Dimensional Reduction

Charles Hadfield

Mathematician

IBM Quantum, IBM Thomas J Watson Research Center, New York

IBM **Quantum**

Outline of talk

Quantum Machine Learning
Dimensional Reduction

Part 0: Introduction

Part 1: Slow Feature Analysis

Part 2: Clustering with SFA

Part 3: Clustering with Quantum SFA

Part 4: Alternative Dimensional Reduction Techniques

Part 0: Introduction

Quantum Machine Learning

Dimensional Reduction

A Toy Example

Consider all smooth functions on the unit interval

$$C^\infty([0,1];\mathbb{R})$$

How can we (minimally) describe a function?

$$f : [0,1] \to \mathbb{R}$$

- Taylor approximation
  (to some approximation)
- Harmonic approximation
  (to some approximation)
- Low/High pass filter
  (within some thresholds)

A Toy Example

Consider all smooth functions on the unit interval

$$C^\infty([0,1];\mathbb{R})$$

How can we (minimally) describe a function?

$$f\,:\,[0,1]\to\mathbb{R}$$

Decompose the space with respect to an operator Δ

And record the coefficients of the *dominant* eigenfunctions

- Taylor approximation
  $$\Delta = -(x\partial_x)^2$$
- Harmonic approximation
  $$\Delta = -\partial_x^2$$

$$C^\infty([0,1];\mathbb{R}) = \mathrm{span}\,\{\varphi_i\}$$

$$f = \sum_i f_i\varphi_i$$

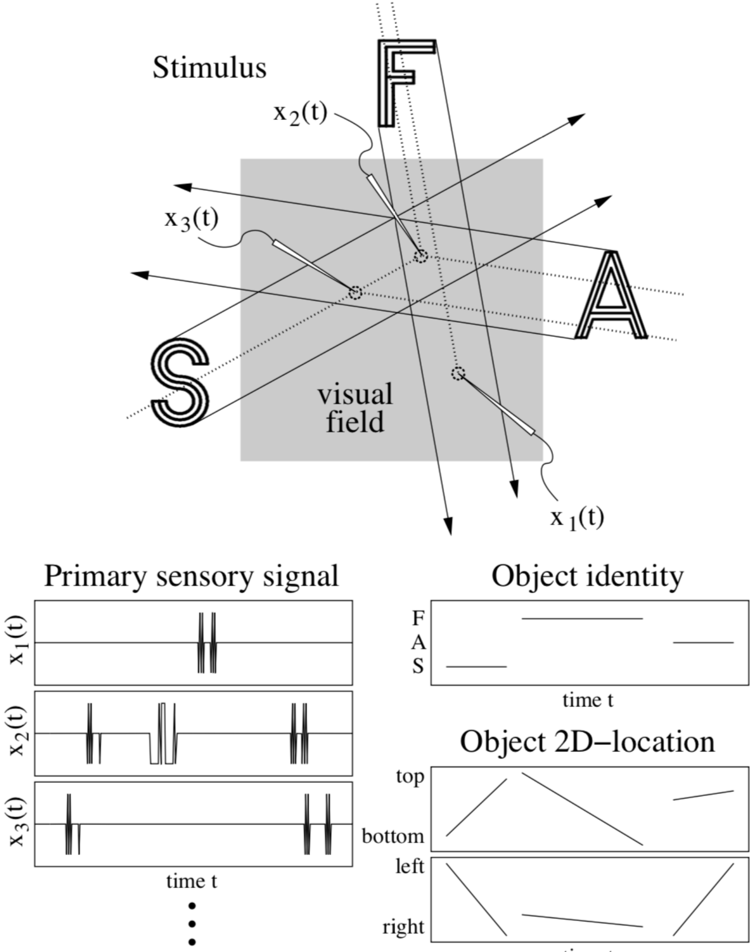
Parts 1,2,3: Slow Feature Analysis

	Unsupervised	Supervised
Classical	2002	2005
Quantum		2018

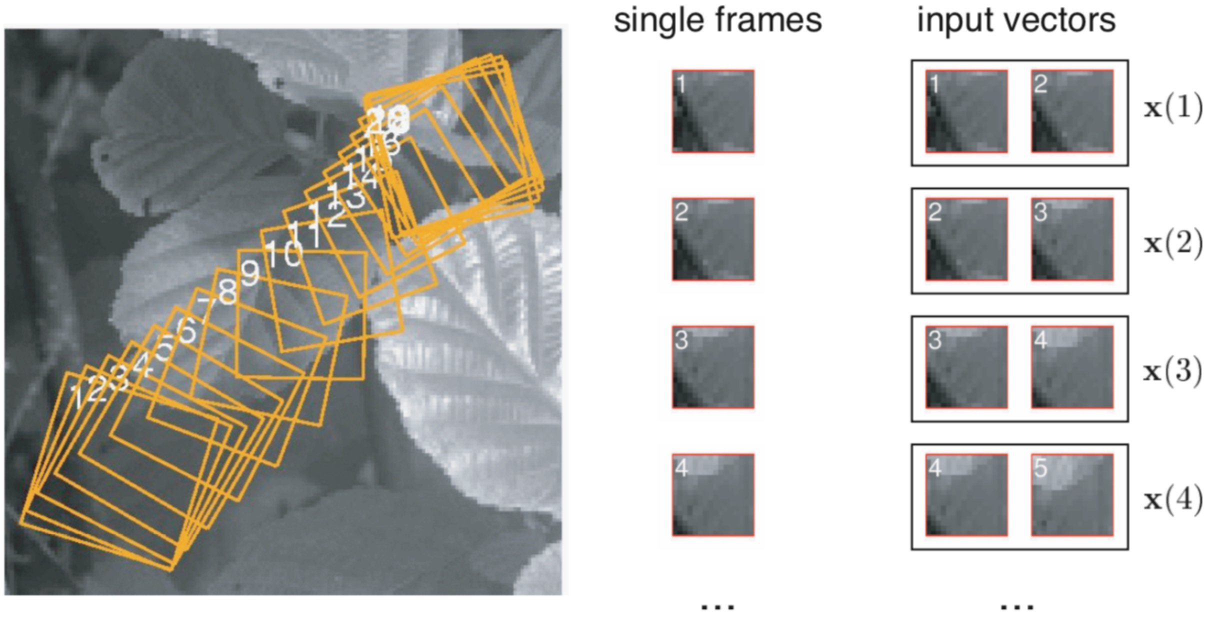
Part 1: Slow Feature Analysis

Quantum Machine Learning

Dimensional Reduction



Wiskott, Sejnowski
Slow Feature Analysis: Unsupervised Learning of Invariances
Neural computation, 2002



Berkes, Wiskott
Slow feature analysis yields a rich repertoire of complex cell properties
Journal of Vision, 2005

The problem statement of SFA

Given a *high* dimensional input signal, find a transformation into a *low* dimensional output signal which varies *slowly* and carries *significant* information.

Input signal $x \sim \begin{pmatrix} x_1(t) \\ \vdots \\ x_N(t) \end{pmatrix} \in \mathbb{R}^N \times [0, T]$

Dimensional reduction transformation $G : \mathbb{R}^N \to \mathbb{R}^K$

What kind of *G* will we allow?

Output signal $y(t) = G(x(t))$

Minimise Energy

subject to:
unit variance (and zero mean)
decorrelation

Minimise: $E = \sum_{i \in [K]} \langle (\dot{y}_i)^2 \rangle_t$

subject to: $\langle y_i \rangle_t = 0$

$\langle y_i^2 \rangle_t = 1$

$\langle y_i \cdot y_j \rangle_t = \delta_{ij}$

$\langle \bullet \rangle_t = \int_0^T dt \bullet (t) = \text{time average of } \bullet$

Linear dimensional reductions

Assume that *G* is a *linear* transformation. $G = W^T$ consisting of weights $W = (w_1, \ldots, w_K)$. And to simplify the problem, assume the input data has already been *whitened*. $B = \langle xx^T \rangle_t = I \in \mathrm{Mat}_{\mathbb{R}}(N,N)$

$$x \sim \begin{pmatrix} x_1(t) \\ \vdots \\ x_N(t) \end{pmatrix} \in \mathbb{R}^N \times [0,T]$$

$$G : \mathbb{R}^N \to \mathbb{R}^K$$

$$y(t) = G(x(t))$$

$$\min \sum_{i \in [K]} E_i, \qquad E_i = \langle \dot{y}_i^2 \rangle_t$$

$$\text{s.t.} \langle y_i y_j \rangle_t = \delta_{ij}$$

$$\begin{aligned} E_i = \langle \dot{y}_i^2 \rangle_t &= \langle (w_i^T \dot{x})^2 \rangle_t \\ &= w_i^T \langle \dot{x}\dot{x}^T \rangle_t w_i \\ &= w_i^T A w_i \end{aligned}$$

This is an eigenvalue problem for the weights with respect to *A*

$$\delta_{ij} = \langle y_i y_j \rangle_t = w_i^T \langle xx^T \rangle_t w_j = w_i^T w_j$$

So minimise the energy by choosing the *lowest* eigenvectors of *A*

Low-degree-polynomial dimensional reductions

BEFORE

Assume that *G* is a *linear* transformation. $G = W^T$ consisting of weights $W = (w_1, \ldots, w_K)$. And to simplify the problem, assume the input data has already been *whitened*. $B = \langle xx^T \rangle_t = I \in \mathrm{Mat}_{\mathbb{R}}(N, N)$

NOW

Assume that *G* is a *linear* transformation in the space of *low-degree-polynomials* $\mathrm{Poly}_{\deg \leq d}(x_1, \ldots, x_N)$
Now solve the problem as we did before.

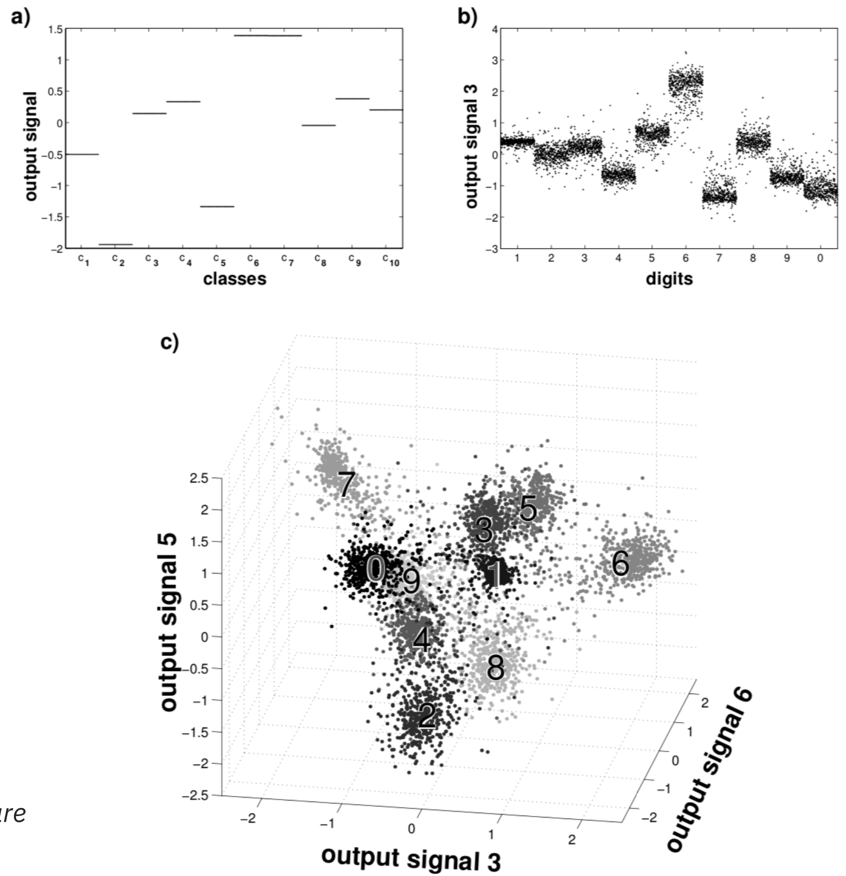
Warning 1: Even for low degree, the dimension of this space is very big. $\sum_{i \in [d]} \binom{N}{d} \sim N^d$

Warning 2: It is even worse for the covariance matrices where are of dimension $\sim N^{2d}$

Part 2: Clustering with SFA

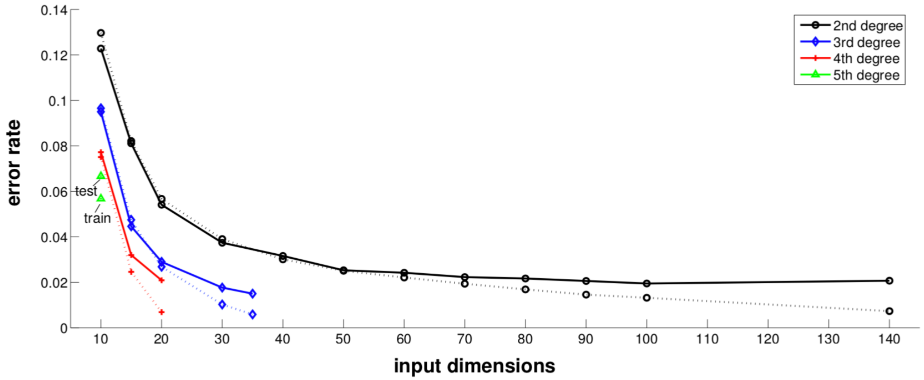
Quantum Machine Learning

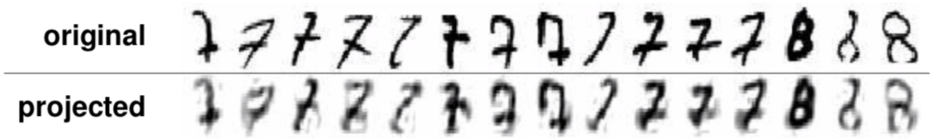
Dimensional Reduction via Slow Feature Analysis



Berkes
Pattern Recognition with Slow Feature Analysis
Cognitive Sciences, 2005

METHOD	% ERRORS
Linear classifier	12.0
K-Nearest-Neighbors	5.0
1000 Radial Basis Functions, linear classifier	3.6
Best Back-Propagation NN (3 layers with 500 and 150 hidden units)	2.95
Reduced Set SVM (5 deg. polynomials)	1.0
LeNet-1 (16×16 input)	1.7
LeNet-5	0.95
Tangent Distance (16×16 input)	1.1
Slow Feature Analysis (3 deg. polynomials, 35 input dim)	**1.5**





The problem statement of SFA Clustering

Given high dimensional data labelled with one of a fixed number of patterns, construct a classifier

P patterns $\{\pi_p\}_{p\in[P]}$ and *n* data points $x \in \mathbb{R}^N$

Minimise: $E = \sum_{k\in[K]} E_k$

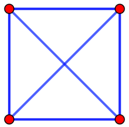
Labelling $\{x(p,i)\}_{i\in[n_p]}$

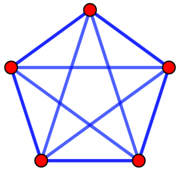
think of $i \in [n_p]$ as $t \in [0,T]$

$$E_k = \frac{1}{Z} \sum_{p\in[P]} \frac{1}{2} \sum_{i,j\in[n_p]} \Big(\big(G(x(p,i)) - G(x(p,j))\big)_k \Big)^2$$

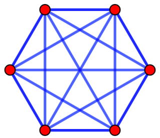
Classifier $G : \mathbb{R}^N \to \mathbb{R}^K$

Lemma: $K = P - 1$





Minimise energy subject to
- (zero mean and) unit variance
- decorrelation



$Z = \sum_{p\in[P]} \binom{n_p}{2}$ = number of pairs of labelled data

The problem statement of SFA Clustering

IBM **Quantum**

Given high dimensional data labelled with one of a fixed number of patterns, construct a classifier

P patterns $\{\pi_p\}_{p\in[P]}$ and *n* data points $x \in \mathbb{R}^N$

Labelling $\{x(p,i)\}_{i\in[n_p]}$

think of $i \in [n_p]$ as $t \in [0,T]$

Classifier $G : \mathbb{R}^N \to \mathbb{R}^K$

Lemma: $K = P - 1$

Minimise energy subject to

- (zero mean and) unit variance
- decorrelation

Minimise: $E = \sum_{k\in[K]} E_k$

$$E_k = \frac{1}{Z} \sum_{p\in[P]} \frac{1}{2} \sum_{i,j\in[n_p]} \big((G(x(p,i)) - G(x(p,j)))_k \big)^2$$

Subject to

$$\frac{1}{n} \sum_{p\in[P]} \sum_{i\in[n_p]} G(x(p,i))_k = 0$$

$$\frac{1}{n} \sum_{p\in[P]} \sum_{i\in[n_p]} G(x(p,i))_k^2 = 1$$

$$\frac{1}{n} \sum_{p\in[P]} \sum_{i\in[n_p]} G(x(p,i))_k G(x(p,i))_\ell = \delta_{k\ell}$$

Eigenvalue problem of SFA Clustering

P patterns
Labelling $\{x(p,i)\}_{i\in[n_p]}$
Classifier $G : \mathbb{R}^N \to \mathbb{R}^K$

Minimise sum of energies:

$$E_k = \frac{1}{Z}\sum_{p\in[P]}\frac{1}{2}\sum_{i,j\in[n_p]}\left((G(x(p,i)) - G(x(p,j)))_k\right)^2$$

Subject to:
zero mean, unit variance,
decorrelation

Assume that *G* is a linear transformation consisting of weights

$$\begin{aligned} E_k &= \frac{1}{2Z}\sum_{p\in[P]}\sum_{i,j\in[n_p]}\left(w_k^T x(p,i) - x(p,j)\right)^2 \\ &= w_k^T\left(\frac{1}{2Z}\sum_{p\in[P]}\sum_{i,j\in[n_p]}\big(x(p,i) - x(p,j)\big)\big(x(p,i) - x(p,j)\big)^T\right)w_k \\ &= w_k^T A w_k \end{aligned}$$

Let's assume the data has been whitened. So the constraint is

$$\delta_{k\ell} = w_k^T B w_\ell \text{ where } B = \frac{1}{n}\sum_{p\in[P]}\sum_{i\in[n_p]} x(p,i)x(p,i)^T = I$$

This is an eigenvalue problem for the weights with respect to *A*

So minimise the energy by choosing the *lowest* eigenvectors of *A*

Scaling considerations of SFA Clustering

P ~ number of patterns Fixed

N ~ input data size Fixed, but can reduce with PCA

K ~ output data size Fixed (*P-1*)

d ~ degree of polynomial expansion Free, effectively increases input size: $N \to \sum_{d' \in [d]} \binom{N}{d'} = \binom{N+d}{d} = O(N^d)$

$n = \sum_{p \in [P]} n_p$ ~ size of test set ...

computational complexity of matrix *A* requiring diagonalisation:

number of entries: $\binom{N+d}{d}^2 = O(N^{2d})$

number of rank 1 entries: $\sum_{p \in [P]} n_p^2 = O(n^2)$

Scaling considerations of SFA Clustering

IBM **Quantum**

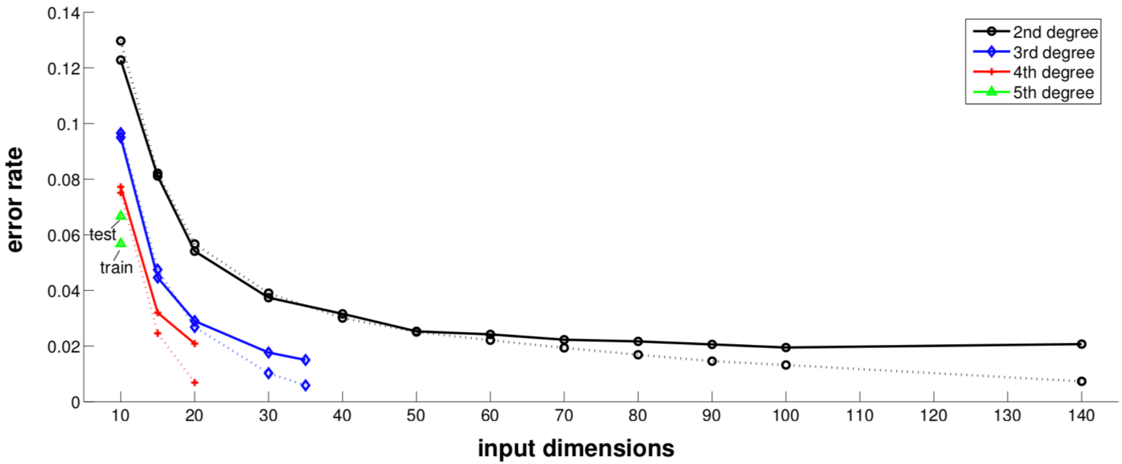
P ~ number of patterns Fixed

N ~ input data size Fixed, (reduce with PCA)

K ~ output data size Fixed

d ~ degree of polynomial expansion Free

$n = \sum_{p\in[P]} n_p$ ~ size of test set ...



computational complexity of matrix A requiring diagonalisation:

number of entries: $\binom{N+d}{d}^2 = O(N^{2d})$

number of rank 1 entries: $\sum_{p\in[P]} n_p^2 = O(n^2)$

Part 3: Clustering with Quantum SFA

Quantum Machine Learning

Dimensional Reduction via Slow Feature Analysis

Quantised SFA Clustering

Goal: Quantise (a part of) the SFA Clustering algorithm to improve run-time.

Some assumptions from our classical setting which we shall keep:
- input data $\{x(p,i)\}_{i\in[n_p]}$ associated with patterns $\{\pi_p\}_{p\in[P]}$
- total of *n* data points in dimension *N*
- *assume this dimension already refers to:*
    - *data has been PCA reduced*
    - *data has been polynomially expanded*
    - *data has been whitened*

Final step: use quantum SVD to obtain classifier $G:\mathbb{R}^N \to \mathbb{R}^{P-1}$

Remember: *G* is matrix weights which are lowest eigenvalues of the covariance matrix associated with the "derivative" of the input vectors.

Kerenidis, Luongo
Quantum classification of the MNIST dataset via Slow Feature Analysis
PRA, 2020 (arXiv, 2018)

Quantised SFA Clustering

Goal: Quantise (a part of) the SFA Clustering algorithm to improve run-time.

Final step: use quantum SVD to obtain classifier $G : \mathbb{R}^N \to \mathbb{R}^{P-1}$

Advantages:
- quantum-speedup from SVD calculation
- if time kept constant, then higher initial PCA and/or higher polynomial expansion may be performed

Challenges:
- data must be in QRAM
- output data is quantum state (which must be projected to one of *P* classes)
- requires knowledge of value of *K-1*, *K* eigenvalues (in order for SVD to project onto lowest eigenvalues)

Ingredients in quantum SVD

IBM **Quantum**

- Phase estimation

  (Kitaev, *Quantum measurements and the Abelian Stabilizer Problem*, Electronic Colloquium on Computational Complexity, 1996)

- Amplitude amplification and estimation

  (Brassar Høyer Mosca Tapp, *Quantum Amplitude amplification and estimation*, Quantum Computation and Information, AMS, 2000)

- SVD

  (Kerenidis Prakash, *Quantum gradient descent for linear systems and least squares*, PRA, 2020 (arXiv 2017))

- Matrix algebra (multiplication, division, projection, composition)

  (qubitisation, block encodings)

  (Chakraborty Gilyén Jeffery, *The power of block encoded matrix powers*, arXiv 2018)

  (Gilyén Su Low Wiebe, *Quantum singular value transformation and beyond*, arXiv 2018)

Part 4: Is that it!?

Quantum Machine Learning

Dimensional Reduction via other techniques

IBM **Quantum**

Other dimensional reduction techniques

Quantum Principle Component Analysis

Lloyd, Mohseni, Rebentrost

Nature Physics 2014

Other dimensional reduction techniques

Quantum Discriminant Analysis for Dimensionality Reduction and Classification

Cong, Duan

New Journal of Physics 2016

Other dimensional reduction techniques

Quantum Algorithms for Topological and Geometric Analysis of Data

Lloyd, Garnerone, Zanardi

Nature Communications 2016

Review of a Quantum Algorithm for Betti Numbers

Gunn, Kornerup

arXiv 1906.07673

Other dimensional reduction techniques

Variational Quantum Algorithms for Dimensionality Reduction and Classification

Liang, Shen, Li, Li

PRA 2020

Variational Quantum Singular Value Decomposition

Wang, Song, Wang

arXiv 2006.02336

Other dimensional reduction techniques

Limitations on quantum dimensionality reduction

Harrow, Montanaro, Short

International Colloquium on Automata, Languages, and Programming 2011

IBM **Quantum**