Ciência de Dados Quântica 2021/22

Data Encoding

LUÍS PAULO SANTOS

Material de Consulta

- [Schuld2021] – Sec 3.4 (3.4.1 ; 3.4.2)
  Chap. 4 (4.1 ; 4.2)

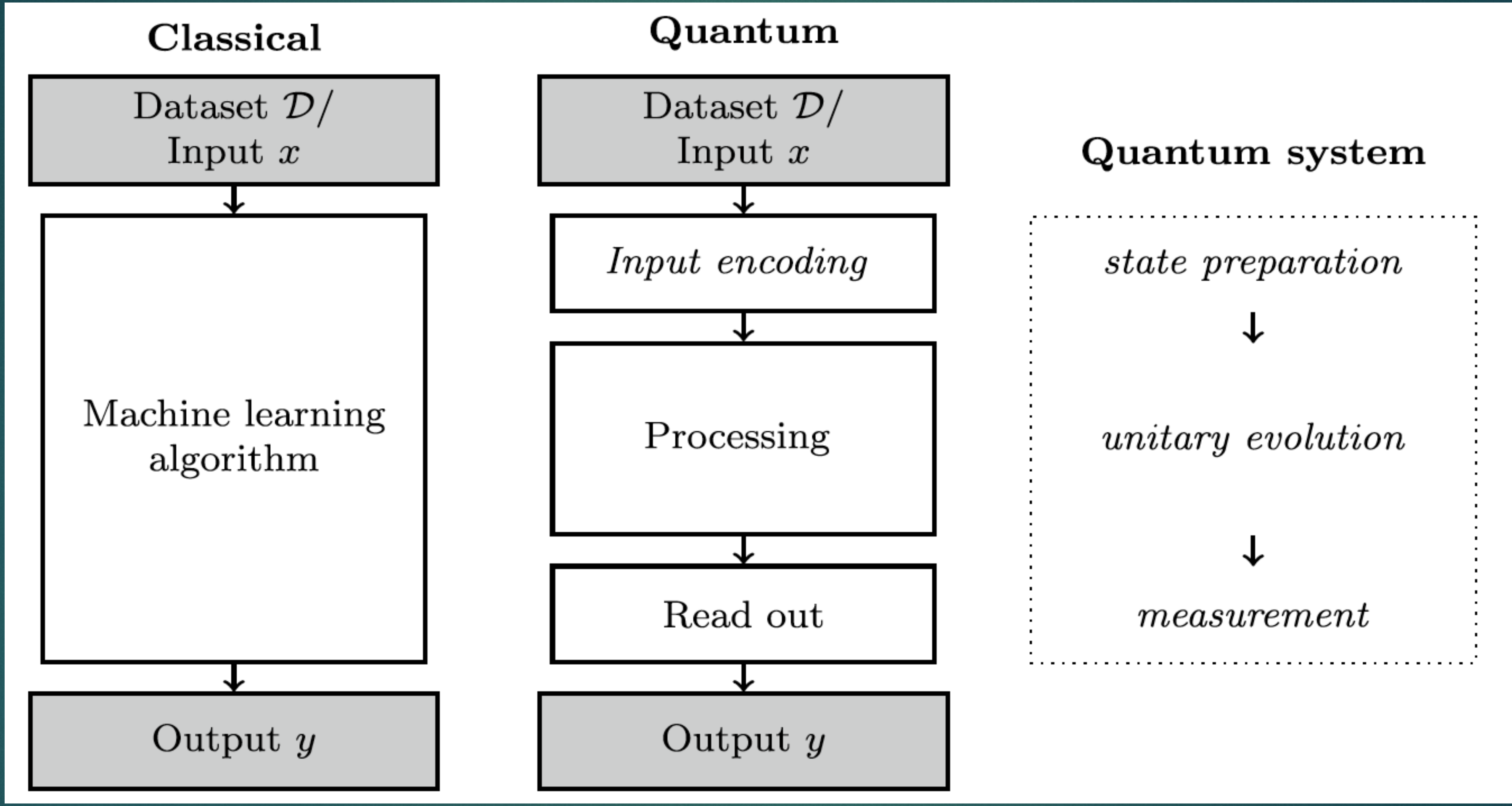
Data Encoding

Consider a dataset $\mathcal{D}$ with M data points, each a vector of N features

$$\mathcal{D} = \{x^1, \cdots, x^M\}, \text{ with } x^m \in \mathbb{R}^N, \text{ i.e. } x^m = \{x_1^m, \cdots, x_N^m\}^T$$

Data encoding (or state preparation) is the process of **representing** either a **single data point** or the **entire dataset** as a **quantum state**.

Data Encoding



[Schuld2021]

Data encoding and algorithm complexity

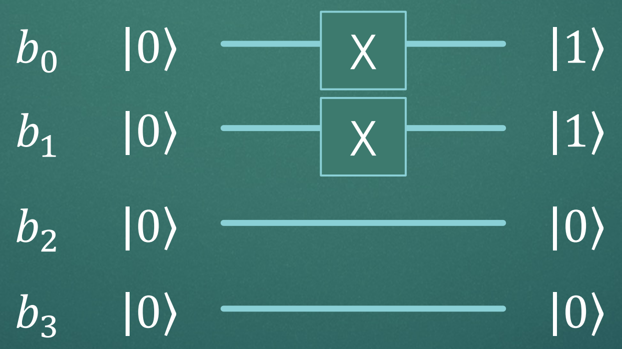
- In classical ML an efficient algorithm is $\mathcal{O}(M^s\, N^t)\,, s, t\ \geq 1$
- In quantum computing an efficient algorithm is $\mathcal{O}(n)$, n the nbr qubits
- State preparation is often the bottleneck of the quantum algorithm
- Theoretical frameworks, software and hardware that address the interface between the classical memory and the quantum device are central for runtime evaluations.

Encoding	# qubits
Basis	$N\tau$
Amplitude	$\log N$
Angle	N

Basis encoding

- Basis encoding associates a computational basis state of an n-qubit system with a classical n-bit string
- Exemplo: $|3\rangle = |0011\rangle$
- **Encoding a single data point** with τ qubits per feature and N features:

$$|3\rangle = |0011\rangle$$



Binary: signal + fixed point amplitude

$$b = b_s b_{\tau_{l-1}} \cdots b_1\ b_0 \,.\, b_{-1} b_{-2} \cdots b_{-\tau_r}$$

$$x = (-1)^{b_s} \left(b_{\tau_{l-1}} * 2^{\tau_{l-1}} + \cdots b_1 * 2^1 + b_0 * 2^0 + b_{-1} * 2^{-1} + \cdots + b_{-\tau_r} * 2^{-\tau_r}\right)$$

Basis encoding in superposition

$\mathcal{D} = \{x^1, \cdots, x^M\}$, where $x^m \in \mathcal{D}$ is given by $x^m = b^m = \{b_1^m, \cdots, b_n^m\}^T$

then a superposition of basis states can be prepared:

$$|\mathcal{D}\rangle = \frac{1}{\sqrt{M}} \sum_{m=1}^{M} |x^m\rangle$$

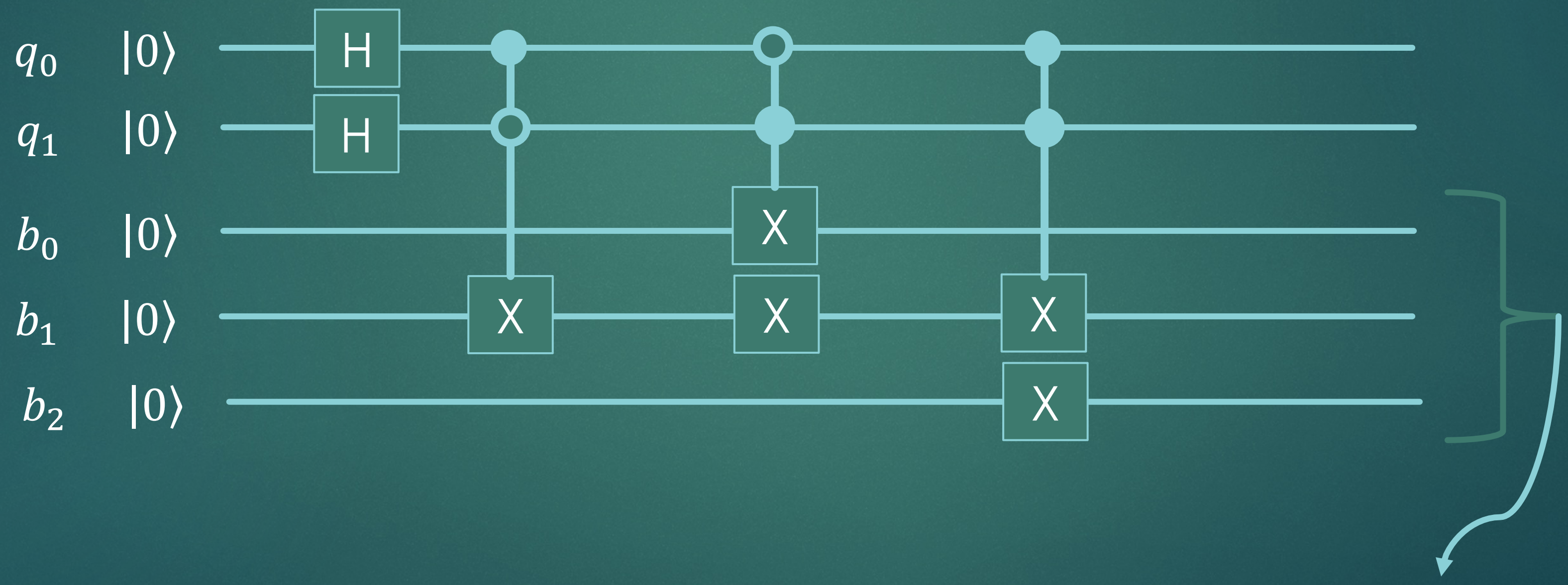
Example: $\mathcal{D} = \{(1,2), (0,3), (2,3)\} \quad \Leftrightarrow \mathcal{D} = \{(01,10), (00,11), (10,11)\}$

$$|\mathcal{D}\rangle = \frac{1}{\sqrt{3}}(|0110\rangle + |0011\rangle + |1011\rangle)$$

$$|\mathcal{D}\rangle = \left(0, 0, 0, \frac{1}{\sqrt{3}}, 0, 0, \frac{1}{\sqrt{3}}, 0, 0, 0, 0, \frac{1}{\sqrt{3}}, 0, 0, 0, 0\right)$$

Basis encoding in superposition

- **Example:** $\mathcal{D} = \{3,6,2\}$ ⇒ $\mathcal{D} = \{0,2,3,6\}$



$$\frac{1}{2}(|000\rangle + |010\rangle + |011\rangle + |110\rangle)$$

Basis encoding: conclusion

- Is expensive in terms of qubits: $NM\tau$
- The complexity of the state preparation circuit depends on the superposition, but in general can be $\mathcal{O}(2^n)$
- Simplest case: uniform superposition

$$|\mathcal{D}\rangle = \frac{1}{\sqrt{2^n}} \sum_{i=0}^{2^n-1} |i\rangle \qquad |\mathcal{D}\rangle = H^{\otimes n} |0\rangle^{\otimes n}$$

QRAM

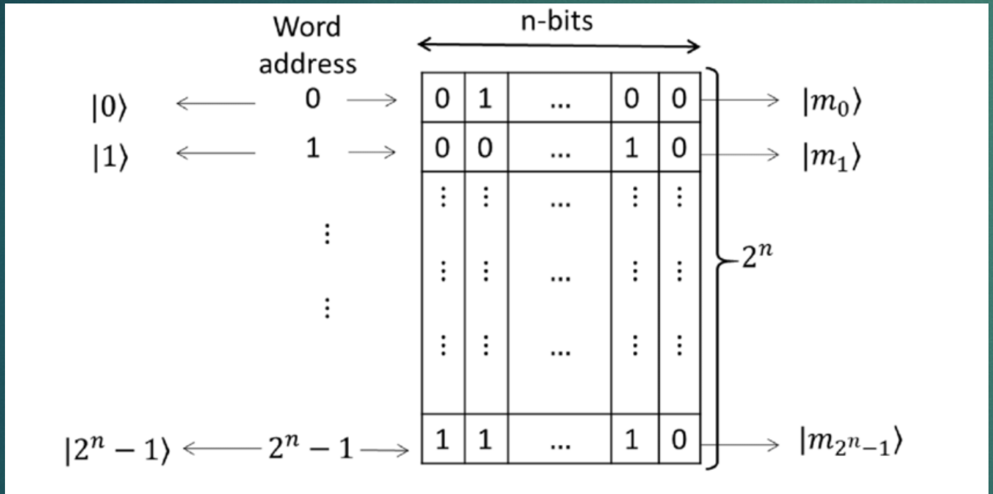
- Theoretical device that loads data patterns **from memory** to a **quantum state**:
  given an address $|m\rangle$ in a quantum register loads the corresponding data bit pattern to a second register :
  $$|m\rangle|0\cdots0\rangle \longrightarrow |m\rangle|x^m\rangle$$

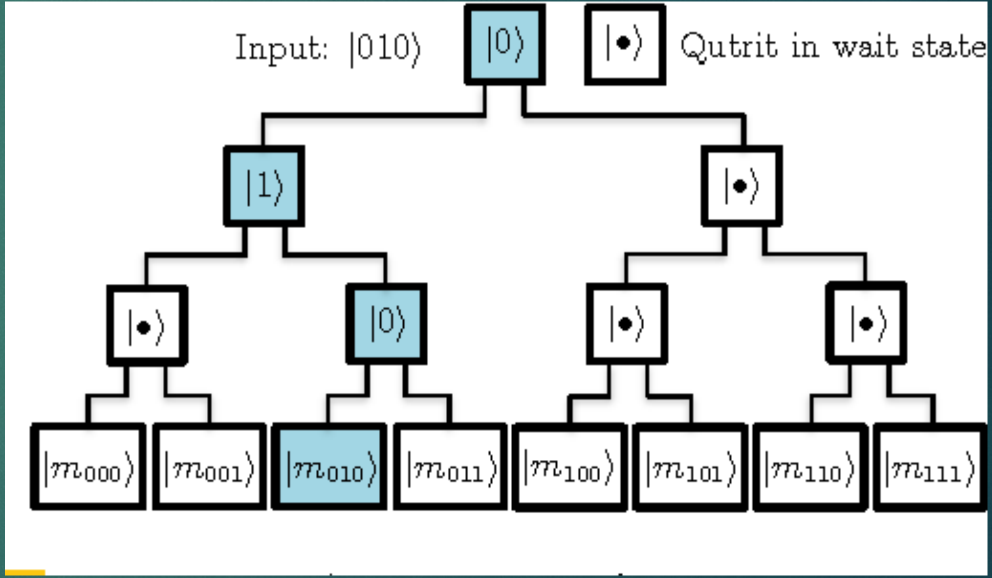
- More importantly, given a superposition of the address register, loads the corresponding data patterns to the superposition:

$$\frac{1}{\sqrt{M}}\sum_{m=0}^{M-1}|m\rangle|0\cdots0\rangle \longrightarrow \frac{1}{\sqrt{M}}\sum_{m=0}^{M-1}|m\rangle|x^m\rangle$$

QRAM

- Architectures have been proposed to realize this query in $\mathcal{O}(n)$ time, but its physical realisation remains an open challenge





Quantum Random Access Memory
Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone
Phys. Rev. Lett. **100**, 160501 – Published 21 April 2008

Angle Encoding

- Each feature of a data point is encoded as a rotation of a qubit. For $x = \{x_1, \cdots x_N\}$ and angle encoding N qubits and N rotation gates are required

$$|x\rangle = \otimes_{i=1}^{N} R_y(x_i)|0\rangle_i$$

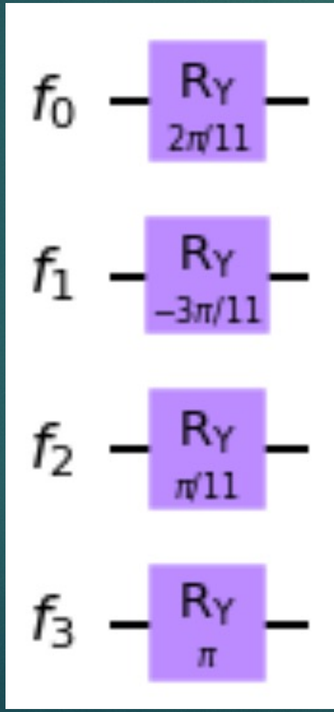
$$|q_i\rangle = \cos\left(\frac{x_i}{2}\right)|0\rangle + \sin\left(\frac{x_i}{2}\right)|1\rangle$$

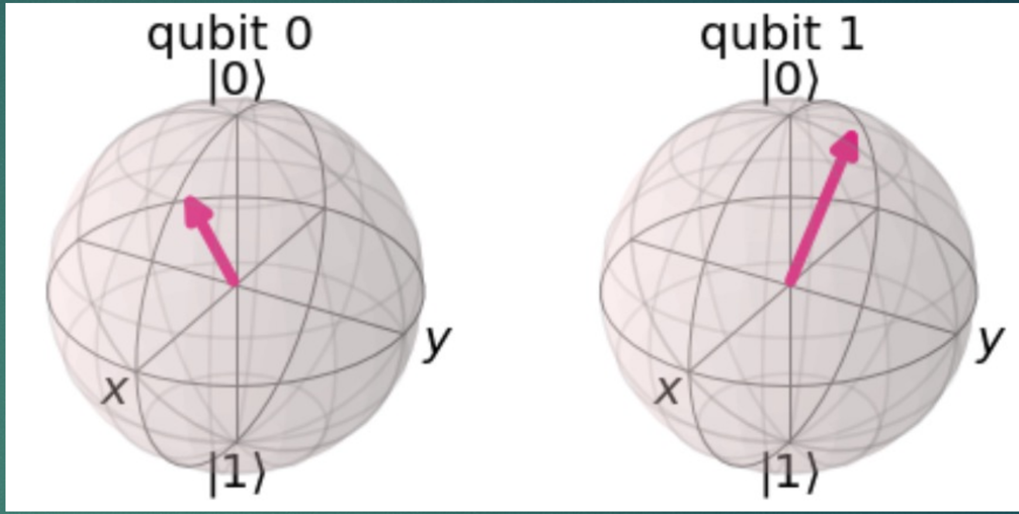
- Each feature has to be normalized in the interval $[-\pi\,,\pi]$

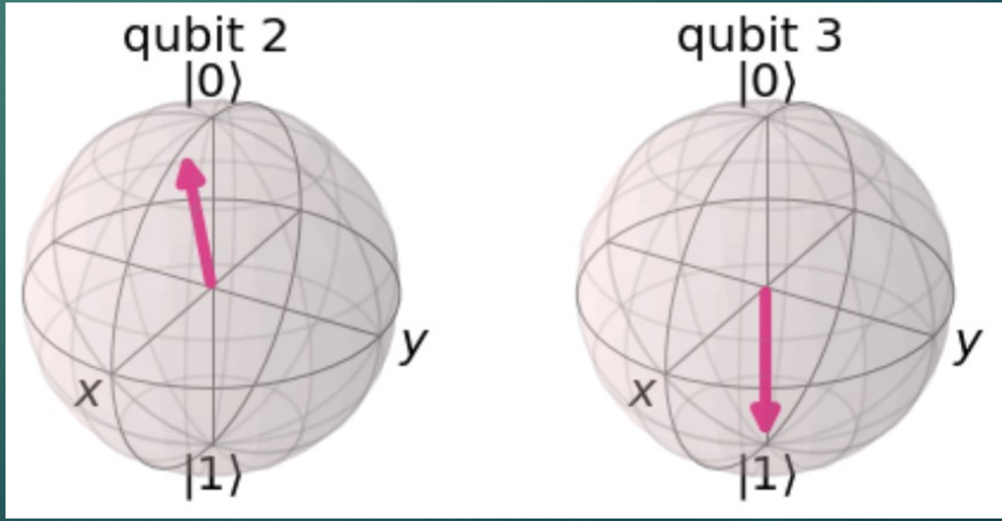
$$\widetilde{x_i} = \frac{x_i}{\max\limits_{x_j}\left(abs(x_j)\right)} * \pi$$

Angle Encoding: example

- $x = (2, -3, 1, 11)$
- $\max(abs(x)) = 11$
- $\tilde{x} = \left(\frac{2}{11}\pi, -\frac{3}{11}\pi, \frac{1}{11}\pi, \pi\right)$







Amplitude Encoding

- A real or complex valued vector $x \in \mathbb{C}^N$ is encoded into the amplitudes of a quantum state:

$$|\psi_x\rangle = \sum_{i=0}^{N-1} x_i \, |i\rangle$$

- x has to be normalised such that $\sum_{i=0}^{N-1} |x_i|^2 = 1$ and padded with zeros such that N is a power of 2
- An entire dataset can be encoded

$$|\psi_{\mathcal{D}}\rangle = \frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} \sum_{i=0}^{N-1} x_i \, |i\rangle |m\rangle = \frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} |\psi_{x^m}\rangle |m\rangle$$

Amplitude Encoding: example

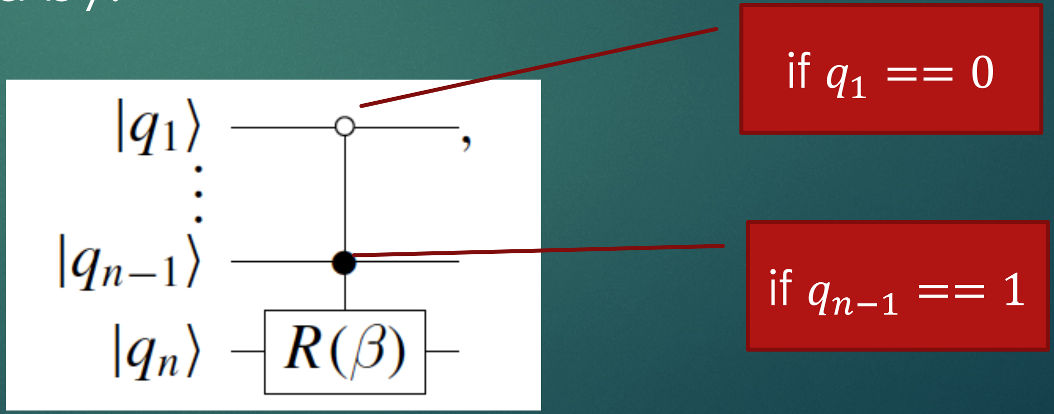
- $x = (0.1, -0.6, 1.0)$
- norm = $\sum|x_i|$
- $\tilde{x} = \frac{x}{norm} = (0.073, -0.438, 0.730, 0.000)$
- $|\psi_x\rangle = 0.073|00\rangle$ - 0.438 $|01\rangle + 0.730|10\rangle + 0.000|11\rangle$

- Amplitude encoding requires only $\log(MN)$ qubits

- The theoretical lower bound of the depth of an arbitrary state preparation circuit is known to be $\frac{1}{n}2^n$
  but currently known algorithms perform slightly worse

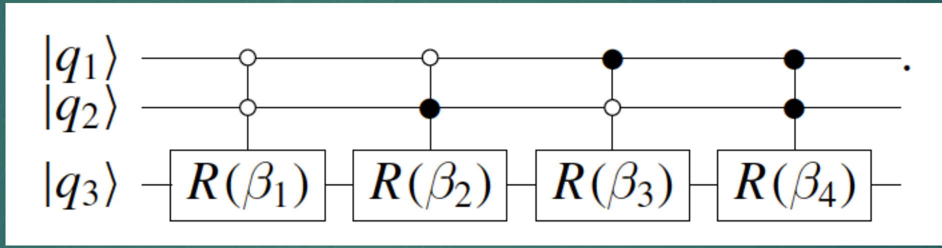
Amplitude Encoding: Mottonen et al.

- The inverse circuit is designed by considering the reverse problem of mapping the arbitrary state $|\psi\rangle$ to the ground state $|0 \cdots 0\rangle$
- The basic idea is to control a rotation on qubit q_s by all possible states of the previous qubits $q_1 \cdots q_{s-1}$, using sequences of so-called multi-controlled rotations.
- A controlled rotation is depicted by:

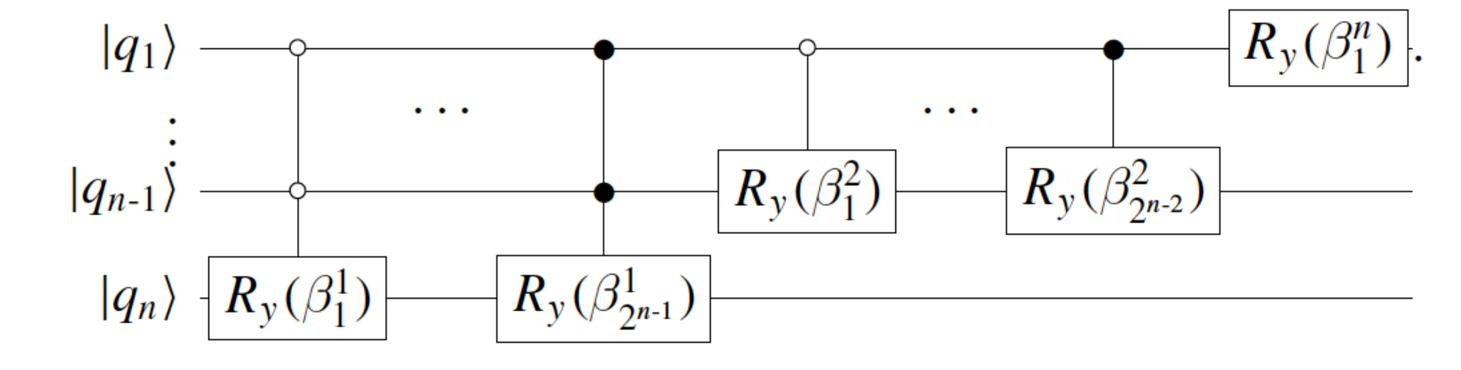


Amplitude Encoding: Mottonen et al.

- Representing conditional rotations on all possible states of the previous qubits:



- The final cascade:



[Schuld2021]

Amplitude Encoding: Mottonen et al.

- The rotation angles β are given by

$$\beta_j^s = 2 \arcsin \left(\frac{\sqrt{\sum_{l=1}^{2^{s-1}} |\alpha_{(2j-1)2^{s-1}+l}|^2}}{\sqrt{\sum_{l=1}^{2^s} |\alpha_{(j-1)2^s+l}|^2}} \right).$$

- This algorithm is still $\mathcal{O}(2^n)$
- It is the algorithm used by Qiskit `initialize()`:

```
qc = QuantumCircuit (1)
state = [1/sqrt(2), 1j/sqrt(2)]
qc.initialize(state, 0)
```

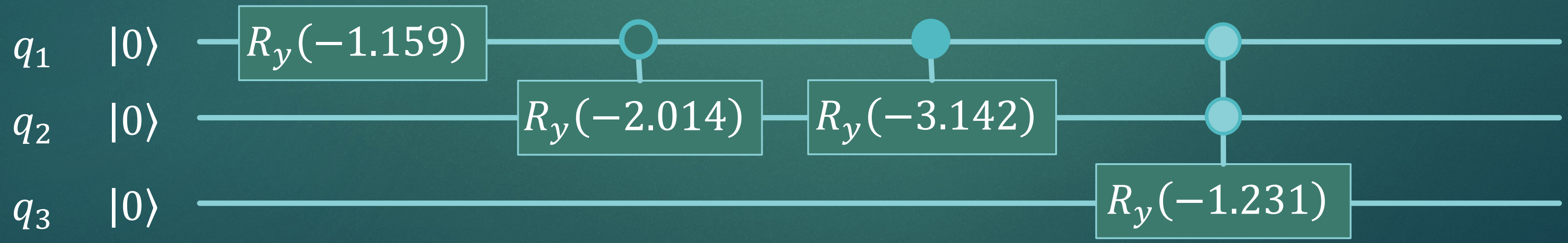
Amplitude Encoding: example

- $|\psi\rangle = \sqrt{0.2}|000\rangle + \sqrt{0.5}|010\rangle + \sqrt{0.2}\,|110\rangle + \sqrt{0.1}\,|111\rangle$

$$\begin{aligned}
&c_{q_1=0}c_{q_2=0}\; R_y^{q_3}(\beta_1^1), && \beta_1^1 = 0,\\
&c_{q_1=0}c_{q_2=1}\; R_y^{q_3}(\beta_2^1), && \beta_2^1 = 0,\\
&c_{q_1=1}c_{q_2=0}\; R_y^{q_3}(\beta_3^1), && \beta_3^1 = 0,\\
&c_{q_1=1}c_{q_2=1}\; R_y^{q_3}(\beta_4^1), && \beta_4^1 = 1.231\ldots,
\end{aligned}$$

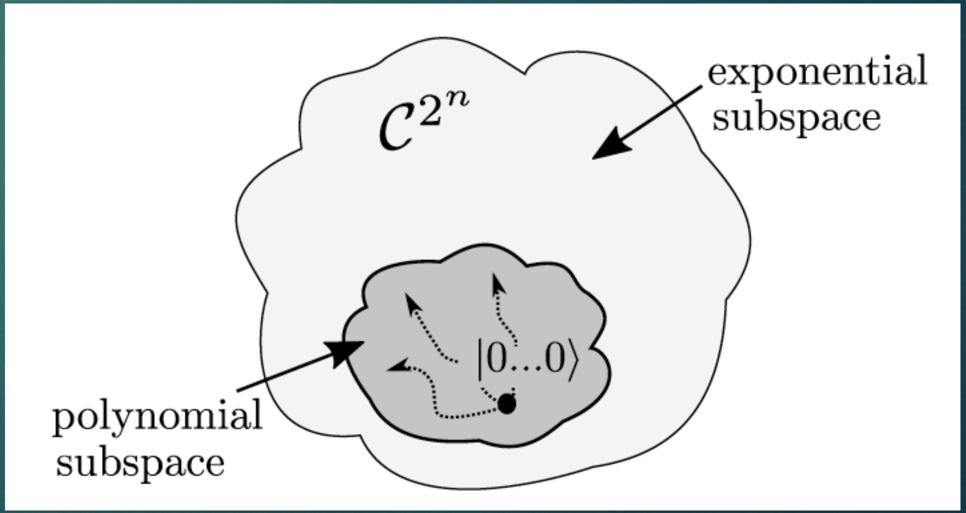
$$\begin{aligned}
&c_{q_1=0}\; R_y^{q_2}(\beta_1^2), && \beta_1^2 = 2.014\ldots,\\
&c_{q_1=1}\; R_y^{q_2}(\beta_2^2), && \beta_2^2 = 3.142\ldots,\\
&R_y^{q_1}(\beta_1^3), && \beta_1^3 = 1.159\ldots.
\end{aligned}$$

- Invertendo o circuito:



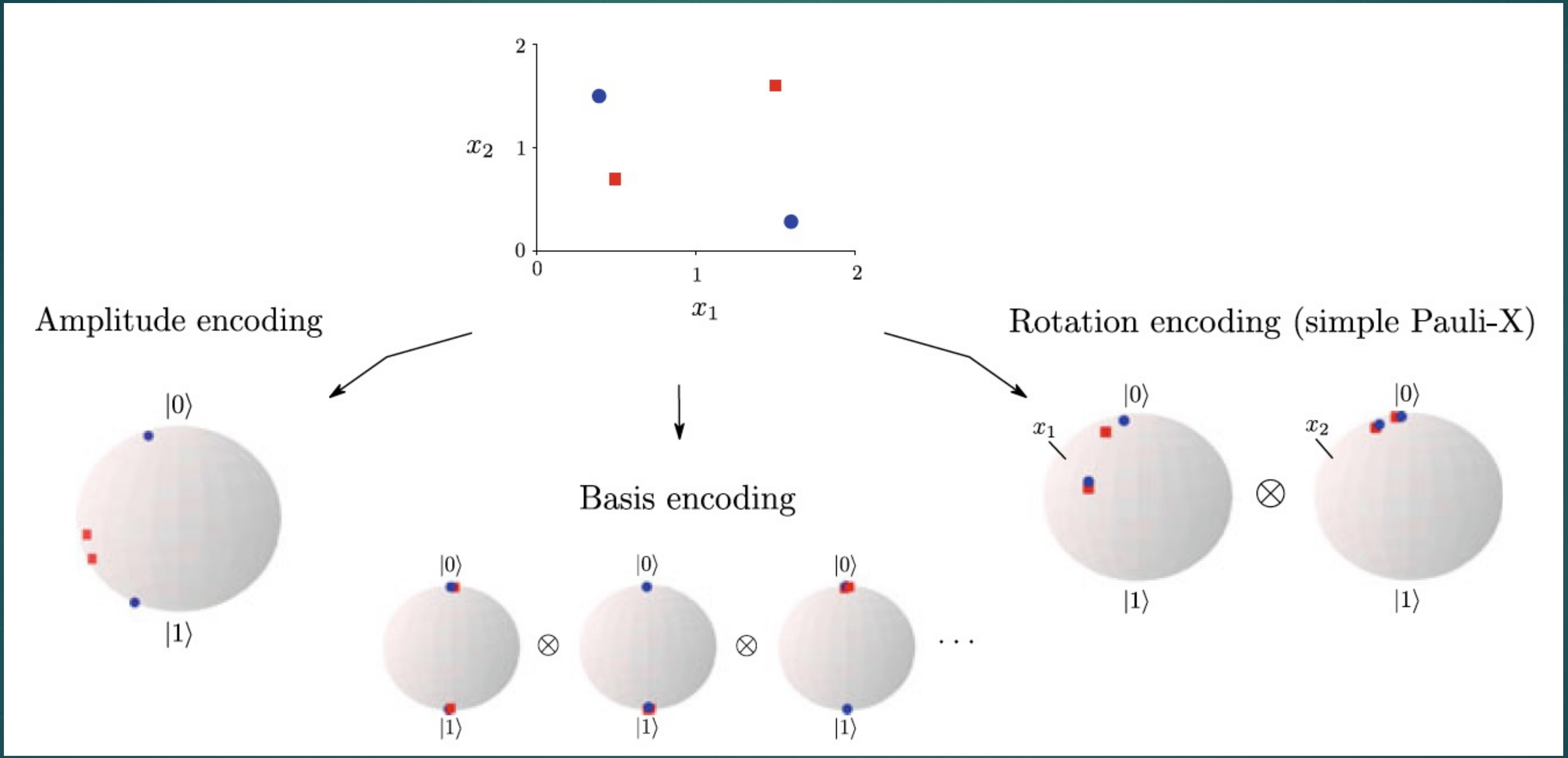
Amplitude encoding: conclusions

- It is an exponentially compact representation:
  requires only n = log N qubits to encode an input of N features, and
  n = log(NM) qubits for an M points dataset
- But is there an algorithm that only uses log(NM) gates to prepare amplitude-encoded states?
  - it is a strange concept since loading the N features from the memory hardware takes time that is linear in N
- It is possible only when the dataset exhibits structure:
  - uniform superposition,
  - very sparse data sets



[Schuld2021]

Data encoding: Bloch visualization



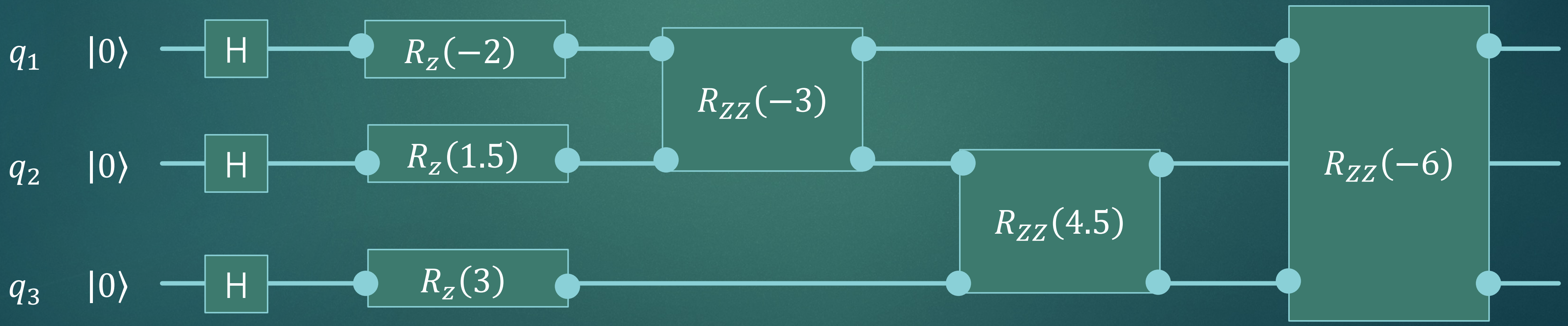
[Schuld2021]

Instantaneous quantum polynomial: IQP

$$|x\rangle = \left(U_Z(x) H^{\otimes n}\right)^r |0\rangle^{\otimes n}$$

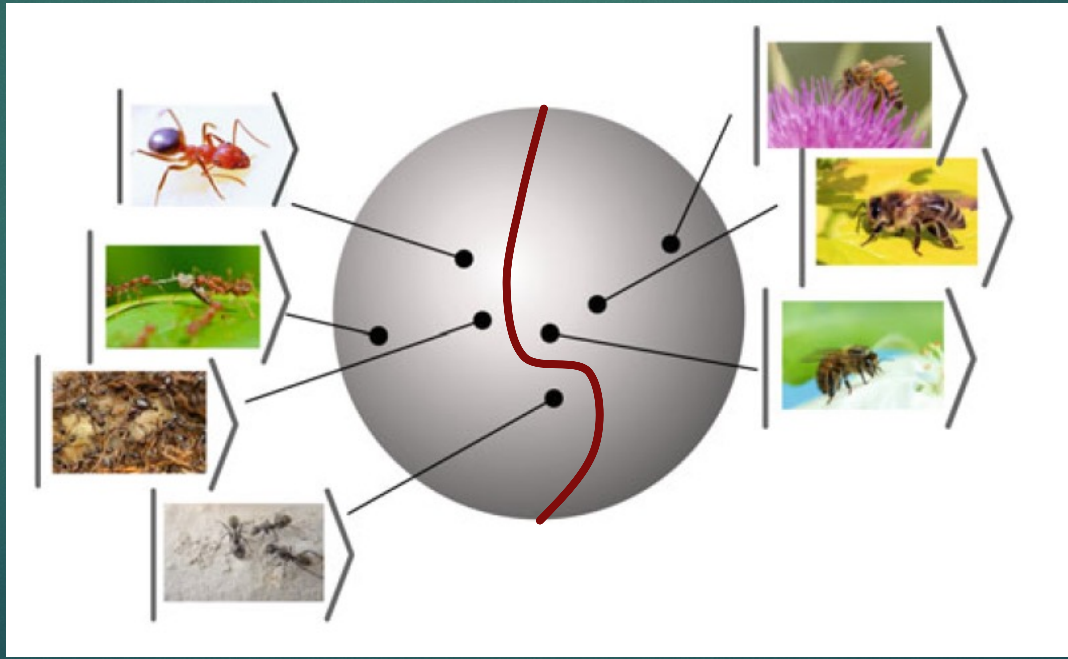
$$U_Z(x) = \prod_{(i,j)\in S} R_{Z_i Z_j}(x_i * x_j) \bigotimes_{k=1\ldots n} R_z(x_k)$$

$\mathcal{D} = \{-2, 1.5, 3\}, r = 1$



Data encoding as a feature map

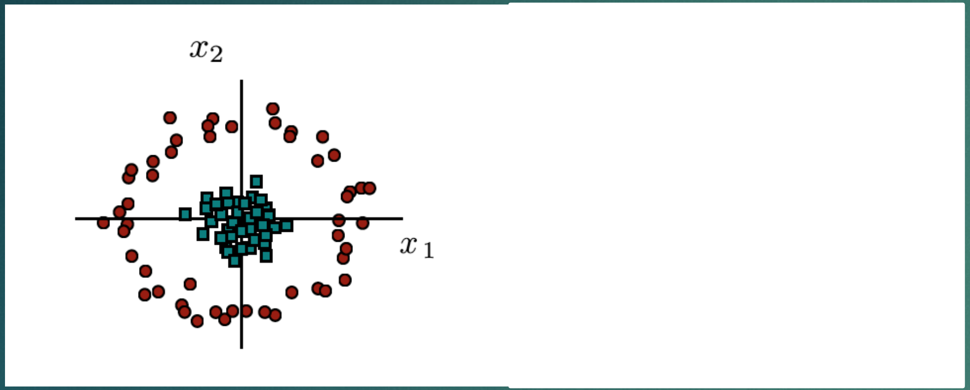
- Data encoding **maps** data points from the input space $\mathcal{X}$ into the quantum Hilbert space $\mathbb{C}^{2^n}$
- The data encoding feature map can **change the structure of the data** in non-trivial (and non linear) manners eventually **determining the success of the learning algorithm**



[Schuld2021]

Data encoding as a feature map

- Feature maps, more often than not, increase the dimensionality of the data



$$\emptyset((x_1, x_2)) = (x_1, x_2,\ 0.5\ ({x_1}^2 + {x_2}^2))$$

- "Hilbert space is a big space" [Carlton Caves]
- Quantum computation accesses an exponential Hilbert state space: 2^n basis states for n qubits
- Data encoding allows mapping classical data (eventually non-linearly) into this hyperdimensional state space

2^{275} > #atoms in the observable Universe