Putting quantum machine learning algorithms to the test

4th QIPCC conference, 2016
Cape Town, South Africa

Mark Fingerhuth

Maastricht University, The Netherlands
Thesis work at Centre for Quantum Technology, UKZN, South Africa

Table of contents

Introduction

Quantum Computing & Qubits

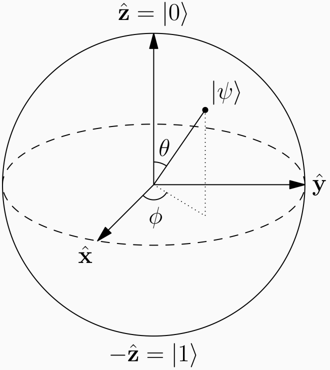


Figure 1: Arbitrary two-dimensional qubit $|\psi\rangle$ visualized on the Bloch sphere[1]

Most general form of a 2-D qubit:

$$|q\rangle = \alpha |0\rangle + \beta |1\rangle \qquad (1)$$

where $\alpha, \beta \in \mathbb{C}$.

Can also be visualized in spherical polar coords on the unit or Bloch sphere as follows:

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

where $0 \leq \theta \leq \pi$ and $0 \leq \phi \leq 2\pi$

[1] Reprinted from Wikipedia, n.d., Retrieved September 7, 2016, from `https://en.wikipedia.org/wiki/Bloch_Sphere`. Copyright 2012 by Glosser.ca. Reprinted with permission.

Classical Machine Learning

- Approximately 2.5 quintillion (10^{18}) bytes of digital data are created every day[1]
- Need for advanced algorithms that can make sense of data content, retrieve patterns and reveal correlations $\rightarrow$ Machine learning (ML)
- ML algorithms often involve
  - solving large systems of linear equations
  - inverting large matrices
  - distance computations
- Performing these computations on large data sets gets increasingly difficult[2]

[1] IBM. (2016). What is big data? https://www-01.ibm.com/software/data/bigdata/what-is-big -data.html. (Accessed: 2016-09-08)
[2] Bekkerman, R., Bilenko, M., & Langford, J. (2011). Scaling up machine learning: Parallel and distributed approaches. Cambridge University Press.

Classical Machine Learning

Machine learning can be subdivided into three major fields.

Supervised ML

- Based on *input* and *output* data

"I know how to classify this data but I need the algorithm to do the computations for me."

Unsupervised ML

- Based on *input* data only

"I have no clue how to classify this data, can the algorithm create a classifier for me?"

Reinforcement learning

- Based on *input* data only

"I have no clue how to classify this data, can the algorithm classify this data and I'll give it a reward if it's correct or I'll punish it if it's not."

Classical Machine Learning

Machine learning can be subdivided into three major fields.$\mathcal{O}(inputsize)$

Supervised ML

- Based on *input* and *output* data

"I know how to classify this data but I need the algorithm to do the computations for me."

Unsupervised ML

- Based on *input* data only

"I have no clue how to classify this data, can the algorithm create a classifier for me?"

Reinforcement learning

- Based on *input* data only

"I have no clue how to classify this data, can the algorithm classify this data and I'll give it a reward if it's correct or I'll punish it if it's not."

Quantum Machine Learning

Some general info about QML. How can quantum computing aid classical machine learning?

Experimental realizations so far

Until now there have been only few experimental verifications of QML algorithms that establish proof- of-concept. Li, Liu, Xu, and Du (2015) successfully distinguished a handwritten six from a nine using a quantum support vector machine on a four-qubit nuclear magnetic resonance test bench. In addition, Cai et al. (2015) were first to experimentally demonstrate quantum machine learning on a photonic QC and showed that the distance between two vectors and their inner product can indeed be computed quantum mechanically. Lastly, Rist et al. (2015) solved a learning parity problem with five superconducting qubits and found that a quantum advantage can already be observed in non error-corrected systems.

Classical k-nearest neighbour

Some description goes here.

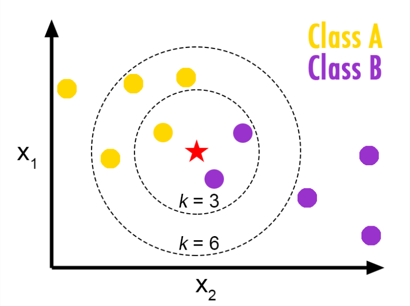


Figure 2: Visualization of a kNN classifier[1]

[1] Reprinted from GitHub, Burton de Wilde, Retrieved September 13, 2016, from http://bdewilde.github.io/blog/blogger/2012/10/26/classification-of-hand-written-digits-3/. Copyright 2012 by Burton de Wilde. Reprinted with permission.

Quantum k-nearest neighbour

Two different algorithms with respect to initial state preparation:

Data encoded into qubits

k-dimensional probability vector requires $4k$ classical bits which are encoded one-to-one into $4k$ qubits, e.g.

$$\begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix} * 10 \rightarrow \begin{pmatrix} 6 \\ 4 \end{pmatrix} \rightarrow \begin{pmatrix} 0110 \\ 0100 \end{pmatrix} \rightarrow n = 01100100 \rightarrow |n\rangle = |01100100\rangle$$

Data encoded into amplitudes

k-dimensional probability vector is encoded into $log_2(k)$ qubits, e.g.

$$\begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix} \quad \rightarrow \quad |n\rangle = \sqrt{0.6}\,|0\rangle + \sqrt{0.4}\,|1\rangle$$

Quantum k-nearest neighbour

Two different algorithms with respect to initial state preparation:

Data encoded into qubits

k-dimensional probability vector requires $4k$ classical bits which are encoded one-to-one into $4k$ qubits, e.g.

$$\begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix} * 10 \rightarrow \begin{pmatrix} 6 \\ 4 \end{pmatrix} \rightarrow \begin{pmatrix} 0110 \\ 0100 \end{pmatrix} \rightarrow n = 01100100 \rightarrow |n\rangle = |01100100\rangle$$

Data encoded into amplitudes

k-dimensional probability vector is encoded into $log_2(k)$ qubits, e.g.

$$\begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix} \quad \rightarrow \quad |n\rangle = \sqrt{0.6}\,|0\rangle + \sqrt{0.4}\,|1\rangle$$

Amplitude-based kNN algorithm

The algorithm

$$\frac{1}{\sqrt{2M}}\sum_{m=1}^{M}(|0\rangle|\Psi_{\tilde{x}}\rangle + |1\rangle|\Psi_{x^m}\rangle)|y^m\rangle|m\rangle \tag{3}$$

where

$$|\Psi_{\tilde{x}}\rangle = \sum_{i=1}^{N}\tilde{x}_i|i\rangle \qquad |\Psi_{x^m}\rangle = \sum_{i=1}^{N}x_i^m|i\rangle \tag{4}$$

$$\frac{1}{2\sqrt{M}}\sum_{m=1}^{M}(|0\rangle[|\Psi_{\tilde{x}}\rangle + |\Psi_{x^m}\rangle] + |1\rangle[|\Psi_{\tilde{x}}\rangle - |\Psi_{x^m}\rangle])|y^m\rangle|m\rangle \tag{5}$$

After successful conditional measurement, the state is proportional to

$$\frac{1}{2\sqrt{M}}\sum_{m=1}^{M}\sum_{i=1}^{N}(\tilde{x}_i + x_i^m)|0\rangle|i\rangle|y^m\rangle|m\rangle \tag{6}$$

Algorithmic complexity

$\mathcal{O}(\frac{1}{p_{acc}})$ where p_{acc} is the probability of measuring ancilla in the $|0\rangle$ state

Maria Schuld (2016), unpublished

Simple binary classification case

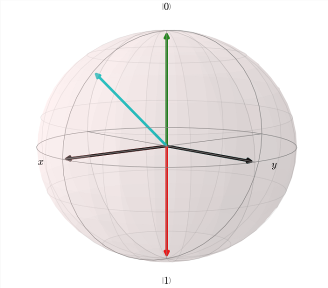


Figure 3: Simple binary classification problem of a quantum state

$$\frac{1}{\sqrt{2M}}\sum_{m=1}^{M}(|0\rangle|\Psi_{\tilde{x}}\rangle + |1\rangle|\Psi_{x^m}\rangle)|y^m\rangle|m\rangle \tag{7}$$

where

$$|\Psi_{\tilde{x}}\rangle = \sum_{i=1}^{N}\tilde{x}_i|i\rangle \qquad |\Psi_{x^m}\rangle = \sum_{i=1}^{N}x_i^m|i\rangle \tag{8}$$

Procedure to load the input vector $\tilde{x}$:

$$|\Psi_0\rangle = \frac{1}{2}\sum_{m=1}^{2}(|0\rangle|0\rangle + |1\rangle|0\rangle)|y^m\rangle|m\rangle \tag{9}$$

Apply controlled rotation ${}_0^1CR_y(\frac{\pi}{4})$ s.t.

$${}_0^1CR_y(\frac{\pi}{4})|\Psi_0\rangle = |\Psi_1\rangle = \frac{1}{2}\sum_{m=1}^{2}(|0\rangle|0\rangle + |1\rangle|\Psi_{\tilde{x}}\rangle)|y^m\rangle|m\rangle \tag{10}$$

Flip the ancilla qubit in the first register

$$(X\otimes\mathbb{1}\otimes\mathbb{1}\otimes\mathbb{1})|\Psi_1\rangle = |\Psi_2\rangle = \frac{1}{2}\sum_{m=1}^{2}(|0\rangle|\Psi_{\tilde{x}}\rangle + |1\rangle|0\rangle)|y^m\rangle|m\rangle \tag{11}$$

Implementation with IBM's quantum computer

Minipage 1

Minipage 2

Controlled U gate

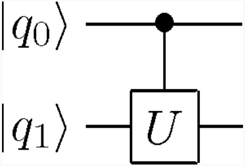


Figure 4: Controlled U-gate

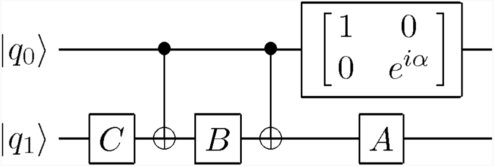


Figure 5: Decomposition of a controlled U-gate[1]

Choose A,B,C and α s.t.

$$e^{i\alpha} * A * X * B * X * C = U \quad and \quad A * B * C = \mathbb{1} \tag{12}$$

Need to solve the following equation[1]

$$U = \begin{pmatrix} e^{i(\alpha - \frac{\beta}{2} - \frac{\delta}{2})} \cos \frac{\gamma}{2} & -e^{i(\alpha - \frac{\beta}{2} + \frac{\delta}{2})} \sin \frac{\gamma}{2} \\ e^{i(\alpha + \frac{\beta}{2} - \frac{\delta}{2})} \sin \frac{\gamma}{2} & e^{i(\alpha + \frac{\beta}{2} + \frac{\delta}{2})} \cos \frac{\gamma}{2} \end{pmatrix} \tag{13}$$

Algorithmic complexity

$\mathcal{O}(\frac{1}{p_{acc}}) + \mathcal{O}(k)$ where k is number of root finding iterations[2]

[1] Nielsen, M. A., & Chuang, I. L. (2010). Quantum computation and quantum information. Cambridge University Press.
[2] Jat, R. N., & Ruhela, D. S. (2011). Comparative study of complexity of algorithms for iterative solution of non-linear equations. Journal of International Academy Of Physical Sciences, 15(4).

Problems with universal gate sets

In our case we need to find A, B, C and α for ${}^1_0CR_y(\frac{\pi}{4})$:

Using a root finding algorithm for non-linear equations we find:

$$\alpha = \pi; \quad \beta = 2\pi; \quad \delta = \frac{7}{8}\pi; \quad \gamma = 0 \tag{14}$$

Then,

$$A = R_z(\beta)R_y(\frac{\gamma}{2}) = R_z(2\pi) = XZXZ \tag{15}$$

$$B = R_y(-\frac{\gamma}{2})R_z(-\frac{\delta+\beta}{2}) = R_z(-\frac{23}{16}\pi) = ??? \tag{16}$$

$$C = R_z(\frac{\delta-\beta}{2}) = R_z(-\frac{9}{16}\pi) = ??? \tag{17}$$

$$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi} \end{pmatrix} = Z \tag{18}$$

[1] Dawson, C. M., & Nielsen, M. A. (2005). The Solovay-Kitaev algorithm. arXiv preprint quant-ph/0505030.

The Solovay-Kitaev theorem

The Solovay-Kitaev theorem guarantees that given a set of single-qubit quantum gates which generates a dense subset of $SU(2)$, then that set is guaranteed to fill $SU(2)$ quickly.[1]

$\rightarrow$ **Hence, given any universal gate set it is possible to obtain good approximations to any desired gate.**

$\rightarrow$ **But needs to be computed classically!**

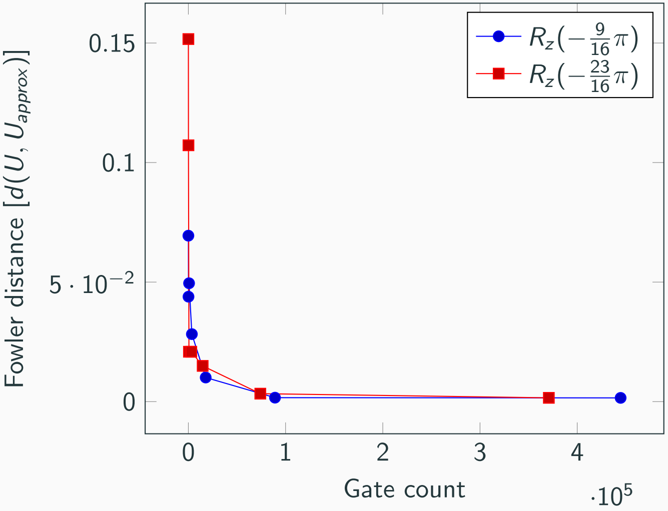
Algorithmic complexity

$\mathcal{O}(\frac{1}{p_{acc}}) + \mathcal{O}(k) + \mathcal{O}(m * log^{2.71}(\frac{m}{\epsilon}))$ for ϵ-approximations of m gates[1]

[1] Dawson, C. M., & Nielsen, M. A. (2005). The Solovay-Kitaev algorithm. arXiv preprint quant-ph/0505030.

The Solovay-Kitaev algorithm

What is fowler distance?



The Solovay-Kitaev algorithm

IBM's quantum computer needs **130ns for single-qubit gates** and **500ns for CNOT gates.**

Qubit decoherence times:

$49.5\,\mu s \leq T1 \leq 85.3\,\mu s$
$56.0\,\mu s \leq T2 \leq 139.7\,\mu s$

Approx. Gate	Distance	Gate count	Execution time
$R_z(-\frac{9}{16}\pi)$	0.04389	121	15.7 μs
	0.02823	3,622	470.9 μs
	0.004698	20,496	2664.5 μs

Table 1: SK algorithm results

Liqui$|\rangle$ simulations

Show that the simple binary classification problem works in Liquid since we can directly implement

Qubit-based kNN quantum algorithm

Typography

```
The theme provides sensible defaults to
\emph{emphasize} text, \alert{accent} parts
or show \textbf{bold} results.
```

becomes

The theme provides sensible defaults to *emphasize* text, accent parts or show **bold** results.

Font feature test

- Regular
- *Italic*
- SmallCaps
- **Bold**
- ***Bold Italic***
- **Bold SmallCaps**
- `Monospace`
- *`Monospace Italic`*
- **`Monospace Bold`**
- ***`Monospace Bold Italic`***

Lists

Items

- Milk
- Eggs
- Potatos

Enumerations

1. First,
2. Second and
3. Last.

Descriptions

PowerPoint Meeh.

Beamer Yeeeha.

Animation

- This is important

Animation

- This is important
- Now this

Animation

- This is important
- Now this
- And now this

Animation

- This is really important
- Now this
- And now this

Figures

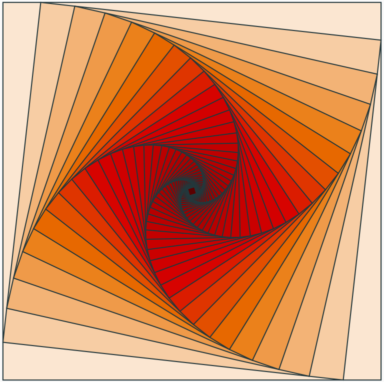


Figure 6: Rotated square from texample.net.

Tables

Table 2: Largest cities in the world (source: Wikipedia)

City	Population
Mexico City	20,116,842
Shanghai	19,210,000
Peking	15,796,450
Istanbul	14,160,467

Blocks

Three different block environments are pre-defined and may be styled with an optional background color.

Default

Block content.

Alert

Block content.

Example

Block content.

Default

Block content.

Alert

Block content.

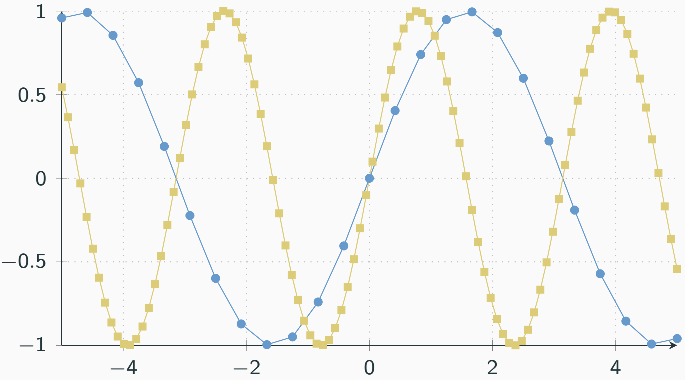
Example

Block content.

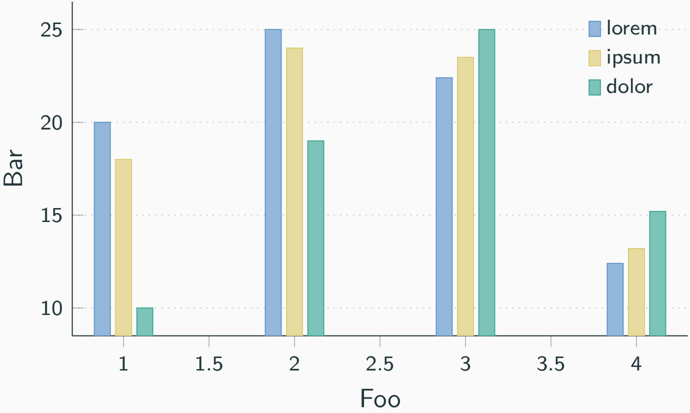
Math

$$e = \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^n$$

Line plots



Bar charts



Quotes

Veni, Vidi, Vici

Conclusion

Summary

sefsefesfsefsef

References

Some references to showcase [allowframebreaks] [**?**, **?**, **?**, **?**, **?**]

Questions?

Backup slide I

fefesfesfesfefesf

Backup slide II I

IBM.

What is big data?

https://www-01.ibm.com/software/data/bigdata/what-is-big-data.html, 2016.

Accessed: 2016-09-08.