日本語訳『Qiskit Textbook』勉強会 第0章

Kaori Namba
Senior Software Engineer

Quantum Tokyo

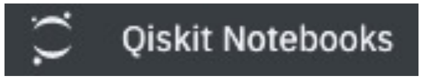
はじめに

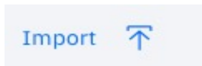
環境設定

ローカル環境の構築

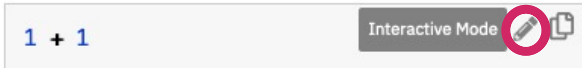
1. Qiskit Textbookのソースをダウンロード
   https://github.com/Qiskit/qiskit-textbook/tree/master-ja
2. Qiskitのインストール
   https://qiskit.org/documentation/install.html
3. Jupyter notebookを立ち上げて、Textbookのファイルを開く

IBM Quantum Experienceを利用

1. IBM Quantum Experienceにサインアップ&ログイン
   https://quantum-computing.ibm.com/
2. Dashboardで、Qiskit Notebooksメニューを選択
   

3. Importボタンをクリックし、Qiskit Textbookのファイルをアップロード
   

お手軽

1. Qiskit Textbookのサイトに行く
   https://qiskit.org/textbook/preface.html
2. Interactive modeに切り替える
   

0.1
Python 及び Jupyter notebook 入門

Python

- コンパイル不要なプログラミング言語
- プログラムを行単位で実行可（インタラクティブ・シェルまたはJupyter notebookを使用）
- プログラミング初心者向き
- 本教科書で使用するバージョンは Python 3 (2020年4月現在の最新バージョン)

Jupyter notebook

- Pythonでのコーディング方法の一つ
- プログラミング、文章、画像を統合
- 全てがセルの中に配置
  - テキスト・セル
  - コード・セル
- コード・セルの実行
  - セル内をクリックし、 Shift + Enter
  - Runボタンを押す

Python 及び Jupyter notebook 入門

Pythonは、コンパイル不要なプログラミング言語です。 プログラムを行単位で実行することができます（これは、Notebookを使用する方法です）。ですので、もしプログラミングについて全く知らないのであれば、Pythonはスタート地点として素晴らしい場所になります。現在のバージョンは Python 3であり、本教科書で使用するものです。

Pythonでコーディングする方法の一つは、Jupyter notebookを使用することです。 これはおそらく、プログラミング、文章、および画像を統合する最良の方法です。 Notebookでは、全てがセルにの中に配置されます。 テキスト・セルとコード・セルは最も一般的なものです。 Jupyter notebookとしてこのセクションを表示している場合、現在読んでいるこのテキストはテキスト・セルに配置されています。 コード・セルは、以下にあります。

コード・セルの内容を実行するには、そのセルをクリックし、 Shift + Enter を押します。または、左側に小さな矢印がある場合は、それをクリックすることもできます。

```
In [1]: 1 + 1
Out[1]: 2
```

Pythonの主な文法

Quantum Tokyo

データ型

- integer
- float
- bool
- str
- None
- list *
- tuple *
- dict

*... 添字は0から

ループ

- for

```
for j in range(5):
  print(j)
```

```
for j in a_list:
  print(j)
```

```
for key in a_dict:
  value = a_dict[key]
  print('key =',key)
  print('value =',value)
  print()
```

条件分岐

- if
- elif
- else

```
if ‘orange' in a_list:
   print('We have an orange!')
elif a_list[5]=='apple':
   print('We have an apple!')
else:
   print('Not much fruit here!')
```

パッケージ

- import
  - numpy
  - matplotlib
  - networkx

```
import numpy as np
np.sin(np.pi/2)
```

関数

- def

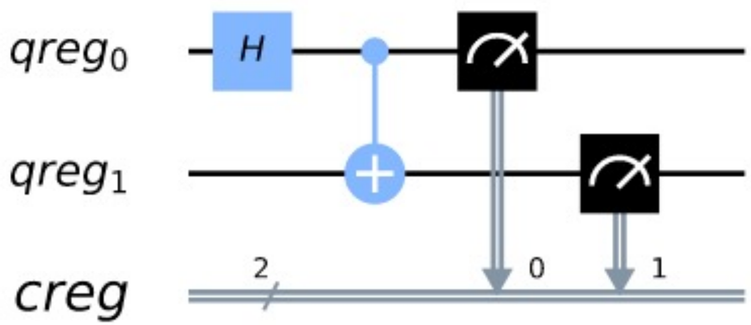
```
def do_add ( In1, In2 ):
  the_answer = In1 + In2
  return the_answer
```

0.2
Qiskit

Qiskitの基礎

量子回路の作成

- QuantumCircuitクラス
- QuantumRegisterクラス
- ClasssicalRegisterクラス



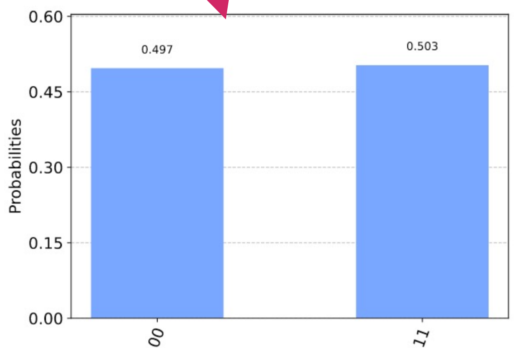

```
from qiskit import *
from qiskit.visualization import plot_histogram
from qiskit.tools.monitor import job_monitor

qc = QuantumCircuit()
qr = QuantumRegister(2,'qreg')
cr = ClassicalRegister(2,'creg')
qc.add_register( qr )
qc.add_register( cr )

qc.h(qr[0])
qc.cx(qr[0], qr[1]);
qc.measure(qr,cr)

qc.draw(output='mpl')
```

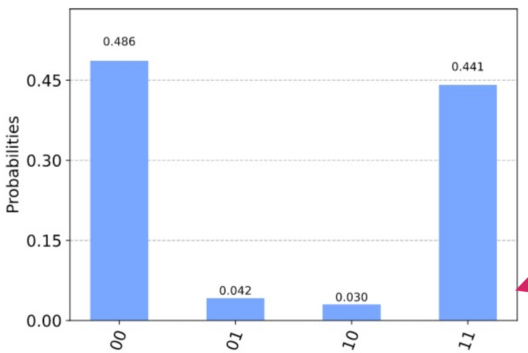
Qiskitの基礎 – cont.

シミュレーターでの実行

- QasmSimulatorクラス

```
emulator = Aer.get_backend('qasm_simulator')
job = execute( qc, emulator, shots=8192 )
hist = job.result().get_counts()
plot_histogram(hist)
```





実機での実行

- IBMQBackendクラス

```
IBMQ.load_account()
provider = IBMQ.get_provider(hub='ibm-q')
provider.backends()

real_device = provider.get_backend('ibmq_16_melbourne')
job = execute( qc, real_device, shots=1024 )
job_monitor (job)
```

```
hist = job.result().get_counts()
plot_histogram(hist)
```

0.3
線形代数

用語集のようなもの

集合Mの元x

$$x \in M$$

体

四則演算に対し閉じている集合

線形結合

ベクトルのスカラー倍の加算

$$\sum_{i=o}^{n} k_i \boldsymbol{v}_i$$

ヌルベクトル

本教科書では、$\boldsymbol{v} + \boldsymbol{0} = \boldsymbol{v}$となる$\boldsymbol{0}$

$\mathbb{R}^n$

n-次元実数空間
$\boldsymbol{x} = (x_1, x_2, \ldots x_n)$

複素共役

$$x \ \ = a + bi$$
$$x^* = a - bi$$

共役転置/随伴行列

$$A_{ij}^{\dagger} = \left(A_{ij}^{T}\right)^* = A_{ji}{}^*$$

ケット・ベクトル

$$|\psi\rangle = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix}$$

ブラ・ベクトル

$$\langle\psi| = |\psi\rangle^{\dagger} = \left(a_1^*, a_2^*, \cdots a_n^*\right)$$

単位行列

$$\mathbb{I} = \begin{pmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{pmatrix}$$

エルミート行列

$$H^{\dagger} = H$$

ユニタリー行列

$$U^{\dagger}U = UU^{\dagger} = \mathbb{I}$$

ヒルベルト空間

取り敢えずの計算に困らないベクトル空間 (by EMANの物理学・量子力学)

ベクトルと行列

ベクトル

- 方向と大きさを持つ量

- 正確には、ベクトル空間の元

ベクトル空間

- 体 F 上のベクトル空間 $\boldsymbol{V}$ があるとき、$\mathbf{a}, \mathbf{b} \in \boldsymbol{V}$, $\mathrm{n} \in \mathrm{F}$ に対し以下が成り立つ

$$\mathbf{a} + \mathbf{b} \in \boldsymbol{V}, \qquad \mathrm{n}\mathbf{a} \in \boldsymbol{V}$$

- 実習: 実数体$\mathbb{R}$上の$\mathbb{R}^2$がベクトル空間であることを証明せよ

行列

- 数や記号を矩形に並べた物

$$M = \begin{pmatrix} 1 & -2 & 3 \\ 1 & 5i & 0 \\ 1+i & 7 & -4 \end{pmatrix}$$

- ベクトルを変換する数学的オブジェクト

行列A, Bの乗算

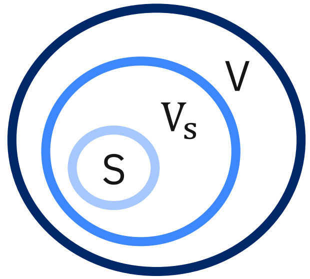
$$C = AB \qquad (\text{注意: } AB \neq BA)$$

$$c_{ij} = \sum_{k=1}^{m} a_{ik} b_{kj}$$

- 実習: $|0\rangle, |1\rangle$にPauli-Xゲートを作用させよ

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

重要な行列と部分空間



エルミート行列

- 共役転置が自分自身になる

$$H^\dagger = H$$

- 実習: Pauli-Y行列がエルミートであることを示せ

$$\sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

ユニタリー行列

- 共役転置が逆行列になる

$$U^\dagger U = UU^\dagger = \mathbb{I}$$

- 実習: Pauli-Y行列がユニタリーであることを示せ

部分空間

- 体F上の、ベクトル空間Vの部分集合Sの線形結合で作る集合V_sがVに含まれている場合、V_s はSが作る部分空間という。このとき、Sをスパン集合、V_sはSが張る空間という

- V_s内の任意のベクトル$|v\rangle$は、スパン集合Sの線形結合で記述できる

$$|v\rangle = \sum_i f_i |v_i\rangle$$

$$where |v_i\rangle \in S, f_i \in F$$

線型独立と基底

$$\sum_i f_i|v_i\rangle = \mathbf{0}$$
$$where|v_i\rangle \in S, f_i \in F$$

線形従属

– 式が成立する少なくとも1つのゼロでないf_iが存在する

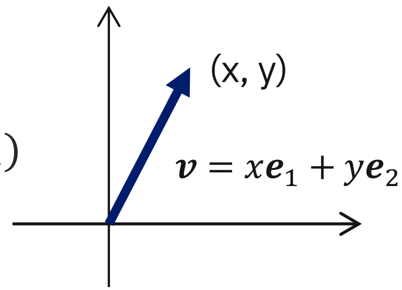
– すなわち、ある$|v_k\rangle$は、他の$|v_i\rangle$の線形結合で表現できる

線形独立

– 全てのf_iがゼロでないと式が成立しない

基底

- 線型独立なスパン集合
- ベクトル空間を表現できる最小のベクトル集合
  - 例: 2次元平面の場合
    - $\boldsymbol{e}_1 = (1, \quad 0), \boldsymbol{e}_2 = (0, \quad 1)$
  - 例: 1量子ビットの場合
    - $|0\rangle = \binom{1}{0}, |1\rangle = \binom{0}{1}$
    - $|+\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}, |-\rangle = \frac{|0\rangle-|1\rangle}{\sqrt{2}}$



量子状態空間

- 複素数体ℂ上のヒルベルト空間
  ヒルベルト空間とはベクトル空間で、内積が定義でき、かつその内積空間が完備性を持つもの
- 量子状態ベクトル
  量子状態空間の元
- 量子状態ベクトル$|a\rangle$と$|b\rangle$の内積

$$\langle a|b\rangle = \begin{pmatrix} a_1^* & a_2^* & \dots & a_n^* \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} = \sum_{i=1}^{n} a_i^* b_i$$

- 正規化条件

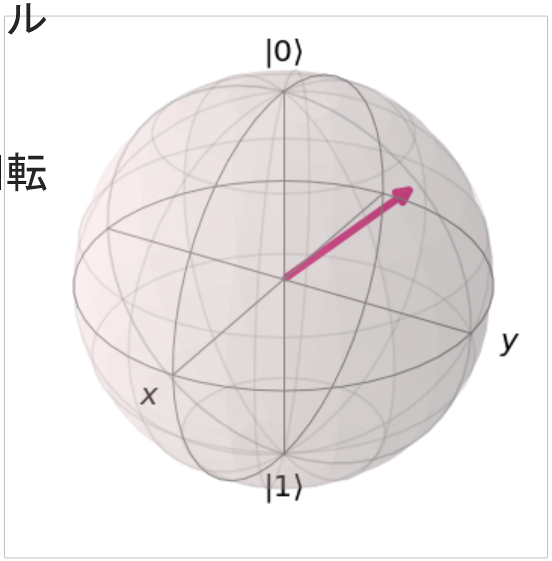
$$\langle\psi|\psi\rangle = 1$$

  - ベクトルの長さの2乗が1
  - 特定方向におけるベクトルの長さは、その特定状態で観測される確率振幅
  - $\langle a|b\rangle$ は、$|a\rangle$が $|b\rangle$にどれだけ沿っているかを示す量 -> 測定の確率
- ユニタリー変換に対し閉じている
  演習：ユニタリー変換後も正規化条件が真であることを示せ

ブロッホ球と固有ベクトル/固有値

ブロッホ球

- 1量子ビットを表現するモデル
  - 球面：1量子状態空間
  - 矢印：量子状態ベクトル
  - ユニタリー変換：
量子状態ベクトルの回転



行列Aに対する固有ベクトル$|v\rangle$と固有値λ

$$A|v\rangle = \lambda|v\rangle$$

固有ベクトル/固有値が存在するための条件

$$\det(A - \lambda \mathbb{I}) = 0$$

固有多項式 (Aが2x2行列の場合)

$$\lambda^2 - (A_{11} + A_{22})\lambda + A_{11}A_{22} - A_{12}A_{21} = 0$$

演習: Pauli-Z行列の固有ベクトル、固有値を求めよ

$$\lambda = 1 \text{のとき} \quad |v\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$$

$$\lambda = -1 \text{のとき} |v\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$$

Z基底で測定する≡Z行列の固有ベクトルに落とし込む

行列指数関数

演習：ある変換U

$$\mathrm{U} = e^{i\gamma H}$$
$$where\ \gamma \in \mathbb{R}\ and\ H\ is\ Hermite$$

がユニタリーであることを証明せよ

演習：Uをマクローリン展開せよ

$$U = e^{i\gamma H} = \sum_{n}^{\infty} \frac{(i\gamma H)^n}{n!} = \cos\gamma \cdot \mathbb{I} + i\sin\gamma \cdot H$$

Pauli行列はユニタリかつエルミートかつ対合

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \sigma_y = \begin{pmatrix} 0 & i \\ -i & 0 \end{pmatrix}, \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$\Rightarrow e^{i\gamma\sigma_k}\ (k \in \{x, y, z\})$がsin-cosで書ける！

演習：行列Aの固有ベクトル$|v\rangle$と固有値λについて以下を証明せよ

$$e^{A}|v\rangle = e^{\lambda}|v\rangle$$

例：$e^{i\gamma\sigma_z}$を適用する意味

σ_zは$\lambda = 1$のとき $|v\rangle = |0\rangle$、$\lambda = -1$のとき$|v\rangle = |1\rangle$。$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$のとき、

$$\begin{aligned} e^{i\gamma\sigma_z}|\psi\rangle &= e^{i\gamma\sigma_z}(\alpha|0\rangle + \beta|1\rangle) \\ &= e^{i\gamma}\alpha|0\rangle + e^{-i\gamma}\beta|1\rangle) \end{aligned}$$

σ_xもσ_yも、それぞれの固有値・固有ベクトルに対し、同様のことが言える

まとめ

量子ビットは

量子状態空間のベクトル

量子ゲートはユニタリー行列

Thank you

Kaori Namba
Senior Software Engineer

knamba@jp.ibm.com