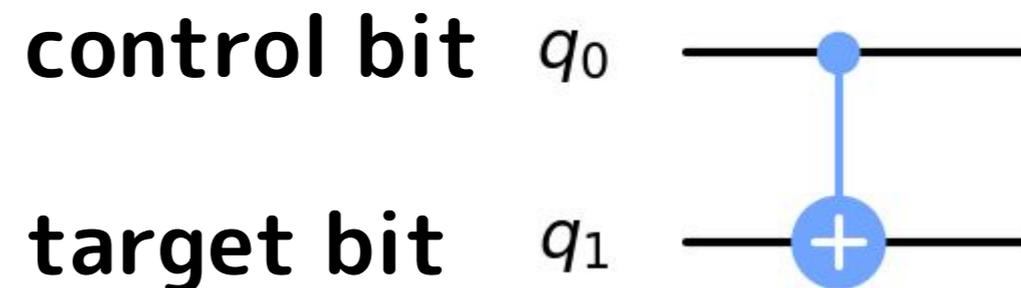
$$X|1\rangle = |0\rangle$$

$$X|0\rangle = |1\rangle$$

量子ゲートの例3 : Controlled-NOT (CNOT)

エンタングルメントを作る

$$C_{\text{NOT}}|q_1\rangle \otimes |q_0\rangle = |q_1 \oplus q_0\rangle \otimes |q_0\rangle$$



control	target	control	$q_0 \otimes q_1$
q_0	q_1	q_0	q_1
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

control bit = 1 のとき
target bit を 反転

量子エンタングルメントをつくってみる



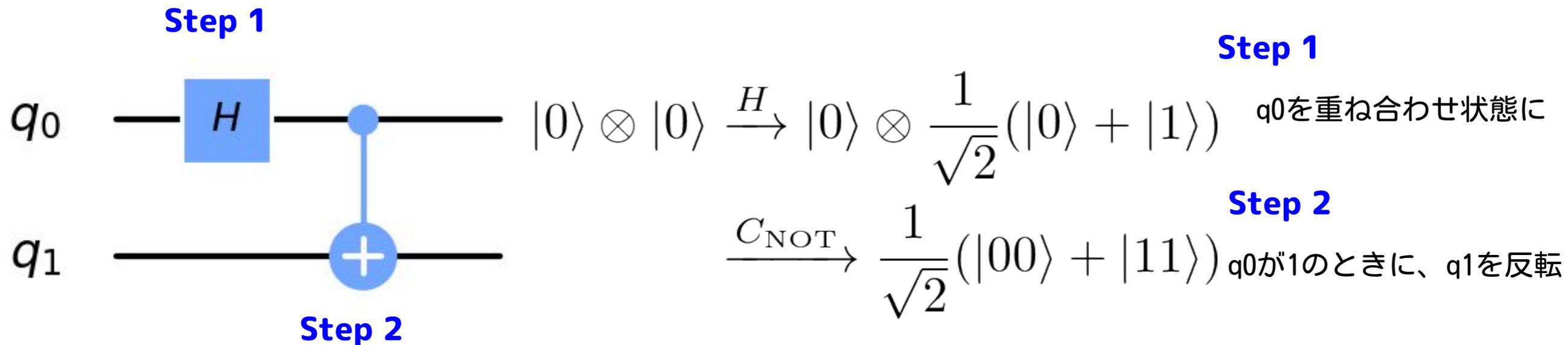
Qiskit による実践

```

from qiskit import QuantumCircuit

qc = QuantumCircuit(2) # 2 qubit の回路を作成
qc.h(0) # qubit 0 に H-gate
qc.cx(0, 1) # qubit 0 を control, qubit 1 を target に CNOT
qc.draw('mpl') # 作図

```



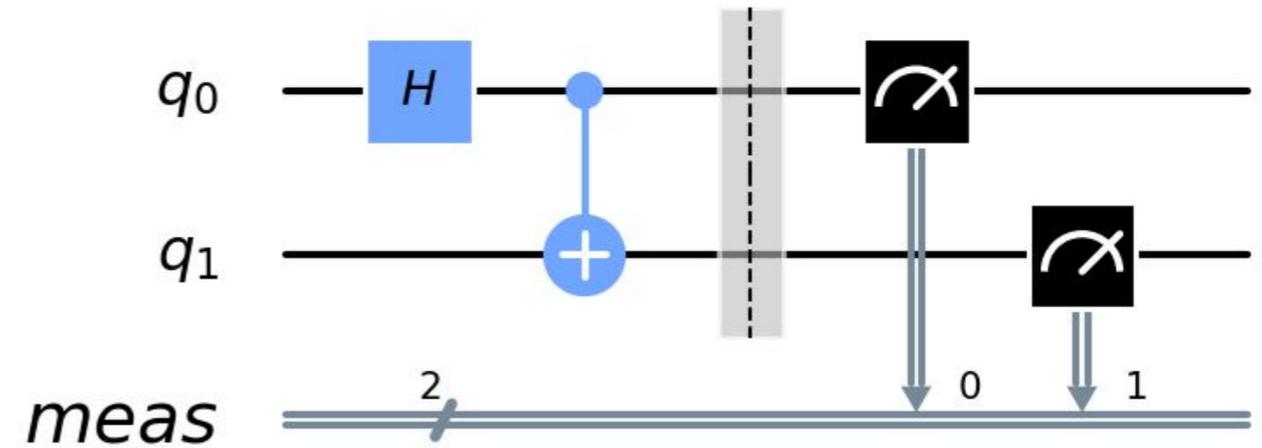
量子エンタングルメントを測ってみる

```

from qiskit import QuantumCircuit

qc = QuantumCircuit(2)
qc.h(0)
qc.cx(0, 1)
qc.measure_all() # 測定
qc.draw('mpl')

```



$$\frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$$

qubit 0 と 1 の状態は必ず一致
00 or 11 の確率は50%

量子回路をシミュレーターで実行

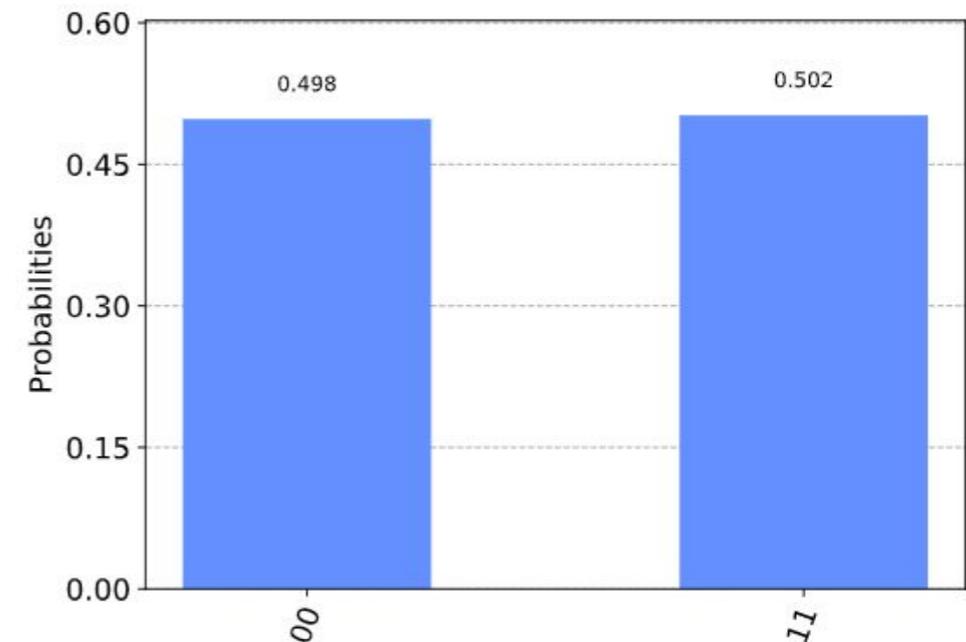
```

from qiskit import BasicAer, execute
from qiskit.visualization import plot_histogram

backend = BasicAer.get_backend('qasm_simulator')
result = execute(qc, backend=backend).result()
counts = result.get_counts()
plot_histogram(counts)

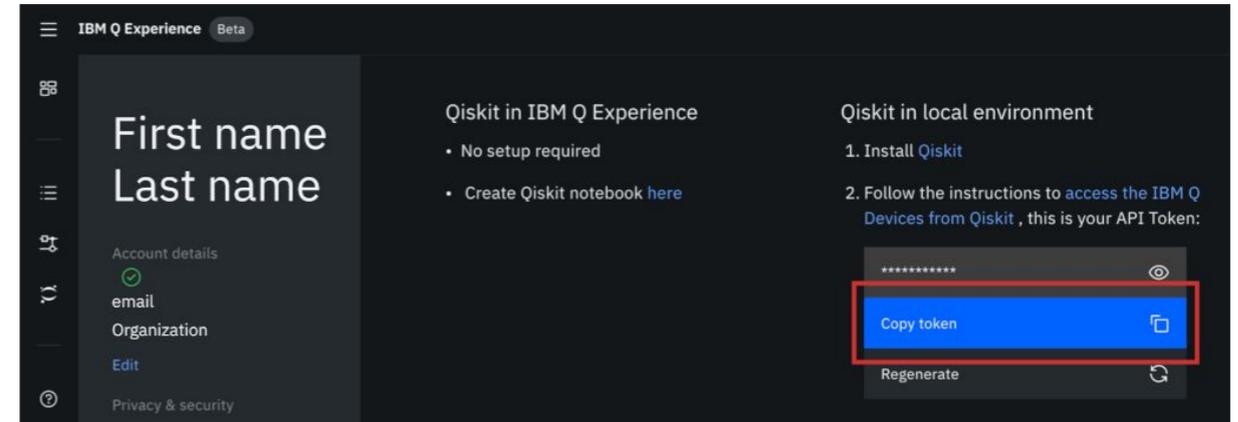
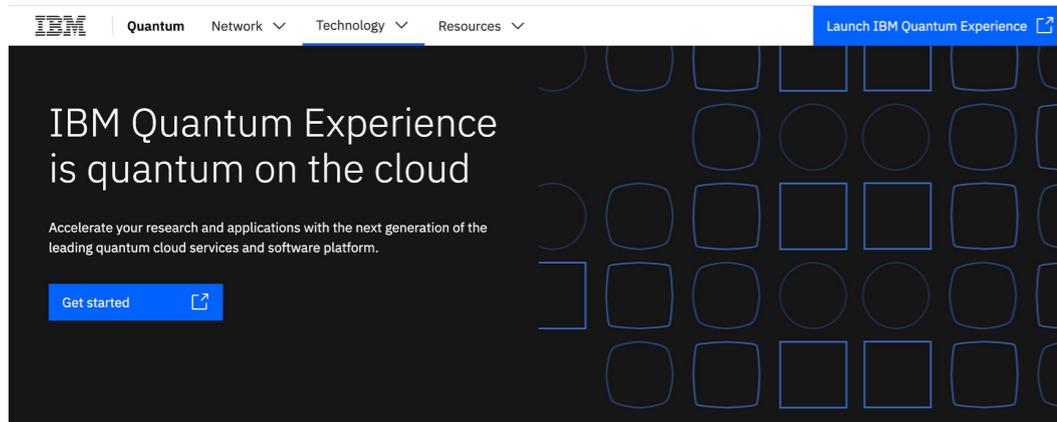
```

00 or 11 の状態のみ
約50%ずつ観測された



量子エンタングルメントを測ってみる・IBM Qでの実践

IBM Quantum Experience のアカウントを作成し、Token を取得



```

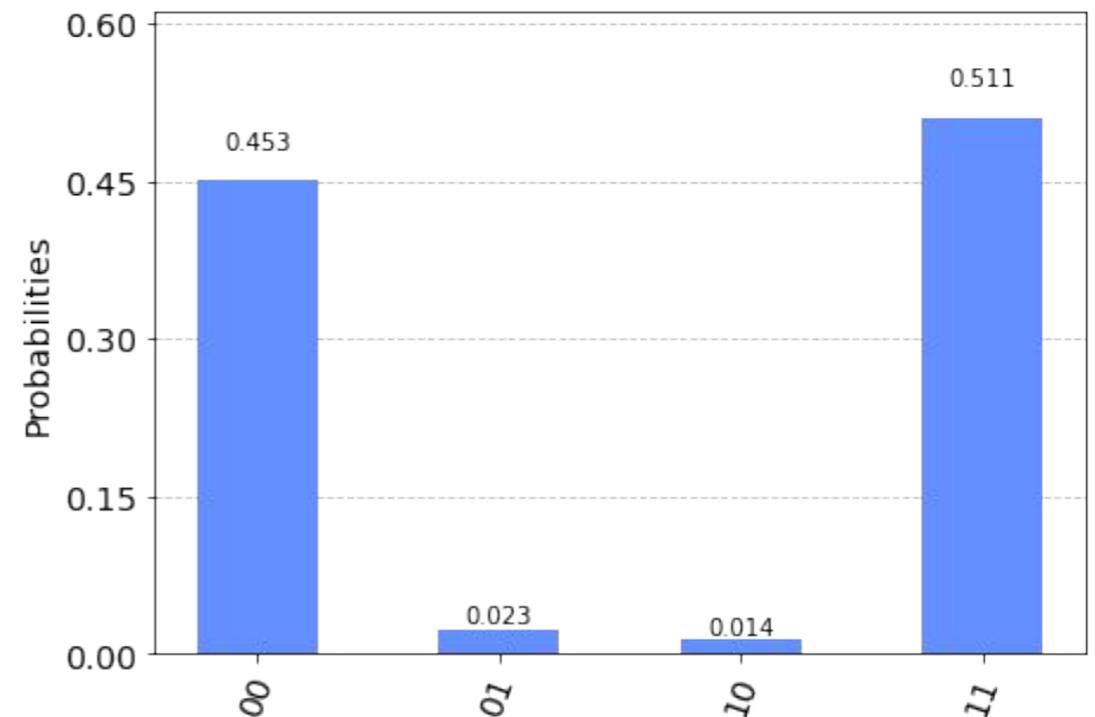
from qiskit import IBMQ
from qiskit.providers.ibmq import least_busy

# アカウントを認証してbackendを修正するだけ
IBMQ.save_account(my_token)
provider = IBMQ.load_account()
backend = least_busy(provider.backends(
    filters=lambda b: b.configuration().n_qubits >= 3
    and not b.configuration().simulator
    and b.status().operational==True))

# 実際の量子コンピュータで実行
job_exp = execute(qc, backend=backend, shots=8192)
exp_result = job_exp.result()
exp_measurement_result = exp_result.get_counts(qc)

# 結果のプロット
plot_histogram(exp_measurement_result)
    
```

実機のノイズのため
00 or 11 以外の状態も観測された



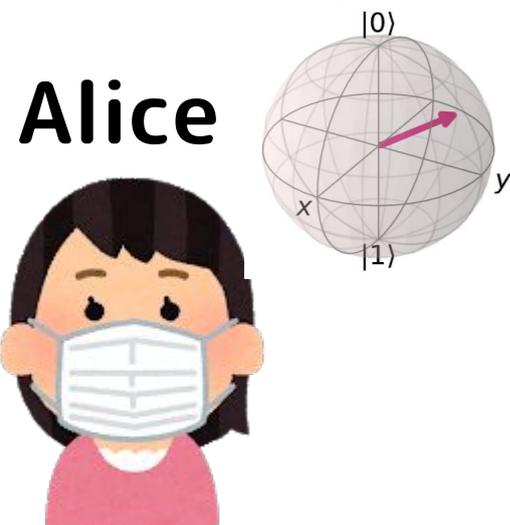
実践 1

量子アルゴリズムを実装・量子テレポーテーション

量子テレポーテーションとは？

→ エンタングルメントを利用した**量子情報転送アルゴリズム**

Alice の qubit を Bob へ**量子状態を壊さず**に**安全**に届けたい



3つのqubit (q_0, q_1, q_2) が必要

q_0 : Alice の送りたい状態

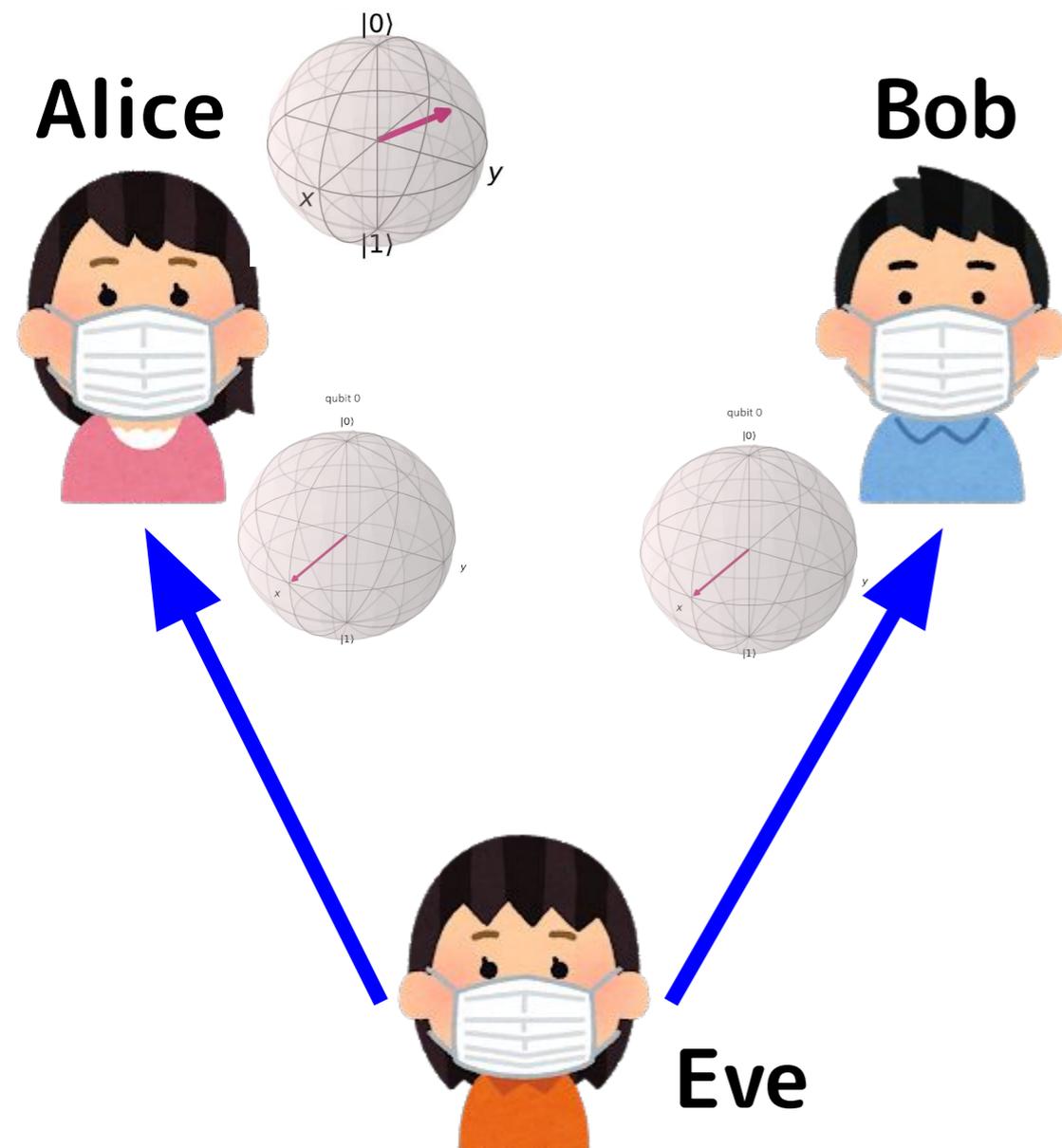
q_1, q_2 : Eve が準備する転送用のqubit



量子テレポーテーションとは？

→ エンタングルメントを利用した**量子情報転送アルゴリズム**

Alice の qubit を Bob へ**量子状態を壊さずに安全に届けたい**



3つのqubit (q_0, q_1, q_2) が必要

1. Eve : entanglement pair をAlice(q_1)とBob(q_2)へ送信

量子テレポーテーションとは？

→ エンタングルメントを利用した**量子情報転送アルゴリズム**

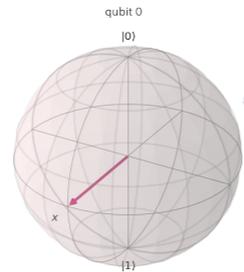
Alice の qubit を Bob へ**量子状態を壊さず**に**安全**に届けたい

Alice



測定結果
classical bit $\{0, 1\}^2$

Bob



3つのqubit (q_0, q_1, q_2) が必要

1. Eve : entanglement pair を Alice(q_1)と Bob(q_2)へ送信
2. Alice : 受信結果(q_1)と送りたい情報(q_0)をentangleさせ測定

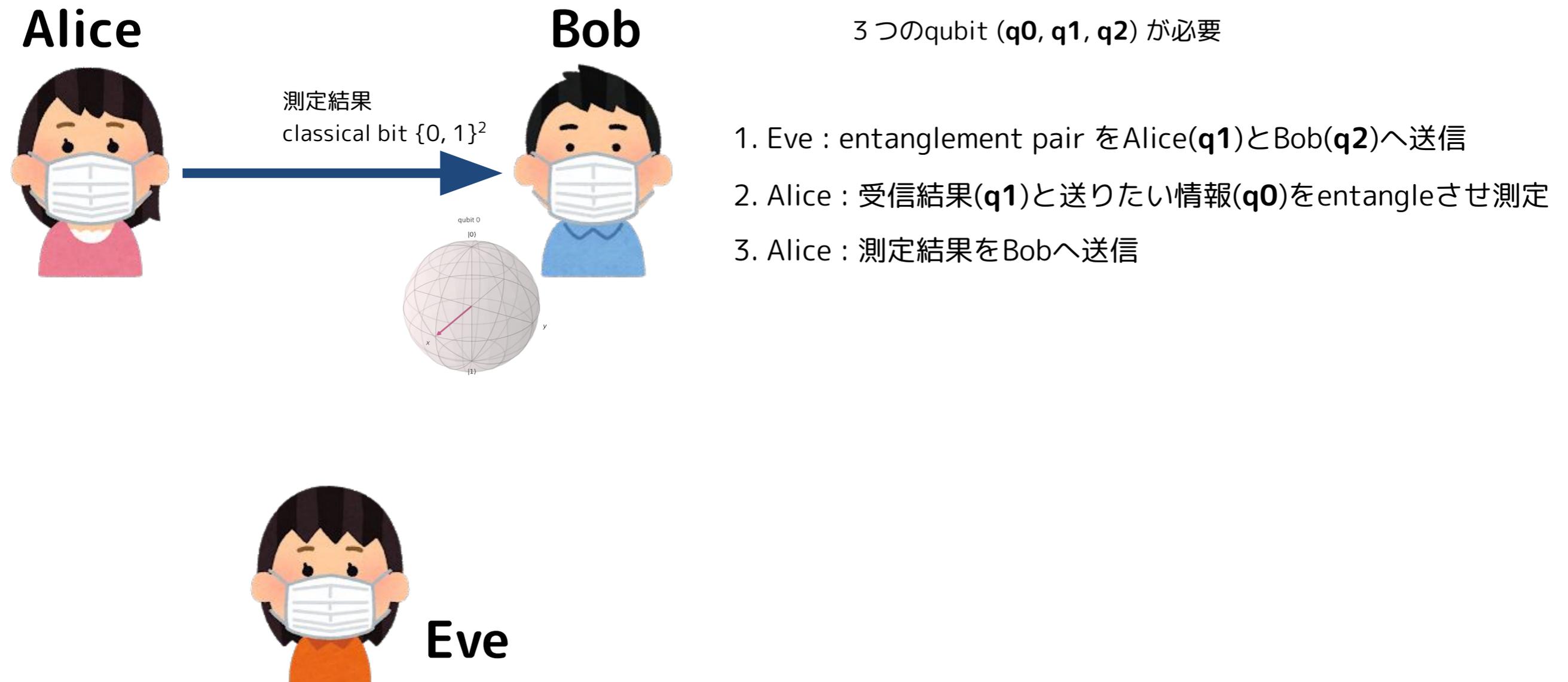


Eve

量子テレポーテーションとは？

→ エンタングルメントを利用した**量子情報転送アルゴリズム**

Alice の qubit を Bob へ**量子状態を壊さずに安全に届けたい**



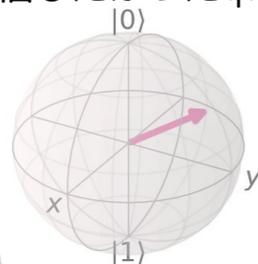
量子テレポーテーションとは？

→ エンタングルメントを利用した**量子情報転送アルゴリズム**

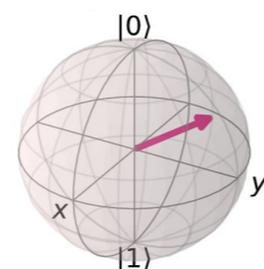
Alice の qubit を Bob へ**量子状態を壊さずに安全に届けたい**

送信したかったqubit

Alice



Bob



3つのqubit (q_0, q_1, q_2) が必要

1. Eve : entanglement pair を Alice(q_1)と Bob(q_2)へ送信
2. Alice : 受信結果(q_1)と送りたい情報(q_0)をentangleさせ測定
3. Alice : 測定結果を Bobへ送信
4. Bob : Alice の測定結果に従って、Eveからのqubit(q_2)を操作
5. Bobのqubit(q_2)はAliceが送信したかったqubit(q_0)と同じ状態

ポイント

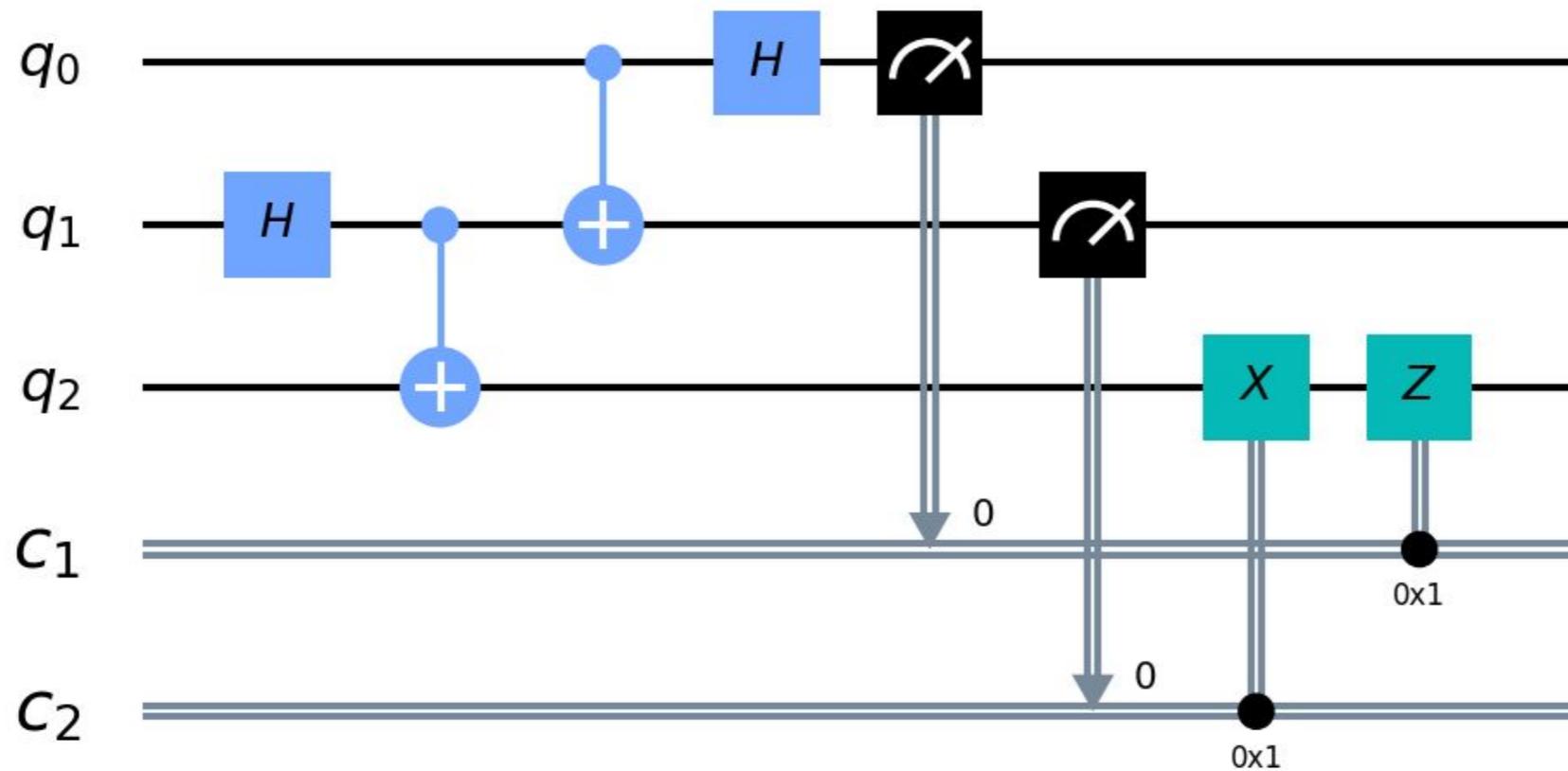
- ・ Aliceの qubitをBobには**送信**していないが、qubit の状態がエンタングルメントを通じて伝わる
- ・ **古典的な情報**をAliceからBobへ通信する必要がある
- ・ **量子暗号通信**としても利用できる



Eve

量子テレポーテーション回路

Alice (q0) の qubit を Bob (q2) の qubit へ転送



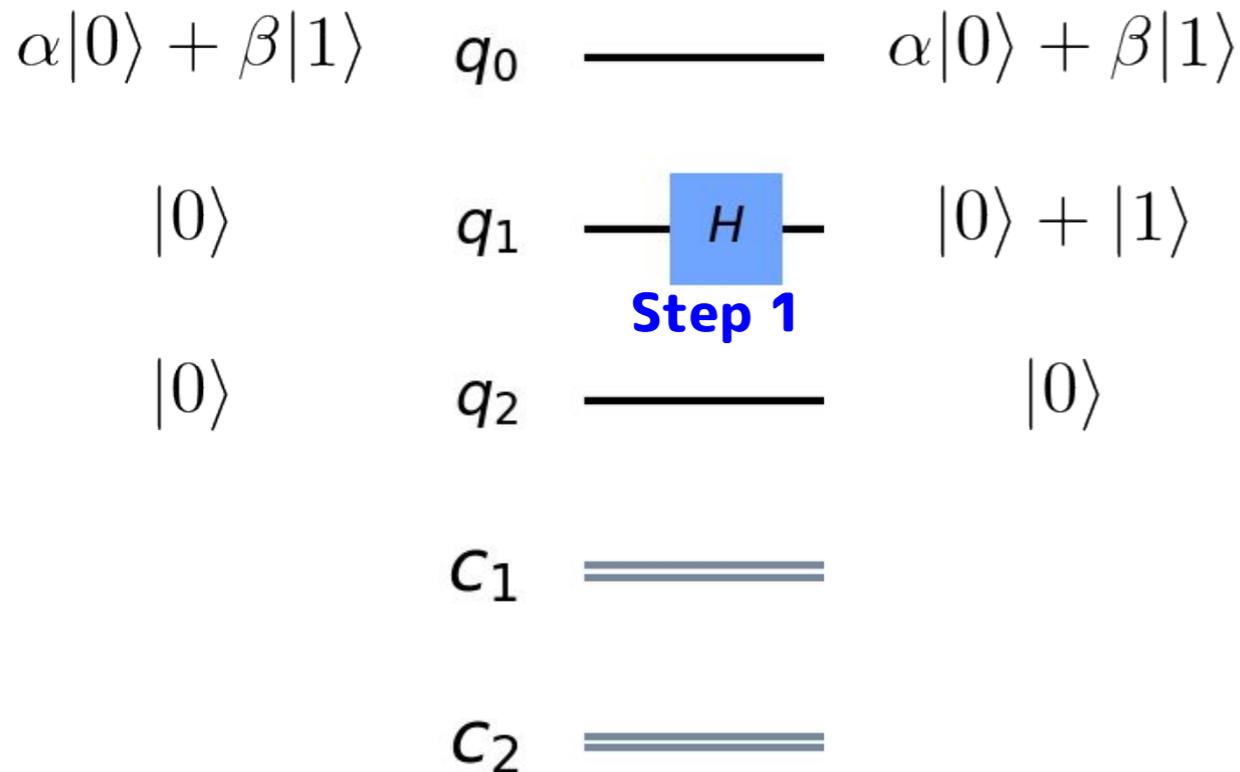
参考資料：Qiskit の Textbook

<https://qiskit.org/textbook/ch-algorithms/teleportation.html>

量子テレポーテーション回路 (Step 1)

$$(\alpha|0\rangle + \beta|1\rangle) \otimes |0\rangle \otimes |0\rangle$$

Step 1 \xrightarrow{H} $(\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle$



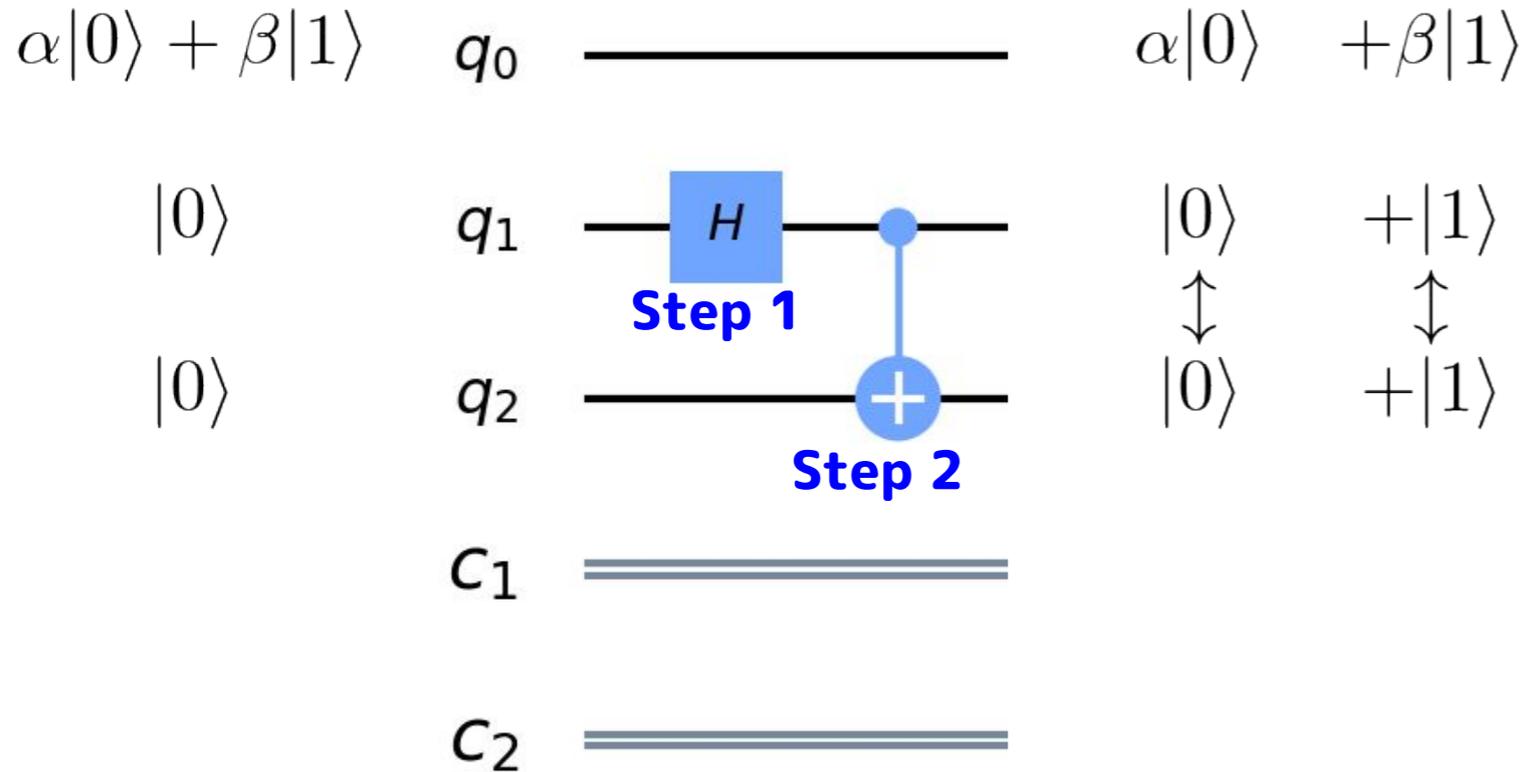
Eve : エンタングルメント・ペアを作成準備
q1が重ね合わせ状態に

量子テレポーテーション回路 (Step 2)

$$(\alpha|0\rangle + \beta|1\rangle) \otimes |0\rangle \otimes |0\rangle$$

Step 1 $\xrightarrow{H} (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle$

Step 2 $\xrightarrow{C_{NOT}} (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$



Eve : エンタングルメント・ペア (q1, q2)を作成し、送信

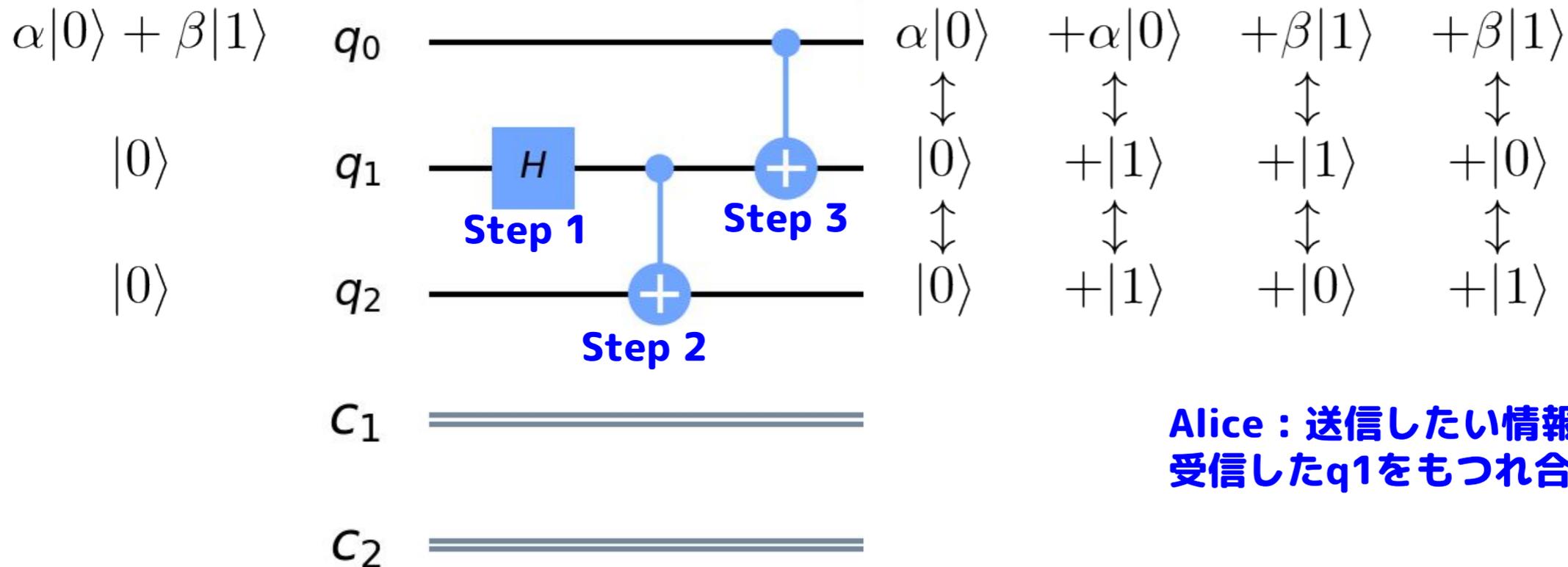
量子テレポーテーション回路 (Step 3)

$$(\alpha|0\rangle + \beta|1\rangle) \otimes |0\rangle \otimes |0\rangle$$

Step 1 $\xrightarrow{H} (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle$

Step 2 $\xrightarrow{C_{NOT}} (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

Step 3 $\xrightarrow{C_{NOT}} \frac{1}{\sqrt{2}}(\alpha|000\rangle + \alpha|011\rangle + \beta|110\rangle + \beta|101\rangle)$

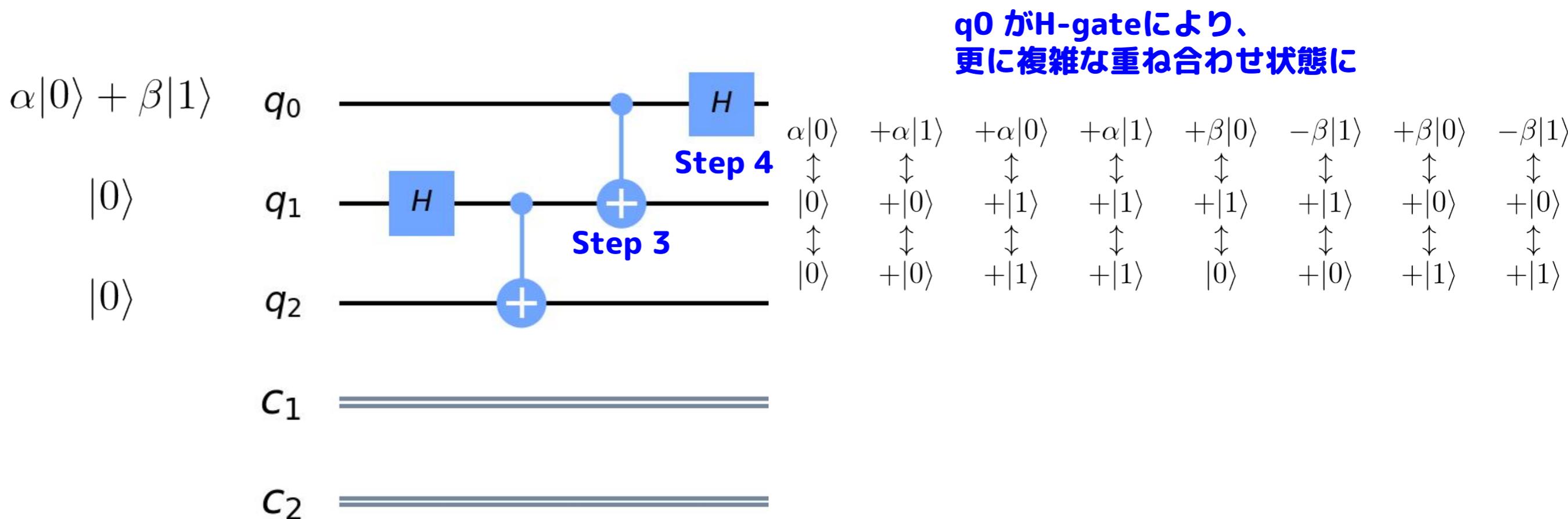


Alice : 送信したい情報q0と受信したq1をもつれ合わせる

量子テレポーテーション回路 (Step 4)

Step 3 $\xrightarrow{C_{\text{NOT}}} \frac{1}{\sqrt{2}}(\alpha|000\rangle + \alpha|011\rangle + \beta|110\rangle + \beta|101\rangle)$

Step 4 $\xrightarrow{H} \frac{1}{2} \{ \alpha(|000\rangle + |100\rangle + |011\rangle + |111\rangle) + \beta(|010\rangle - |110\rangle + |001\rangle - |101\rangle) \}$



量子テレポーテーション回路 (Step 5)

Step 4 $\xrightarrow{H} \frac{1}{2} \{ \alpha(|000\rangle + |100\rangle + |011\rangle + |111\rangle) + \beta(|010\rangle - |110\rangle + |001\rangle - |101\rangle) \}$

Step 5
q0, q1 : 測定

c0	c1	q2
0	0	$\alpha 0\rangle + \beta 1\rangle$
0	1	$\alpha 1\rangle + \beta 0\rangle$
1	0	$\alpha 0\rangle - \beta 1\rangle$
1	1	$\alpha 1\rangle - \beta 0\rangle$

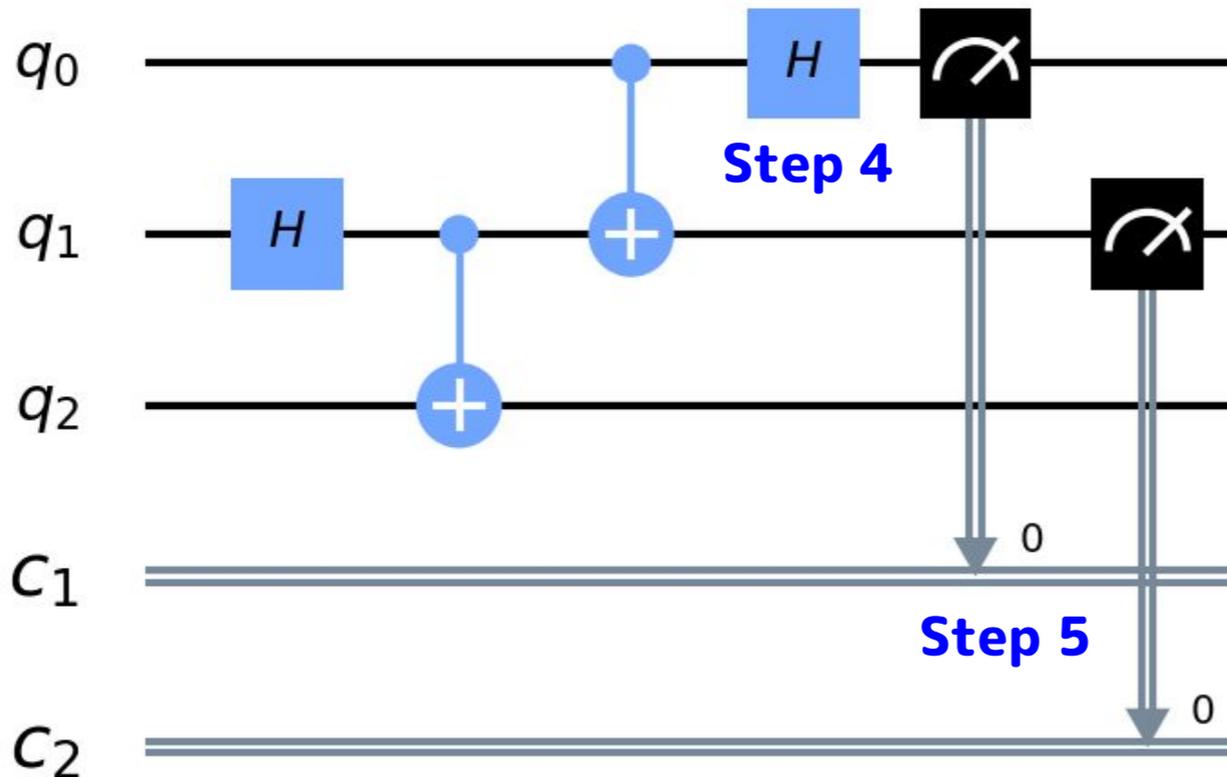
Alice : q0, q1 を測定

⇒ Bob :
q2 の状態は左の4種類のいずれかに確定
(0, 0) は送信したかった情報だが
他はちょっと違う

$\alpha|0\rangle + \beta|1\rangle$

$|0\rangle$

$|0\rangle$



q0, q1 の測定結果を元に補正したい
⇒条件付きゲートを利用

$$X|0\rangle = |1\rangle, X|1\rangle = |0\rangle$$

$$Z|0\rangle = |0\rangle, Z|1\rangle = -|1\rangle$$

q0, q1 の測定結果を元に補正したい
⇒条件付きゲートを利用

量子テレポーテーション回路 (Step 6)

$$X|0\rangle = |1\rangle, X|1\rangle = |0\rangle$$

$$Z|0\rangle = |0\rangle, Z|1\rangle = -|1\rangle$$

Step 5

q0, q1 : 測定

c0	c1	q2
0	0	$\alpha 0\rangle + \beta 1\rangle$ Step 6
0	1	$\alpha 1\rangle + \beta 0\rangle \xrightarrow{X} \alpha 0\rangle + \beta 1\rangle$
1	0	$\alpha 0\rangle - \beta 1\rangle \xrightarrow{Z} \alpha 0\rangle + \beta 1\rangle$
1	1	$\alpha 1\rangle - \beta 0\rangle \xrightarrow{ZX} \alpha 0\rangle + \beta 1\rangle$

Bob:

Alice の転送したかった

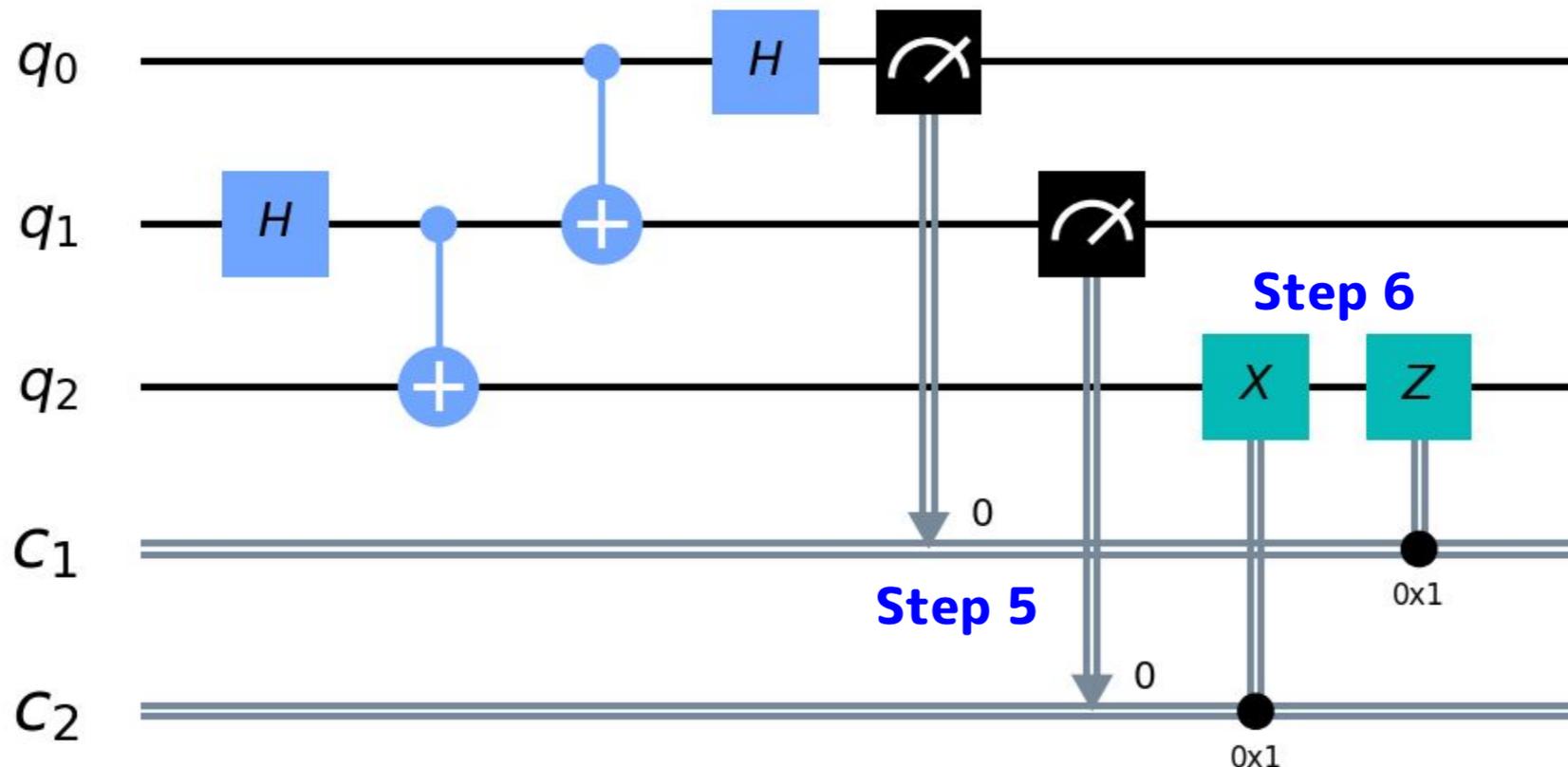
$$|q_2\rangle = \alpha|0\rangle + \beta|1\rangle$$

が受信できた!

$$\alpha|0\rangle + \beta|1\rangle$$

$$|0\rangle$$

$$|0\rangle$$



Step 6

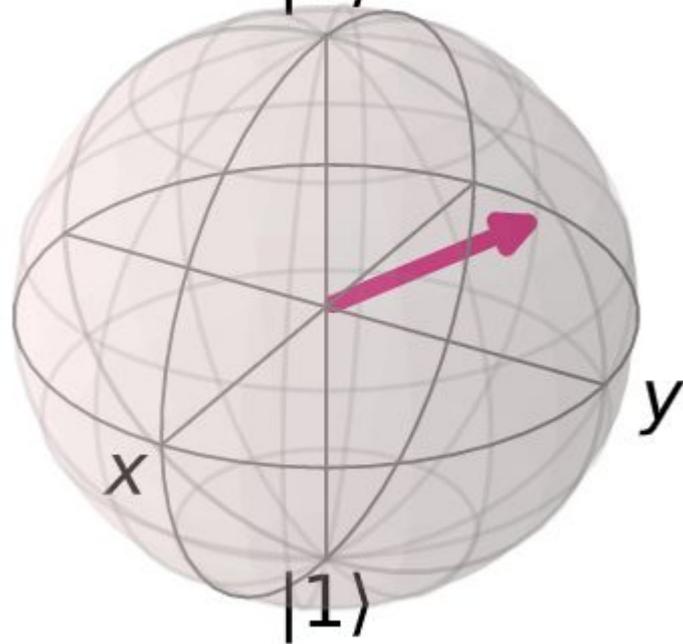
$$\alpha|0\rangle + \beta|1\rangle$$

量子テレポーテーション回路：実践

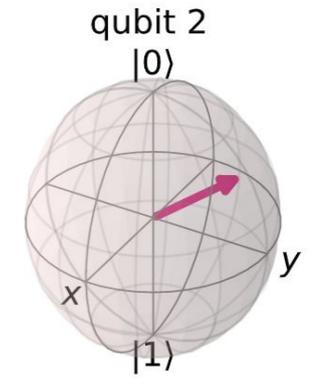
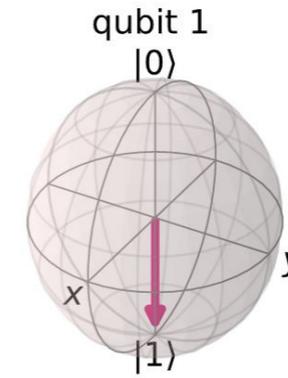
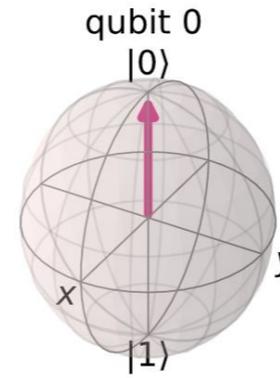
転送したい情報

qubit 0

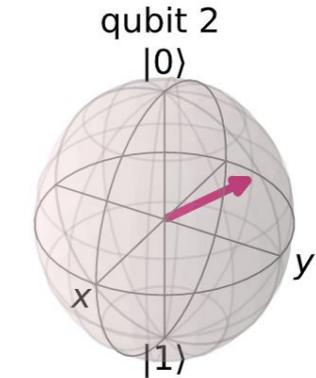
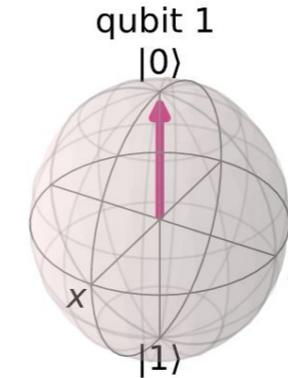
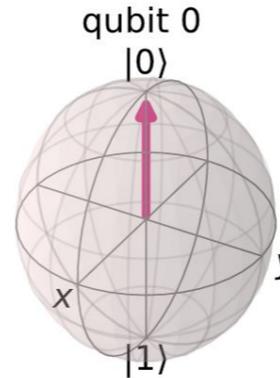
$|0\rangle$



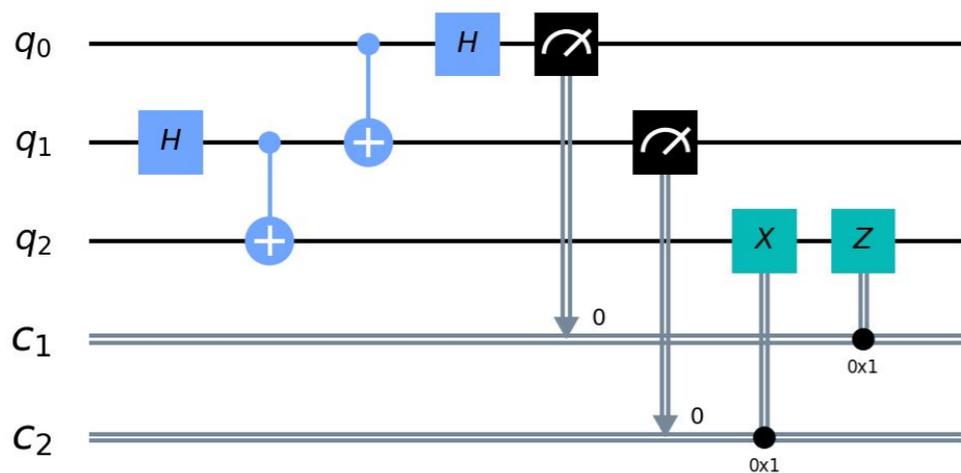
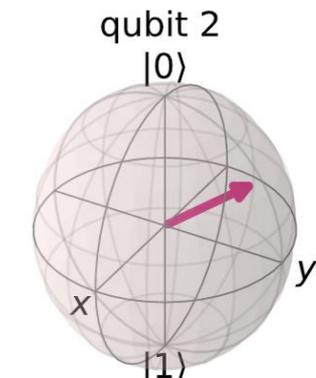
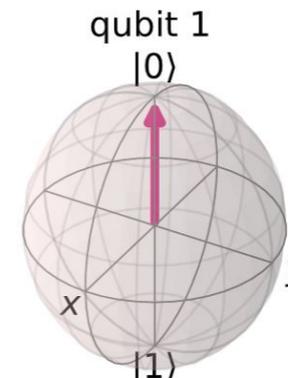
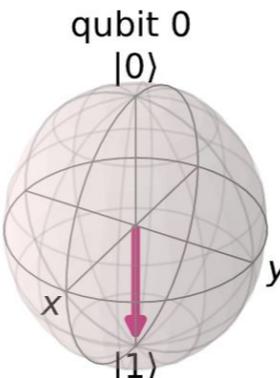
試行 1



試行 2



試行 3



Bob: q_0, q_1 の測定結果に関わらず
Alice の送信したかった状態が転送された

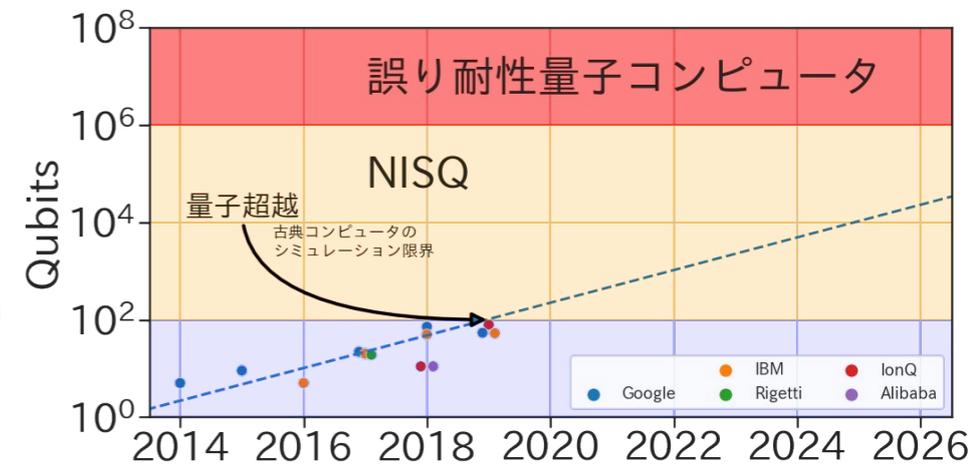
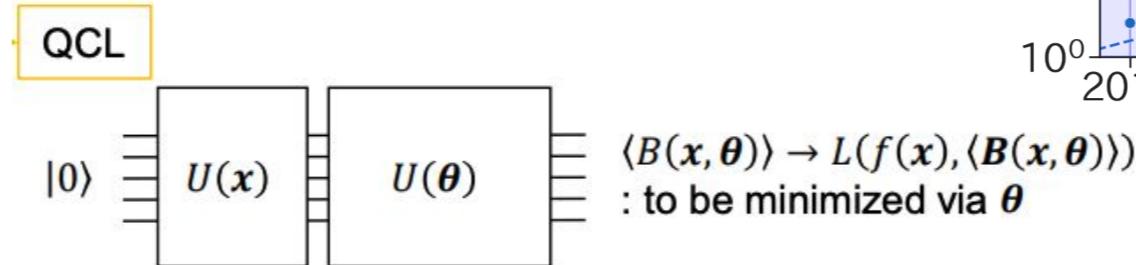
実践 2

量子機械学習・量子回路で分類問題を解く

量子機械学習

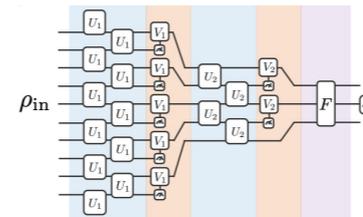
機械学習モデル + 量子回路(NISQ)

- 量子回路学習(QCL)



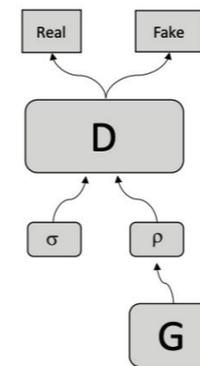
Quantum Circuit Learning
K. Mitarai,^{1,*} M. Negoro,^{1,2} M. Kitagawa,^{1,3} and K. Fujii^{4,2,†}
arXiv:1803.00745

- 量子畳み込みニューラル・ネットワーク



Article | Published: 26 August 2019
Quantum convolutional neural networks
Iris Cong, Soonwon Choi & Mikhail D. Lukin
Nature Physics **15**, 1273–1278(2019) | Cite this article
10k Accesses | 9 Citations | 146 Altmetric | Metrics

- 量子敵対的生成ネットワーク(QuGAN)



Quantum generative adversarial learning
Seth Lloyd¹ and Christian Weedbrook²
arXiv:1804.09139

- 量子サポート・ベクター・マシーン

Quantum support vector machine for big data classification
Patrick Rebentrost,^{1,*} Masoud Mohseni,² and Seth Lloyd^{3,†}
arXiv:1307.0471

他にも多様な機械学習モデルの拡張が研究されている

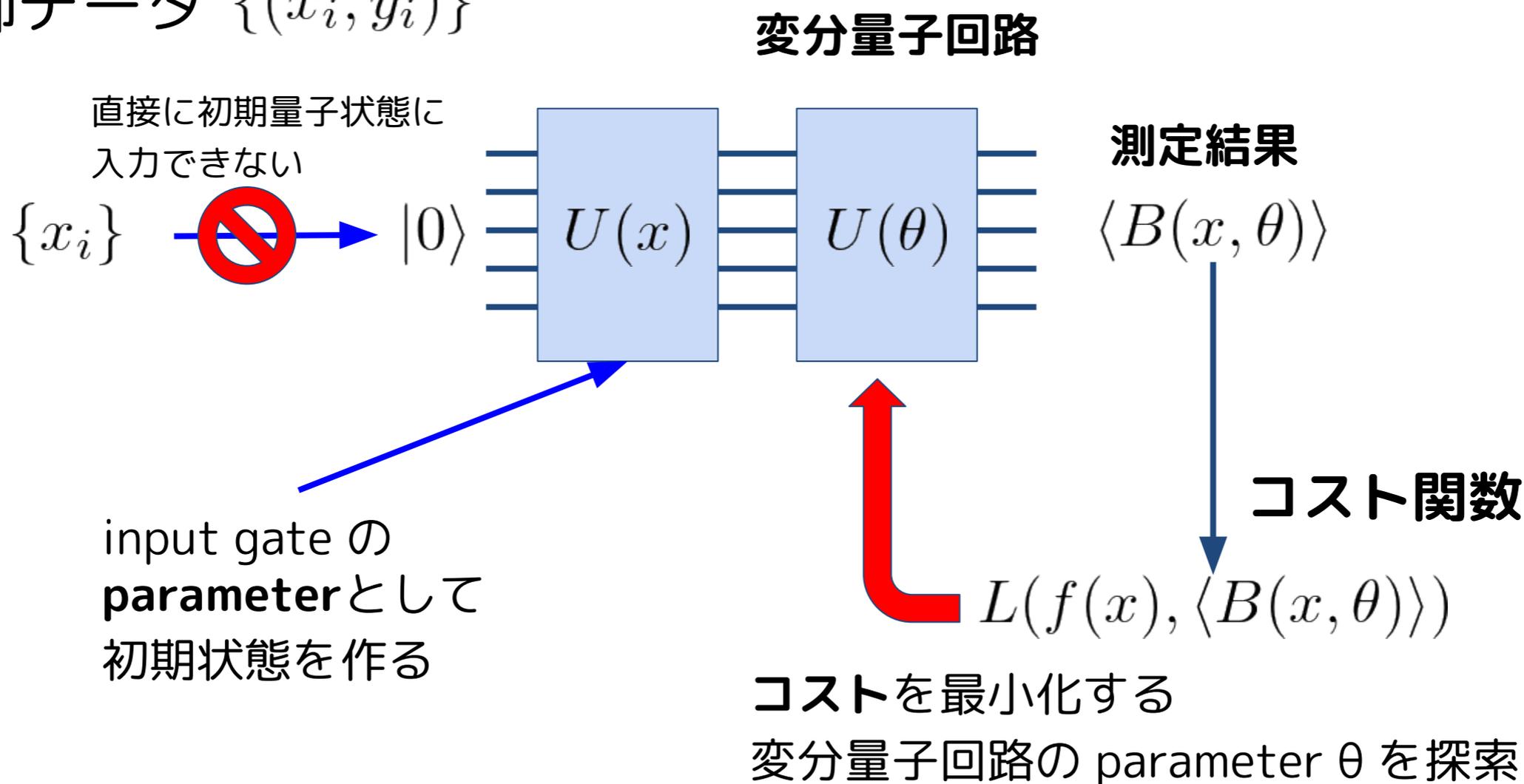
量子回路学習(QCL)

ニューラル・ネットワークを量子回路に置き換えた機械学習モデル

利点

- ・ 量子重ね合わせから指数関数的な状態を用いることができる
- ・ 量子回路のゲートの性質(unitary性)から overfitting が防げる(かも)

教師データ $\{(x_i, y_i)\}$

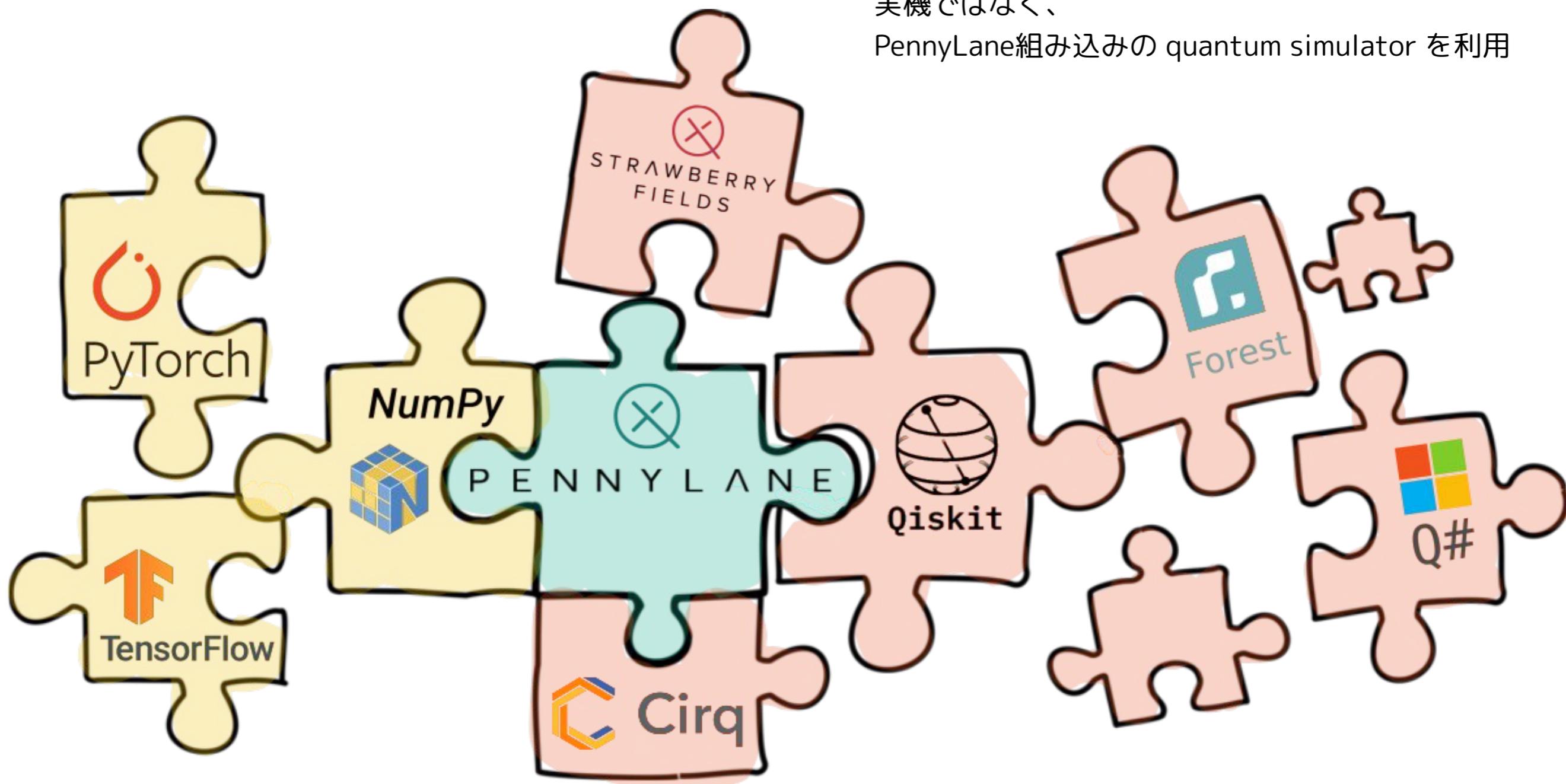


PennyLane

- ・今回は+ PyTorchで量子機械学習を試みる

実機ではなく、

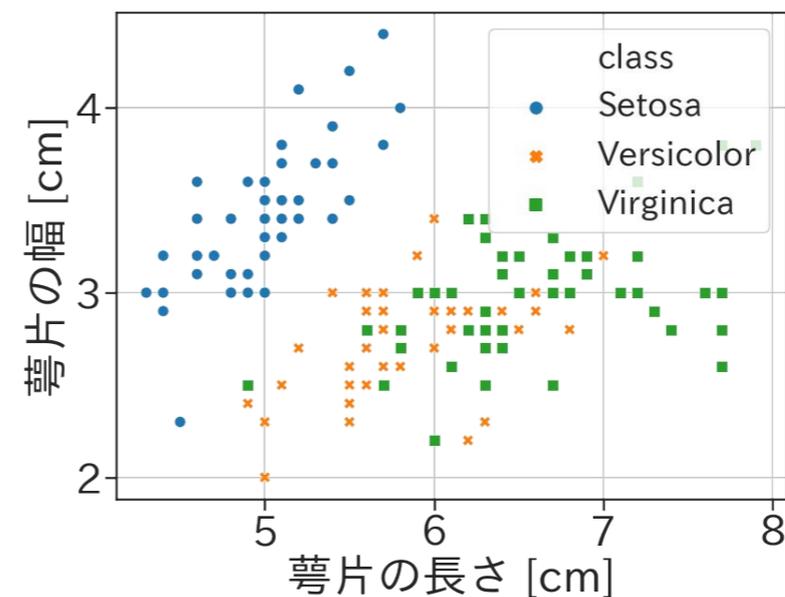
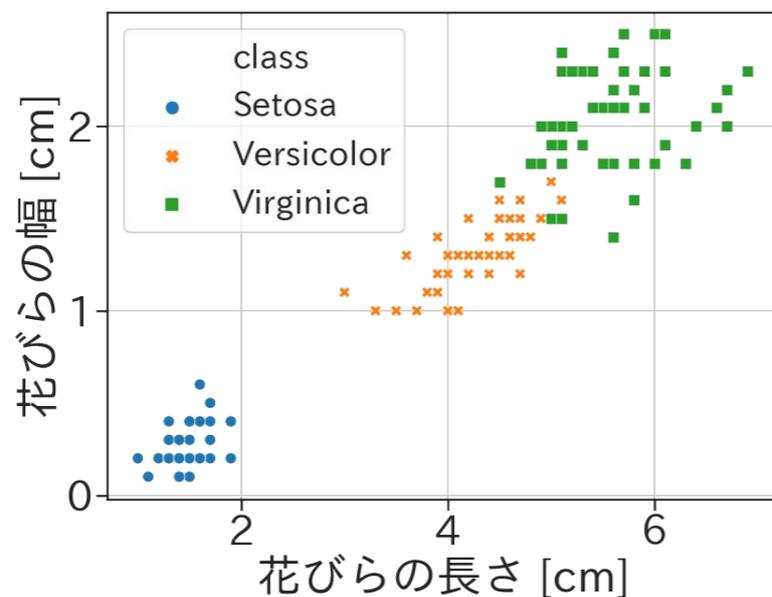
PennyLane組み込みの quantum simulator を利用



課題：量子回路でアヤメを分類する

Fisher's Iris data set : 機械学習で典型的な test data

- ・ 花びらと萼片(花冠の外側の部分)の長さ^{と幅} : 4 種類の特徴量

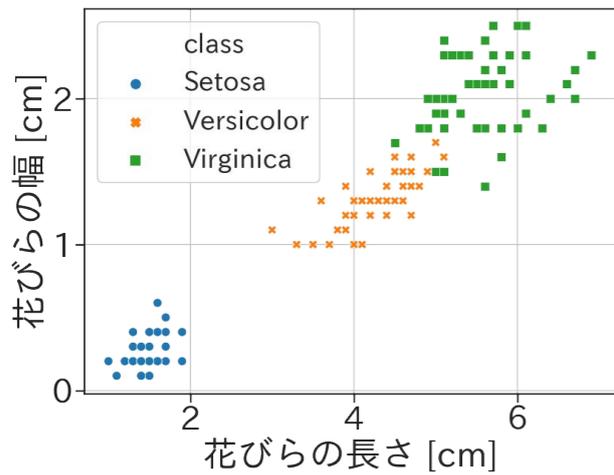
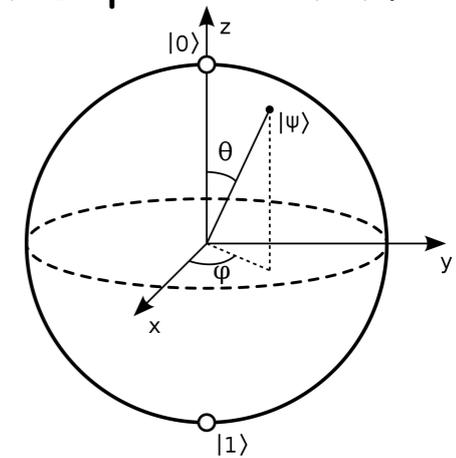


- ・ 3種類のカラス

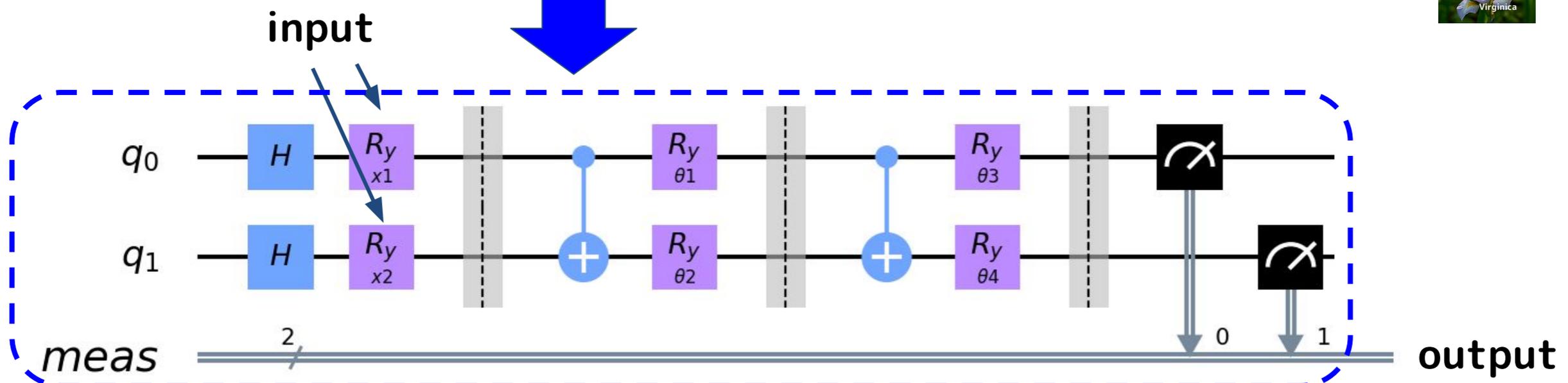
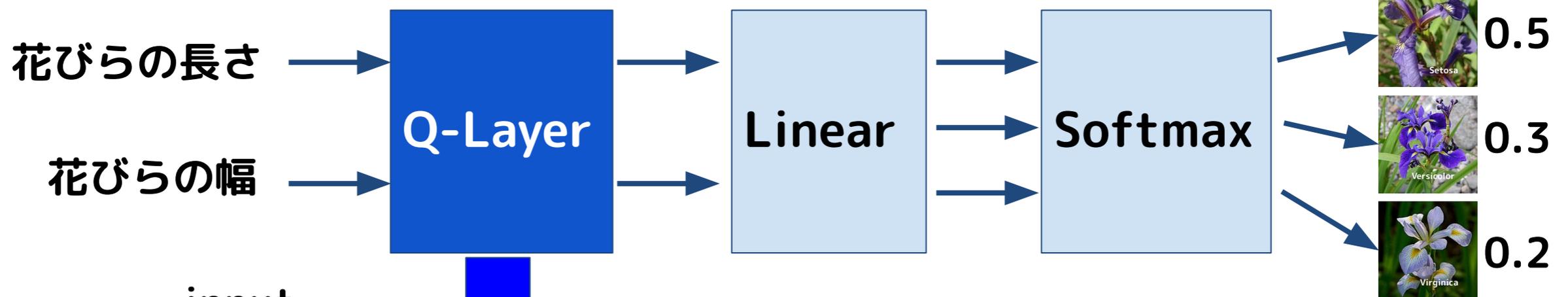


量子回路によるクラス分類モデルの設計

R_y
 θ_1 R_y -gate
Y軸でqubitを回転



- PennyLane + PyTorchで実装
 - 花びらの長さ・幅をinput
 - 2 qubit の Quantum Circuit
 - Cross-entropy loss を最小化



PennyLane + PyTorch によるモデル実装

```
@qml.qnode(dev, interface='torch')
def quantum_net(q_input_features, q_weights_flat):
    q_weights = q_weights_flat.reshape(q_depth, n_qubits)

    H_layer(n_qubits)

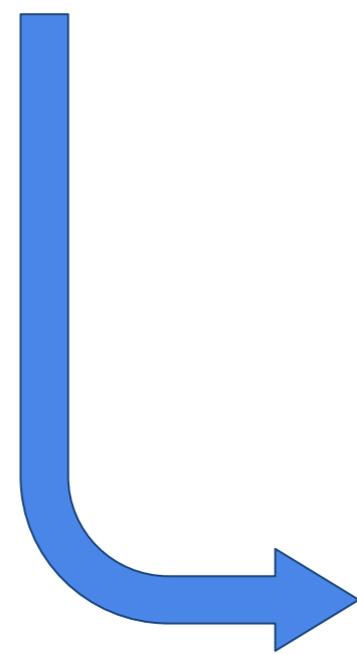
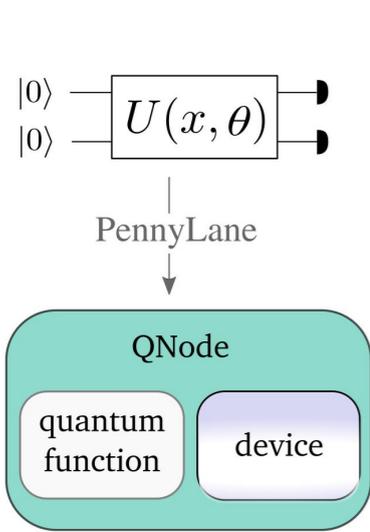
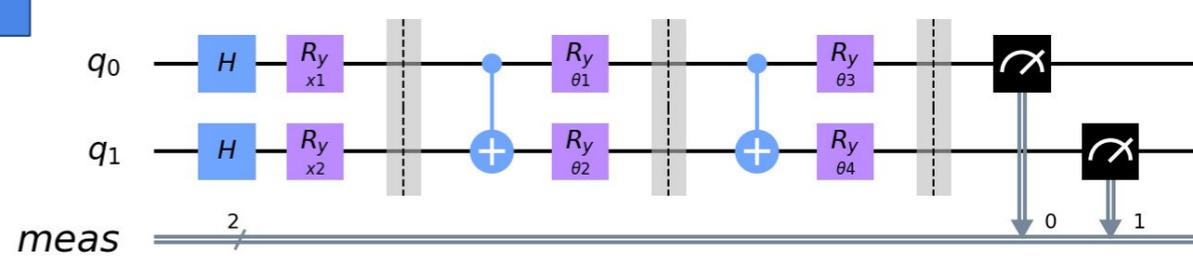
    RY_layer(q_input_features)

    for k in range(q_depth):
        entangling_layer(n_qubits)
        RY_layer(q_weights[k])

    exp_vals = [qml.expval(qml.PauliZ(position))
                 for position in range(n_qubits)]
    return tuple(exp_vals)
```



PennyLaneによる量子回路の作成



```
class QML_Model(nn.Module):
    def __init__(self, input_dim):
        super().__init__()
        self.q_params = nn.Parameter(q_delta * torch.randn(q_depth * n_qubits))
        self.post_net = nn.Linear(n_qubits, 3)

    def forward(self, input_features):
        q_in = input_features * np.pi / 2.0
        q_out = torch.Tensor(0, n_qubits)

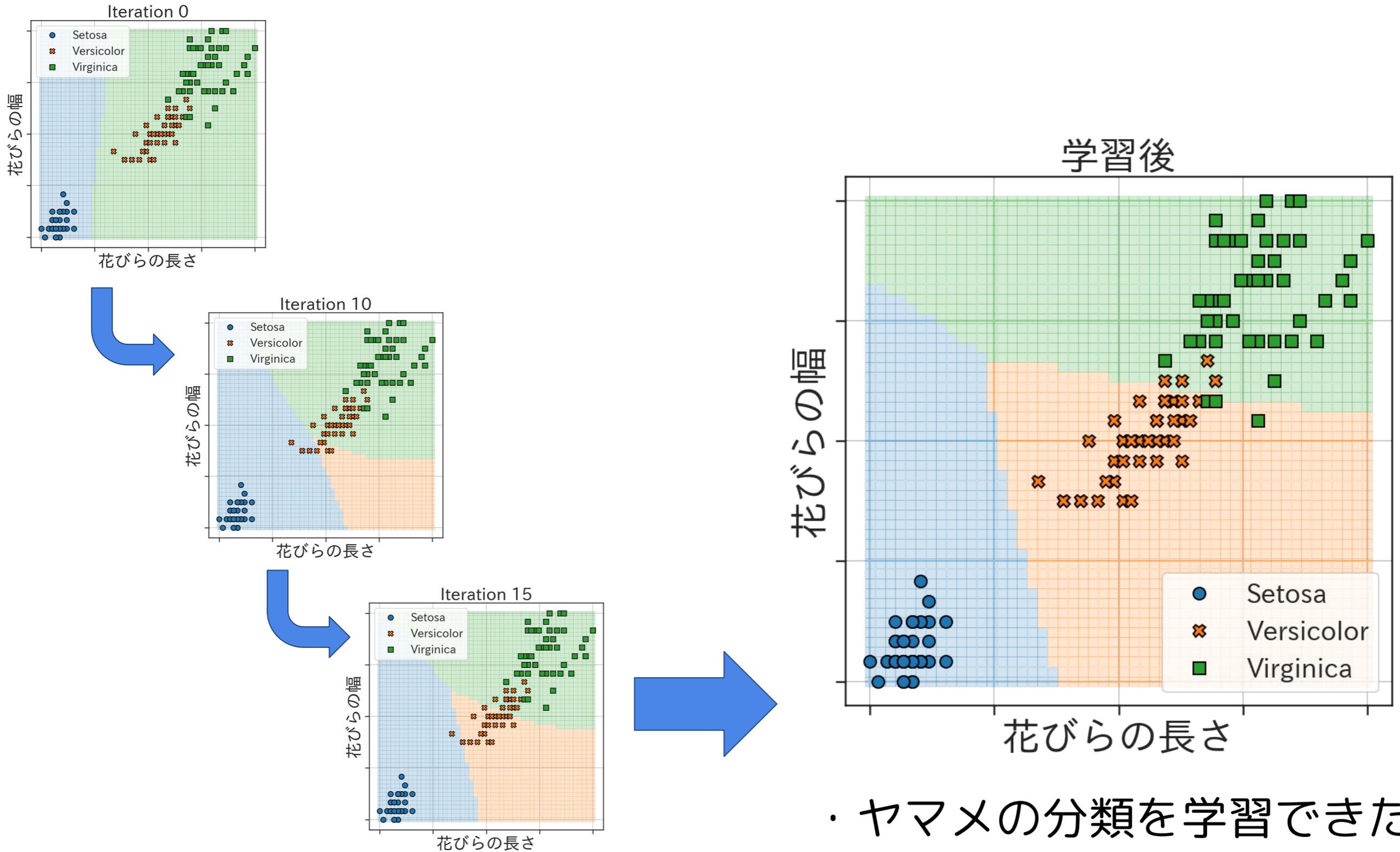
        for elem in q_in:
            q_out_elem = quantum_net(elem, self.q_params).float().unsqueeze(0)
            q_out = torch.cat((q_out, q_out_elem))

        post_out = self.post_net(q_out)
        x = nn.functional.softmax(post_out)

        return x
```

QNode object は
PyTorch の Neural Network に
組み込める

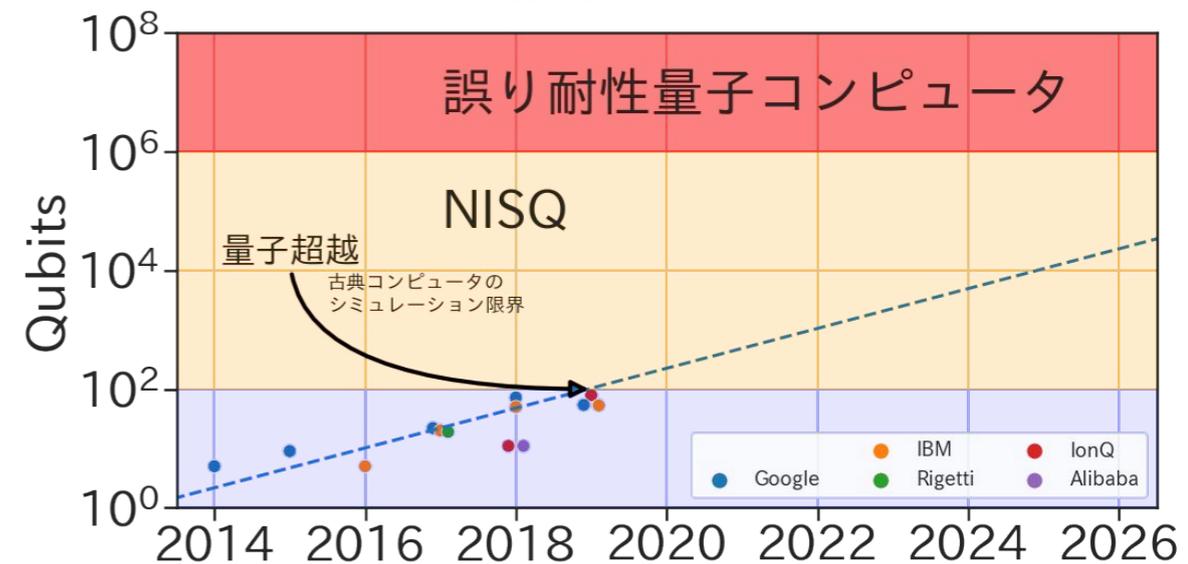
量子回路学習によるクラス分類・結果



・ヤマメの分類を学習できた

まとめ

- ・ 量子の“重ね合わせ”と“エンタングルメント”を利用すると、従来型コンピュータよりも飛躍的な性能がでる(かも)
- ・ ゲート式量子コンピュータが、限られたタスクではあるが、従来型のコンピュータを超えつつあり、更に性能改善が今後期待される
- ・ 汎用的なアルゴリズムはまだ無理であるが、NISQ で可能なアルゴリズム・問題を研究する意義がある
- ・ 機械学習への応用も、まだ研究が始まったばかり、これからの展開に注目



参考資料

- ・ Tutorials & Documents
 - ・ Quantum Native Dojo (日本語) <https://dojo.qiiaacs.org/ja/latest/index.html>
 - ・ IBM Quantum Experience <https://quantum-computing.ibm.com/>
 - ・ XANADU PennyLane <https://pennylane.ai/>
- ・ 解説
 - ・ 日本経済新聞「分かる教えたくなる量子コンピュータ」<https://vdata.nikkei.com/newsgraphics/quantum-computer-basic/>
 - ・ 藤井啓祐「驚異の量子コンピュータ」
- ・ 更に詳しく勉強したいなら
 - ・ 森川智之「量子計算理論」
 - ・ 佐川弘幸・吉田宣章「量子情報理論」
 - ・ 富田章久「量子情報工学」
- ・ 量子アニーリングについても知りたいなら
 - ・ 西村秀稔・大関真之「量子アニーリングの基礎」