Master's Thesis

Quantum-Inspired Forest: A Physical Perspective on Machine Learning

（量子に触発されたフォレスト: 機械学習に関する物理的な視点）

August, 2017

XIE Zeke
謝 沢柯

Master of Engineering
Department of Electrical Engineering and Information Systems
Graduate School of Engineering
The University of Tokyo

Supervisor
Prof. HIRAKAWA Kazuhiko
Co-Supervisor
Prof. SATO Issei

Abstract

Machine learning has gained in popularity over recent years. It shows great power in many real-world problems, including Computer Vision, Natural Language Processing and Speech Recognition. However, a clear theoretical mechanism is still lacked behind many machine learning algorithms, such as Deep Learning. Good theoretical analysis can not only explain the current success of machine learning, but also reveal new approaches to future advancements.

In last few years, more and more physicists contributed good works that provided novel insights and philosophy for machine learning communities. Due to two reasons, we believe physical approaches can be very promising. First, we think machine learning share some similar theoretical mechanism with physical theories, particularly quantum physics and statistical physics. We may employ physical languages to study machine learning models in some physical frameworks. Second, real-world problems and information itself must obey some fundamental laws, which are often physical laws for our universe. Good physical prior knowledge which may be helpful to reduce model complexity and improve generalization. The physical perspective deserve more attention.

In this thesis, we review a few related works on the physics based approach and related traditional machine learning. It helps us understand how physics may interact with machine learning. We propose quantum interpretations for machine learning and a class of original physics-inspired machine learning algorithms, named Quantum-Inspired Forest. Both theoretical proof and empirical analysis are presented in details. The success of quantum-inspired machine learning encourages us to pay attention to multiple physical viewpoints. Even without solid theoretical analysis, important prior knowledge and heuristics may be obtained from physical models. The proposed algorithm is an example that illustrates how to combine physics and machine learning together. We in particular discuss a physical perspective on machine learning. We hope our physical analysis on machine learning would inspire more novel works in near future.

Keyword: Machine Learning, Ensemble Learning, Quantum Physics, Statistical Physics

Acknowledgements

I would like to thank my supervisor Professor Kazuhiko Hirakawa. I received your unselfish and kind help when I faced serious trouble and challenge. I want you to know that it means a lot for me in my deep heart. It's like a follower in the desert. The environment is dying dried, but the follower still flourishes.

I would also like to thank my co-supervisor Professor Issei Sato. You gave me invaluable supervision on my research. I would never find the research I love so much without your guidance.

I also thank all people who contribute to my growing up. It's all of you make me unique.

In addition, I am especially grateful to my parents in China. I always know you miss me as much as I miss you. I always know you would support me until the ending of the world. I always know your sacrifice for my growing up these years. I hope to become a man that makes you feel proud.

I enjoy every rain drop in the summer.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Machine learning not only has gained in popularity over recent years, but also has fundamentally changed the way humans interact with data. Benefitted from increasing data size and computational resources, a lot of promising works have emerged in recent years. On the one hand, a lot of works have been applied on real-world problems. Some of works even outperform humans in specific tasks. For example, [Taigman et al., 2014] first achieved human-level face recognition; [Mnih et al., 2015] performed human-level control through deep reinforcement learning; AlphaGo beat the famous Go world champion Lee Sedol in 2016 [Silver et al., 2016]. On the other hand, these advancements also have deepen our understanding about data and intelligence. Machine learning is a science that employ "experiences" to improve learning systems based on computation. And data is exactly the experience for computers. Intelligent life like human can learn laws of nature from rich experience, and take action based on the laws. And a good machine learning system also have similar characteristics. Machine learning systems possesses the ability to learn models from data. In our current understanding, the set of abilities is corresponding to machine learning algorithms. Driven by big data and large computational resources, researchers have proposed various machine learning algorithms. According to different tasks or "actions", we have different types of machine learning algorithms, mainly including supervised learning and unsupervised learning. Both regression and classification are supervised learning, that needs to learn a mapping from inputs $\vec{x}$ to target variables $\vec{y}$. Regression has real-valued target variables, while classification categorical target variables. Unsupervised learning points a class of learning that tries to learn "interesting patterns" in the data without any target variables.

Applications of machine learning range from Computer Vision to Natural Language Processing. However, a clear theoretical mechanism is still lacked behind many machine learning algorithms, such as Deep Learning[LeCun et al., 2015]. It is not strange that Deep Neural Networks (DNN) have strong representation power with so many model parameters. But it is very hard to understand the unreasonably wonderful generalization ability of DNN. Good theoretical analysis can not only explain the current success of machine learning, but also reveal new approaches to future advancements.

In last few years, more and more physicists contributed good works that provided novel insights and philosophy different from traditional machine learning communities. Due to two reasons, we believe physics-inspired machine learning can be very promising. First, we think machine learning share some similar theoretical mechanism with physical theories, particularly quantum physics and statistical physics. We may employ physical languages to study machine learning models in some physical framework. Second, real-world problems must obey some fundamental laws, which are exactly physical laws for our universe. Current machine learning approach almost explores all mathematically possible hypothesis space, but ignore important physical prior knowledge. Physical prior knowledge may be helpful to reduce model complexity and improve model generalization abilities. We consider theoretical connections and physical prior knowledge as two main way to achieve physics-inspired machine learning.

And among various physics-inspired approaches, the quantum theoretical approach has attracted most attention, while the statistical physical approach is another one. One reason is that the natural laws are quantum mechanical at the scales of modern information processing technology, while the more familiar classical physics dominates at the human scale. And information itself not only has natural connections with statistical physics from a information theoretical viewpoint, but also plays a core role in quantum mechanics as quantum information. Quantum theory provides us new tools and insights toward information. Quantum machine learning and quantum-inspired machine learning are two confusing concepts. Generally speaking, Quantum machine learning refers to machine learning algorithms that need be implemented on quantum machines. The common quantum machines are general-purpose quantum computers that perform real quantum computing using qubits. But a quantum machine doesn't have to be a general-purpose quantum computer. Actually any machines that employ some quantum effects to perform

computing beyond classical computing may be called quantum machines [Boixo et al., 2014]. In last decade, a lot of works about quantum machine learning have been reported [Schuld et al., 2015, Wittek, 2014]. Quantum machine learning has become a promising interdisciplinary research field. Quantum-inspired machine learning differs from Quantum Machine Learning in several aspects. quantum-inspired machine learning means machine learning algorithms that involve in some quantum theoretical elements but don't require a quantum machine for implementing it. Quantum physics and machine learning can be deeply interconnected in theoretical analysis. Several works of algorithms utilizing Quantum physics are introduced in Chapter 2 and Chapter 5.

The success of quantum-inspired machine learning encourages us to pay attention multiple physical viewpoints. Good theoretical works may deeply inovate machine learning. But even without solid theoretical analysis, important prior knowledge and heuristics may be obtained from physical models. We present several interesting examples that illustrates how to combine physics models and machine learning models together. We hope these physical analysis would inspired more novel works in near future.

1.1 Thesis Overview

The thesis is organized as follows. In Chapter 2, we review a few recent works on ensemble learning and quantum-inspired machine learning. In particular, we also introduce necessary elements of quantum physics and take the famous Quantum Annealing as an example for quantum-inspired models. In Chapter 3, we propose quantum interpretations for machine learning and a class of original physics-inspired machine learning algorithms, named Quantum-Inspired Regression Forest. Both theoretical proof and empirical analysis are presented in details. In Chapter 4, we further develop Quantum-Inspired Classification Forest inspired by similar heuristics. A empirical analysis is studied. Chapter 5 is a relatively isolated chapter but provide highly rich knowledge and insights. No specific algorithm is presented. The meaning lies in the physical perspective on machine learning. Chapter 6 is a summary of the thesis. We present the main conclusions in this chapter.

Chapter 2

Background

2.1 Ensemble Learning

The goal of ensemble learning [Zhou, 2012] is to combine the predictions of multiple base learners to get more accurate aggregate predictions. Ensemble learning algorithms frequently rank top in many data mining competitions, and consistently outperform single learners, such as Support Vector Machines [Cortes and Vapnik, 1995]. This approach has proven to be a powerful method in practical applications, especially for those general-purpose tasks. The ensemble method generally is favored in terms of increasing robustness and accuracy. Since the theoretical analysis of ensemble models, particularly tree ensembles, has been carefully studied, we are able to theoretically analyze novel ensemble algorithms besides the empirical analysis. Many researchers have contributed to a significant amount of good works in last decades.

Researchers have known that ensemble diversity and the accuracy of base learners are two main factors deciding the performance of ensemble models [Zhou, 2012]. We usually inject randomness into ensemble models aiming at generating diversified base learners and ensemble strategies. Unfortunately, the randomness approach generally reduced the accuracy of base learners. It's not surprising that randomness may lead some slight deviation from optimal base learners. Researchers find it quite difficult to improve ensemble diversity without damaging the accuracy of base learners. How to deal with the trade-off between diversity and accuracy becomes one of core challenges in ensemble learning.

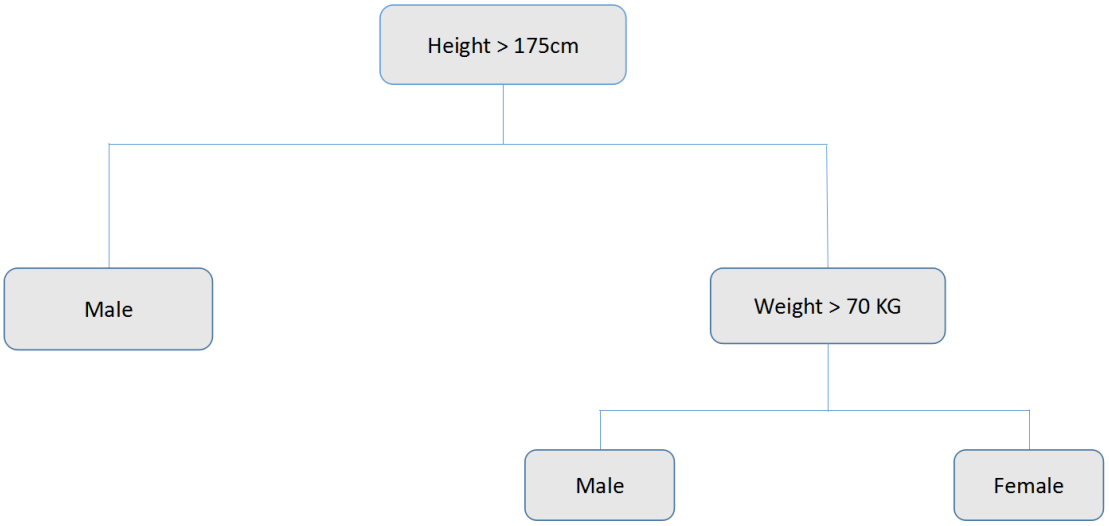

FIGURE 2.1: The structure of CART

2.1.1 Tree and Forest

Decision Tree is an important type of algorithm for predictive modeling machine learning. It has a more modern name, Classification And Regression Tree (CART) [Breiman, 1984]. The classical decision tree algorithms have been studied for decades and modern tree ensembles like Random Forest [Breiman, 2001] are among the most powerful techniques available.

Decision tree consists of a set of tree-structured decision tests working in a divide-and-conquer way. The tree are obtained by recursively partitioning the data space and fitting another simple tree model within each partition. The recursively constructed model finally has a typical tree structure, seen in 2.1. Decision trees used in data mining are of two main types. Classification tree is when the predicted variable is the class or category. Regression tree analysis is when the predicted predicted variable can be considered a real number, such as the price of stocks.

Algorithms for constructing decision trees usually work top-down. A typical tree chooses a variable at each step that best splits the set of items. And we may use different metrics for measuring the quality of the split according to requirements of tasks. The key part of a decision tree algorithm is exactly how to select the splits properly. Selection strategies that can balance accuracy and diversity well is usually considered as excellent ones.

A key advantage of the tree structure is its scalable applicability to any high-dimensional data. The main difference among different kinds of Decision Trees

Algorithm 1: Basic Steps for Decision Tree Construction

1 Start at the root node.
2 For each $\vec{x_k}$, find the set F_k that minimizes the sum of the node impurities in the two child nodes and choose the split $x^\star \in F^\star$ that gives the minimum overall $\vec{x_k}$ and F_k.
3 If a stopping criterion is reached, exit. Otherwise, apply step 2 to each child node in a turn.

lies in the measure of node impurities. For regression trees, mean square errors is used for measuring information gain. The first published classification tree THAID [Messenger and Mandell, 1972] measures node impurities based on the distribution of the target variables in the node. In more modern tree algorithms, two criterions are commonly used for the information gain in splitting. One is Gini index, the other is entropy. They are defined as follows, respectively,

$$Gini(E) = 1 - \sum_{j=1}^{c} p_j^2, \tag{2.1}$$

$$H(E) = -\sum_{j=1}^{c} p_j \log p_j. \tag{2.2}$$

Gini measurement is the probability of a random sample being classified correctly if we randomly pick a label according to the distribution in a branch. Entropy is a measurement of information that measures how you reduce the uncertainty about the label. However, in practice both Gini index and Entropy typically perform pretty much same. It is often not worth spending much time on evaluating trees using different impurity measurements in practice.

Decision Trees are very fast but weak machine learning algorithms. We usually combine the predictions of multiple trees to get more accurate aggregate predictions in practice. And the tree ensemble methods are also called forests. Researchers have proposed multiple good ensemble algorithms. One of most classical ensemble method is Bagging (Bootstrap AGGregatING) [Breiman, 1996], whose two key ingredients are bootstrap and aggregation. Given a standard training set D of size n, Bagging generates T new training sets D_i, each of size n^{prime}, by sampling from D uniformly and with replacement. By sampling with replacement, some samples may be repeated in each D_i. Statistically, a sample is likely to appear at least once in the sample with a probability of 64%. Random Subspace [Ho, 1998] is another classical ensemble method, also named Attribute Bugging

or Feature Bagging. The random subspace method is similar to bagging except that the features are randomly sampled, with or without replacement, for base each learner. In each pass, we randomly choose some dimensions from the given feature space, and all samples are projected to this subspace. And a decision tree is constructed using the projected training samples. Finally, we obtain many trees constructed in randomly chosen subspaces.

Algorithm 2: Random Forest

```
1 function RandomForest (S, F, T)
  Input  : A training set S = (x^1, y^1), ..., (x^n, y^n), features F, and the forest T
  Output: Random Forest H
2 H ← ∅
3 for i ← 1 to T do
4     S^i ← a bootstrap sample from S
5     F^i ← a subset from F
6     h_i ← TreeLearn(S^i, F^i)
7     H ← H ∪ {h_i}
8 return H
```

Random Forest [Breiman, 2001] is a representative of the state-of-the-art tree ensemble algorithm. Random Forest combines Bagging and Random Subspace together, and then improve the performance significantly. Random Forest has several desirable characteristics as Breiman describes:

(1) Its accuracy is as good as Adaboost and sometimes better.

(2) It's relatively robust to outliers and noise.

(3) It's faster than bagging or boosting.

(4) It gives useful internal estimates of error, strength, correlation and variable importance.

(5) It's simple and easily parallelized.

These characteristics make Random Forest a powerful and fast algorithm widely used in many data competitions. We can find recent enhancements of Random Forest in [Fawagreh et al., 2014], including perfect random tree ensembles [Cutler and Zhao, 2001], extremely random trees [Geurts et al., 2006], and completely random decision trees [Fan et al., 2006, Liu et al., 2005]. Improving strength of individual trees and decreasing the correlation between trees are main factors in reducing the Random Forest error rate [Breiman, 2001]. It means we need strong trees, but the trees need become strong in different ways. Our proposed Quantum-Inspired Forest aims at improve individual accuracies and decreasing the correlation between trees at same time.

2.2 Quantum Mechanics

Quantum mechanics, also called quantum theory, usually refers to a fundamental physical theory of nature at small scales and low energy levels of atoms and subatomic particles. And quantum mechanics further led us to discover the field of quantum computation and quantum information [Nielsen and Chuang, 2010]. This forms a bridge between physical world and information, which many people thought are quite distinct concepts before.

Quantum computing is of course an important bridge between quantum mechanics and machine learning. In last decade, it has become a promising interdisciplinary research filed, named quantum machine learning. Generally speaking, quantum machine learning refers to machine learning algorithms that need be implemented on quantum machines. The common quantum machines are general-purpose quantum computers that perform real quantum computing using qubits. But a quantum machine doesn't have to be a general-purpose quantum computer. Actually any machines that employ some quantum effects to perform computing beyond classical computing may be called quantum machines [Boixo et al., 2014]. In last decade, a lot of works about quantum machine learning have been reported [Schuld et al., 2015, Wittek, 2014]. We discuss this topic, particularly quantum annealers, in 2.3.

In our opinion, quantum mechanics is more than a kind of physical theory. It has a deeper meaning than its physical part. Quantum mechanics actually is a mathematical framework that can used for construction of physical theories. Some advanced physical theories, such as quantum electrodynamics, is built on the framework of quantum theory, buy they also have elements not determined by quantum theory itself. They are developed within the mathematical framework, but involve in new rules. This is the philosophy that we could also apply to machine learning.

Specifically, quantum theory is naturally a theory of probability [Rédei and Summers, 2007] due to the probabilistic essence of nature. The mathematical framework of quantum mechanics provides a set of useful tools for probabilistic descriptions which we could apply to general fields other than physics. This is a approach that may connect quantum mechanics and machine learning directly. How can quantum probability theory benefit machine learning is an interesting question.

Might quantum theory revolutionize the whole probabilistic framework of machine learning like how it revolutionized physics last century? Some researchers have proposed their thoughts. For example, [Melucci and van Rijsbergen, 2011] proposed a quantum theoretical framework for information retrieval.

2.2.1 Density Matrices

The density matrix is the main tool we apply in Quantum-Inspired Forest. The formalism of the density matrix or operators was first introduced by [Von Neumann, 1927] independently. [Nielsen and Chuang, 2010] Section 2.7 discusses density matrix and operators in detail. We briefly introduce the density matrix here. A density matrix are a matrix that describes quantum systems in a mixed state, an ensemble of several pure states. In quantum mechanics, physicists often denote a pure state as a state vector $|\psi\rangle$. However, there exist mixed states, which cannot be written as a state vector. A mixed quantum state corresponds to a probabilistic mixture of pure states, also called a quantum ensemble. [Von Neumann, 1927] proposed the powerful tool that can describe both pure states and mixed states well in one frame. We show how to introduce density matrix and quantum operators into machine learning in this section.

2.3 Quantum Annealing

Quantum Annealing(QA) is an optimization method that employs quantum effects to optimize the cost function. QA escapes local minima by quantum tunneling through barriers separating local minima. QA can be implemented by Quantum Monte Carlo Simulation Method on classical computers, which is a well discussed topic. Quantum devices that implement QA directly have also become another important direction. As quantum devices that employs quantum effects have the potential of quantum speedup, quantum annealers like D-wave Systems have attracted most attention recently. However, the quantum speedup of Quantum Annealing is still an open question. As a famous algorithm inspired by quantum mechanics, QA has showed favorable performance in previous works. QA is a very good example showing how we could employ physical models to invent new algorithms. As QA plays an important role in both quantum and quantum-inspired

machine learning, we carefully introduce its background, principle, and comparison with classical simulated annealing in this section.

2.3.1 Background of Quantum Annealing

We call minimizing or maximizing an objective function as optimization problems. Optimization problems are common and important in many fields, including machine learning. Training a machine learning model requires minimizing its cost function. Researchers have proposed the convex optimization method for most optimization problems, but a class of nonconvex optimization problems, such as Combinatorial Optimization, are still too hard to solve efficiently. Finding the ground state of a Ising spin glasses is a typical combinatorial optimization problem in quantum physics.

Simulated annealing (SA), another physics-inspired algorithm, was first proposed as a general probabilistic method for optimization problems in [Kirkpatrick et al., 1983]. Actually it is said thermal annealing is likely to the oldest optimization method in human history. By first heating metal and let it cool down slowly, we can make the metal materials relieve internal stresses and reach a low energy state. Simulated annealing is easy to implement and effective in solving most combinatorial problems. Simulated annealing, also known as classical annealing (CA), simulates thermal annealing process to allow the system to escape from local minima of the cost function so that the system reaches the global minimum under an appropriate annealing schedule, controlled by the rate of decrease of temperature.

Researchers have widely applied SA to various problems, particularly those nonconvex optimization problems, even if SA lacks a theoretically guaranteed speed of convergence. Comprehensive studies [Johnson et al., 1992, 1989, 1991] discuss the application of simulated annealing to four problems: the traveling sales man(TSP), graph partitioning problem (GPP), graph coloring problem (GCP)and number partitioning problem (NPP). Other typical problems include job scheduling, circuit minimization, and chain optimization which are all non-deterministic polynomially (NP) hard problems. Bohachevsky and his partners proposed a generalized simulated annealing framework for continuous optimization problems [Bohachevsky et al., 1986]. Overall, simulated annealing is a generally applicable, efficient, and easy-to-implement probabilistic approximation algorithm to produce

good solutions for combinatorial optimization problems. And it requires little domain knowledge on problems and cost functions.

Quantum annealing is a quantum-mechanical paradigm to solve combinatorial optimization problems and was proposed by Kadowaki and Nishimori in 1998 [Kadowaki and Nishimori, 1998]. Some other pioneering works also made important contributions, theoretically by [Amara and Kuhar, 1993, Farhi et al., 2001, Finnila et al., 1994, Santoro et al., 2002], experimentally by [Brooke et al., 1999]. Quantum annealing introduces quantum fluctuations into annealing process of optimization problems, aiming at faster convergence to the optimal state. Quantum annealing can be simulated on conventional computers using Quantum Monte Carlo Method. But the most natural way of implementing QA is directly using a quantum mechanical system. Such quantum mechanical systems are also called quantum annealers. D-wave System is the first commercial quantum annealer. Recently, researchers have made a few experimental progress on D-Wave Systems.

2.3.2 Principle of Quantum Annealing

In this subsection, we introduce the fundamental principle of quantum annealing [Das and Chakrabarti, 2008]. We can see the basic difference of principles between simulated thermal annealing and quantum annealing in 2.2.

We first consider the simple Ising model with transverse fields:

$$H(t) = -\sum_{ij} J_{ij} S_i^z S_j^z - \Gamma(t) \sum_i S_i^x \tag{2.3}$$

$$= H_c + \Gamma(t) H_{kin} \tag{2.4}$$

The artificial quantum kinetic term $\Gamma(t)H_{kin}$ causes quantum tunneling between various classical states. And we could let the parameter Γ decrease gradually from a large value to zero, the Hamiltonian evolves into the optimal state, namely the ground state of H_0. Initially Γ is kept high so that the quantum fluctuations term H_{kin} dominates and the ground state is trivially a uniform superposition of all the classical configurations. And then we decrease Γ following some annealing schedule. If we let the annealing schedule be slow enough, the evolving system will always remain at the instantaneous ground state assured by the adiabatic theorem of the quantum mechanics [Sarandy et al., 2004]. The system will be

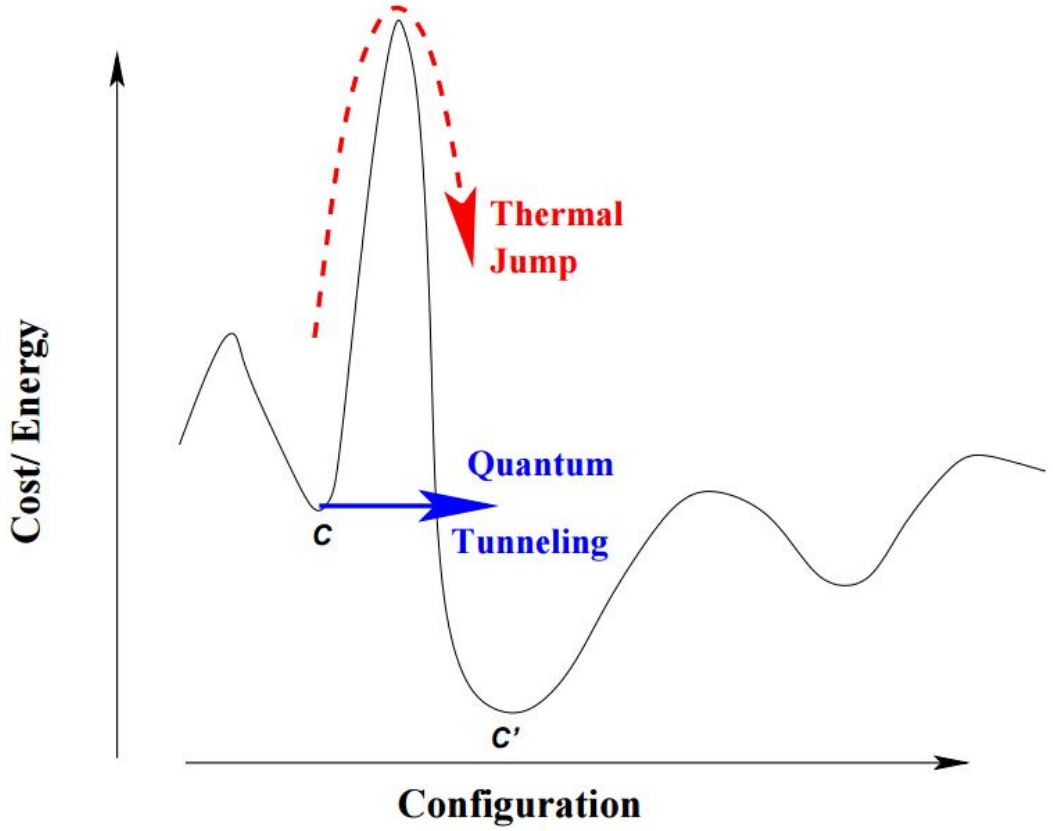

FIGURE 2.2: While optimizing the cost function of a computationally hard problem (like the ground state energy of a spin glass or the minimum travel distance for a traveling salesman problem), one has to get out of a shallower local minimum like the configuration C (spin configuration or travel route), to reach a deeper minimum C'. This requires jumps or tunneling like fluctuations in the dynamics. Classically one has to jump over the energy or the cost barriers separating them, while quantum mechanically one can tunnel through the same. If the barrier is high enough, thermal jump becomes very difficult. However, if the barrier is narrow enough, quantum tunneling often becomes quite easy. [Das and Chakrabarti, 2008]

found in the ground state of the original classical Hamiltonian H_c as desired when Γ finally decreases to zero. Although some arguments still exit, most researchers believe strictly adiabatic and quasi-stationary quantum annealing is equivalent to Quantum Adiabatic Evolution [Farhi et al., 2001].

Three more important questions worthy be discussed are how to decide annealing schedule in order to assure quantum adiabaticity, how to choose an appropriate H_{kin} and how to prepare the classical Hamiltonian H_c.

There are a few common choices of function Γ, such as $\Gamma(t) = 1 - \frac{t}{T}$. But for assuring adiabatic evolution, according to the adiabatic theorem of quantum mechanics,

the evolution time τ must satisfy the following condition

$$\tau \gg \frac{|\langle H\rangle|_{max}}{\Delta^2_{min}} \tag{2.5}$$

, where

$$|\langle H\rangle|_{max} = \max_{0\leq t\leq\tau}[|\langle\phi_0(t)|\frac{\mathrm{d}H}{\mathrm{d}s}|\phi_1(t)\rangle|] \tag{2.6}$$

$$\Delta^2_{min} = \min_{0\leq t\leq\tau}[\delta^2(t)]; s = \frac{t}{\tau}; 0 \leq t \leq \tau \tag{2.7}$$

, $\phi_0(t)$ and $\phi_1(t)$ being respectively the instantaneous ground state and the first excited state of the total Hamiltonian H, and $\Delta(t)$ the instantaneous gap between the ground state and the first exited state energies [Sarandy et al., 2004]. QA may work out very well for any finite systems, as the gap Δ_{min} is very unlikely vanish for a random system.

Quantum annealing has a flexibility that classical annealing does not have. Quantum annealing may choose an appropriate quantum kinetic term, which may bring significant improvements. Morita and Nishimori demonstrated this flexibility well for QA of random field Ising model by introducing a ferro-magnetic transverse field interaction, in addition to the conventional single-spin-flip transverse field term [Morita and Nishimori, 2007]. If a ferromagnetic transverse term of the form:

$$H^{(2)}_{kin} = -\Gamma(t)\sum_{ij} S^x_i S^x_j \tag{2.8}$$

is added to the original Hamiltonian H, an considerable improvement of QA is observed. Figure 2.3 shows how a quantum kinetic term may increase the energy gap. The improvements are observed because the ferromagnetic transverse field term effectively increases the gap Δ between the ground state and the first excited state and thus decreases the characteristic timescale for the system. This example illustrates how one can utilize the flexibility in choosing the kinetic term in QA to formulate faster algorithms. This also reminds us how the knowledge of the phase diagram of the system, such as the position of the quantum critical point in particular (where the gap tends to vanish), helps us choose appropriate additional kinetic terms so that the annealing paths that can avoid the regions of very low gap at least to some extent.

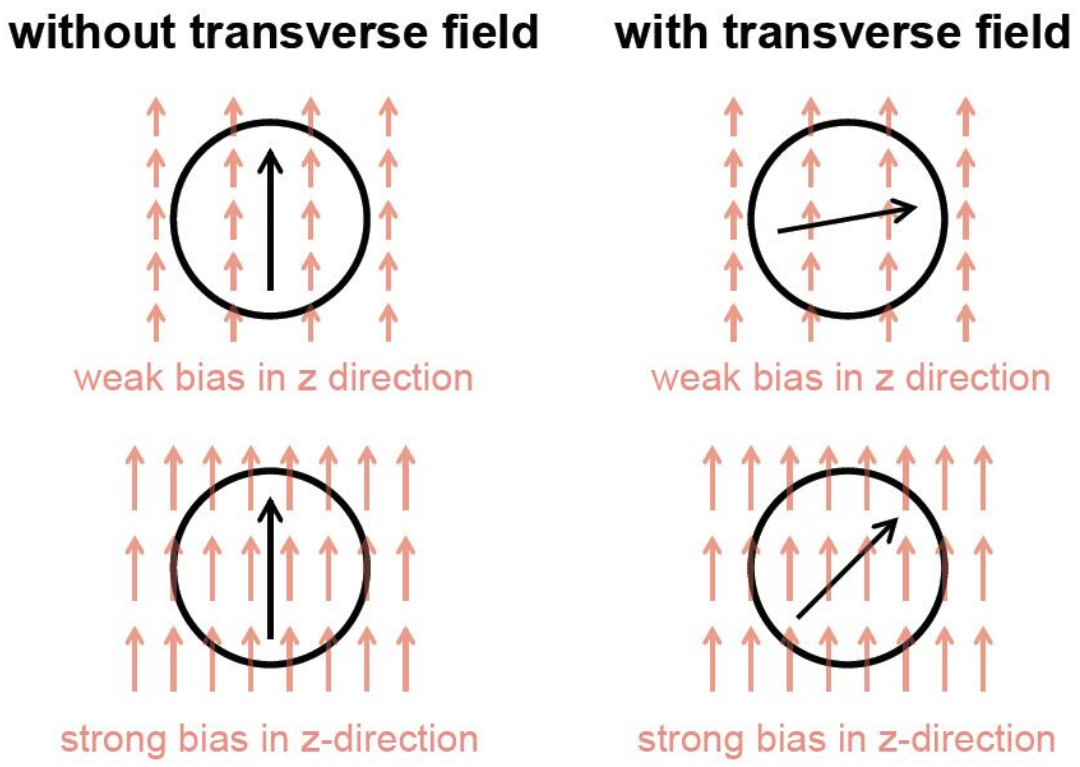

FIGURE 2.3: The transverse field as the kinetic term effectively increases the energy gap Δ between the ground state and the first excited state[Shin et al., 2014]

2.3.3 Quantum Annealing vs. Classical Annealing

As we mention above, simulated annealing is a powerful algorithm for many NP-hard combinatorial optimization problems. A consequence of NP-hardness is that any method to efficiently solve Ising spin problem would stand as an efficient method of solving other important problems. We mainly discuss recent reports [Heim et al., 2015] on the performance of quantum annealing and classical annealing on Ising spin glass problems in this section. In particular, we study the Ising spin glasses with N spins whose Hamiltonian is written as:

$$H_c(t) = -\sum_{ij} J_{ij} S_i S_j - \sum_i h_i S_i \tag{2.9}$$

where S_i takes the values ± 1 and represents the orientation of the spin on the lattice site i. The couplings between spin i and j are denoted by J_{ij} and h_i are local fields.

Researchers demonstrated that simulated annealing using the Metropolis algorithms is a powerful algorithm to minimize H_c. The initial thermal excitations allow the system to escape from a suboptimal and relax into a low energy state with energy close to that of the ground state. The principle behind this algorithm is same as that behind metal materials reach a low energy state after thermal annealing. We will refer to the difference between the final energy and the energy of the ground state as the residual energy $E_{res} = E - E_0$.

As we discussed above, quantum annealing that employs quantum tunneling instead of thermal annealing can be advantageous for cost functions with narrow but tall barriers, which are easier to tunneling through than to thermally climb over. To perform quantum annealing of Ising spin glasses, we add a non-commuting kinetic term as usual:

$$\begin{aligned} H_{tot}(t) &= -\sum_{ij} J_{ij} S_i^z S_j^z - \sum_i h_i S_i^z - \Gamma(t) \sum_i S_i^x \\ &= H_c + \Gamma(t) H_{kin} \end{aligned}$$

The transverse field term $\Gamma(t)$ shall decrease from a large value to zero as usual.

Although the classical simulation of quantum annealing grows exponentially with system size, QA can be efficiently implemented using Path Integra Monte Carlo (PIMC). We call it simulated quantum annealing (SQA) in this section.

The performance comparison of SQA and SA gives the strongest evidence for quantum annealing superpior to classical annealing for Ising spin glasses [Heim et al., 2015, Martoňák et al., 2002]. In Figure 2.4, researchers show best results of 32 independent SA simulations, whose computational cost is roughly same as the computation cost of SQA simulation. There, DT-SQA stands for a discrete time SQA simulation, and CT-SQA stands for a continuous time SQA simulation. DT-SQA is performed with a finite number of time slices M and a corresponding non-zero time step $\Delta_\tau = \frac{\beta}{M} = 1$, which brings time discretization errors of order $O(\frac{\beta^3}{M^2})$. Please refer to [Heim et al., 2015] for more details of simulation algorithms and annealing schedules. Overall, we can see that, in either case, the performance of PIMC-based SQA is superior to that of SA.

The reason behind the advantage is also interesting [Heim et al., 2015]. Bettina Heim and his co-authors believe that the advantage observed for PIMC compared

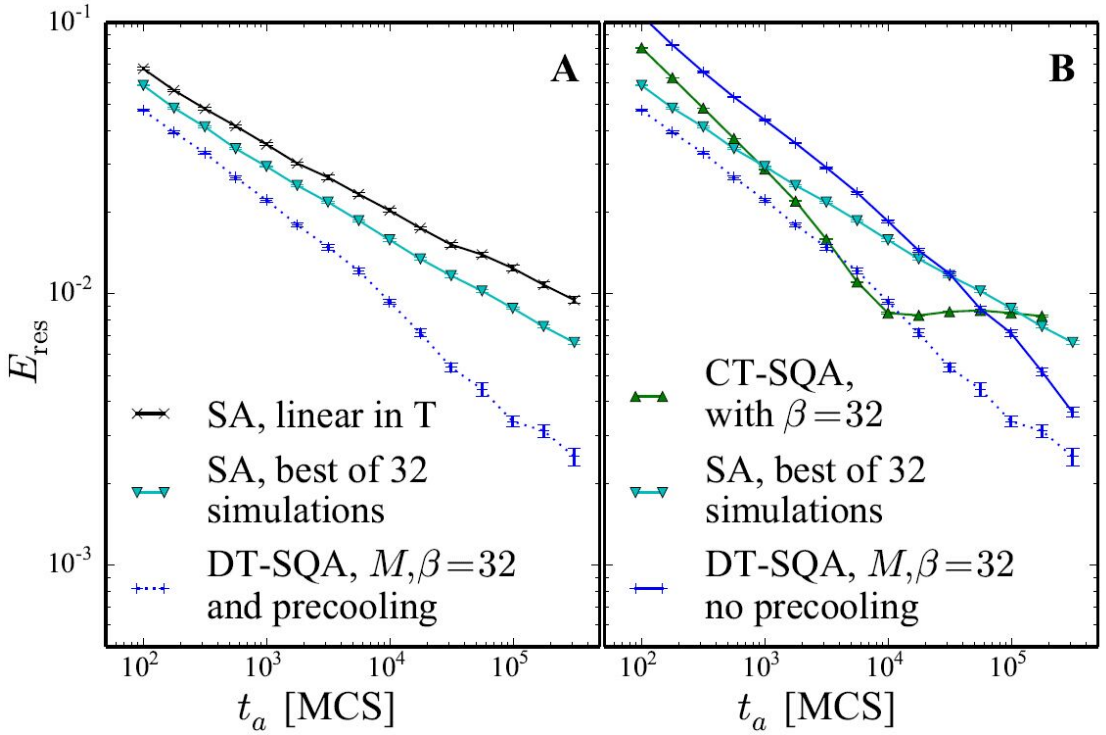

FIGURE 2.4: Decrease of the residual energy E_{res} for SA, DT-SQA (panel A) and CT-SQA (panel B) as a function of the annealing time t_a [in units of Monte Carlo steps (MCS)] for the square lattice Ising spin glass instance of [Santoro et al., 2002] with 6400 spins and uniformly distributed couplings in $(-2, 2)$. The plotted value of Eres and error bars are obtained by averaging over forty annealing runs. [Heim et al., 2015]

to classical annealing is due to the use of large imaginary times steps in the path integral. When we take the physical limit of continuous time and measuring the average energy, the advantage tends to vanish. We note that the continuous time SQA is the one more like a real quantum annealing using quantum machines.

This experimental result shows that upon increasing the annealing time SQA minimizes the residual energy faster than SA. It has truly indicated some quantum advantage. But, in contrast to quantum advantage of SQA over SA seen, recent studies of D-wave Systems still failed to prove solid quantum speedup, while the evolution of hardware is consistent with that of a quantum annealer.

It is fair to stress that it is a priori but not obvious or guaranteed that a QA approach should outperform a CA on a given problem. The comparative performance of QA and CA reply on the energy landscape of the problem at hand, in particular on the type of barriers separating the different local minima, and the kinetic term, which also plays a crucially important role. Unfortunately, when dealing with practical problems, we usually know little about the energy landscape

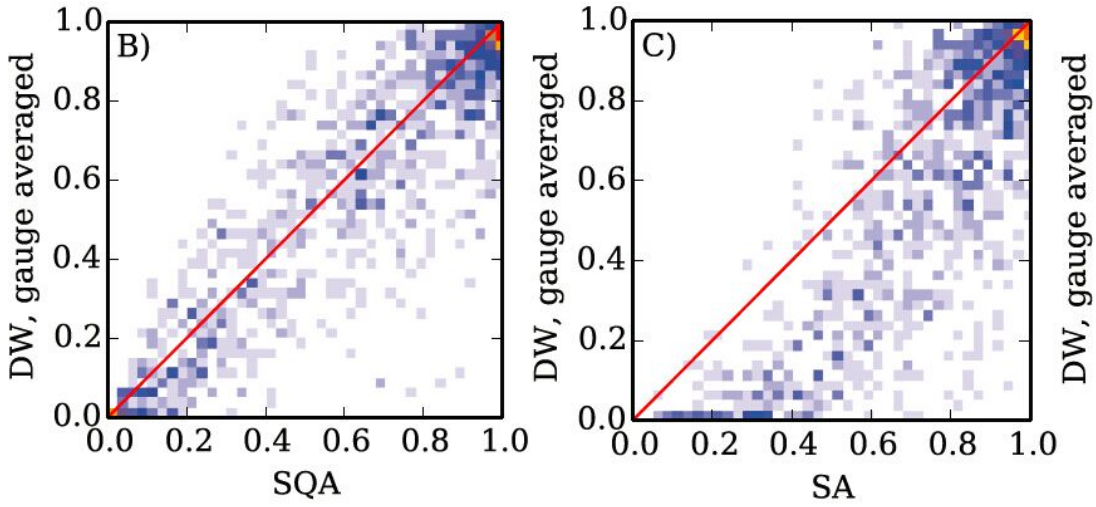

FIGURE 2.5: Scatter plot of success probabilities. The correlation between D-Wave and SQA is noticeably better than that between D-Wave and the classical models.[Boixo et al., 2014]

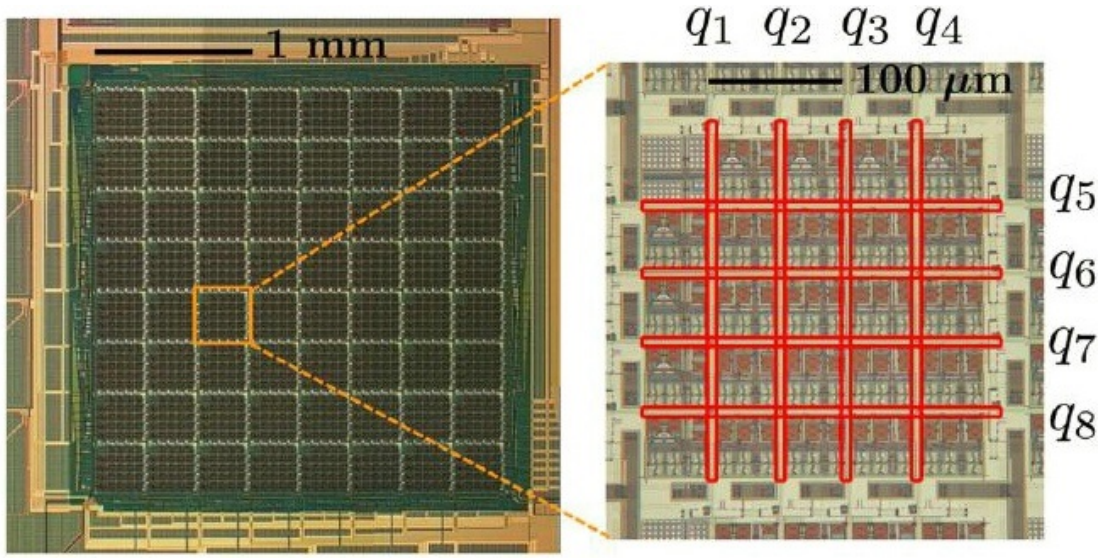

FIGURE 2.6: Photograph of the QA processor. Measurements performed on the eight-qubit unit cell indicated. The bodies of the qubits are extended loops of Nb wiring (highlighted with red rectangles). Inter-qubit couplers are located at the intersections of the qubit bodies. [Lanting et al., 2014]

of cost function. And there is still no reliable theory predicting the performance of a QA algorithm, in particular correlating it with the energy landscape of the given optimization problem.

2.4 Quantum-Inspired Machine Learning

We briefly review some recent works about quantum-inspired machine learning. Readers may compare their quantum-inspired approaches with ours. Quantum machine learning and quantum-inspired machine learning are two confusing concepts. Generally speaking, quantum machine learning refers to machine learning algorithms that need be implemented on quantum machines. Quantum-inspired machine learning differs from quantum machine learning in several aspects. Quantum-Inspired Machine Learning means machine learning algorithms that involve in some quantum theoretical elements but don't require a quantum machine for implementing it. Quantum physics and machine learning can be deeply interconnected in theoretical analysis.

In recent years, multiple quantum-inspired machine learning algorithms have been proposed. [Wolf, 2006] reported Learning using the Born Rule. In Quantum Mechanics, Born rule states that the probability of an outcome $|a\rangle$ given a state $|\Phi\rangle$ is the square of their inner products $\langle a|\Psi\rangle$. It provides a new tool for probabilistic descriptions different from the Bayesian rule. Lior Wolf unraveled a new probabilistic justification for popular algebraic algorithms, based on the Born rule. Lior Wolf discussed several algorithms include two-class and multiple-class spectral clustering, and algorithms based on Euclidean distances. The Born rule actually has the potential to upgrade many classical algorithms into quantum or quantum-inspired algorithm.

[Leifer and Poulin, 2008] reported quantum belief propagation. Belief propagation, also known as sum-product message passing, is a message passing algorithm for performing Bayesian inference on graphical models of classical probability distributions, such as Bayesian Networks, Factor Graphs and Markov Random Fields. It calculates the marginal distribution for each unobserved node, conditional on any observed nodes. Belief propagation algorithms prove to be amongst the most powerful known methods for deriving probabilistic inferences amongst large numbers of random variables. Quantum belief propagation presents a generalization of

these classical concepts and methods to the quantum case, driven by the idea that quantum theory is also a generalized probability theory, which is noncommutative and operator-valued.

[Weinstein and Horn, 2009] reported a quantum-inspired clustering method, named dynamic quantum clustering. This method associates data samples in some feature space with a Schrodinger equation whose potential is determined by the data. This is a typical approach to transforming data science into physical problems. Data patterns decide the potential landscape of the data quantum system. Here Schrodinger evolution is used for clustering, an important unsupervised learning problem.

[Huang et al., 2012] reported a quantum-inspired anomaly detection algorithm. In machine learning, anomaly detection (outlier detection) is the identification of data samples which do not conform to an expected pattern or other samples in a data set. This work has made the first attempt to apply quantum mechanics to anomaly detection in high-dimensional datasets for data mining. It originally proposed Fermi Density Descriptor which can represent the probability of measuring a fermion at a specific location for anomaly detection, where Fermi-Dirac statistics replace classical statistics to describe the location of those non-physical particles, namely data samples.

[Dong et al., 2012] reported a novel quantum-inspired reinforcement learning algorithm for navigation control of autonomous mobile robots. The main originality is to adopt a probabilistic action selection policy and a new reinforcement strategy, which are inspired, respectively, by the collapse phenomenon in quantum measurement and amplitude amplification in quantum computation. It originally employs quantum probabilistic concepts, including the Born rule and probability amplitudes, to replace classical probability theory. According to the experimental analysis, the Quantum-Inspired RL is more robust to learning rates and initial states than traditional reinforcement learning. It indicates that quantum-inspired methods may generalize the power of classical methods. And, different from quantum machine learning that requires a quantum machine, quantum-inspired machine learning can be applied to real-world problems directly.

[Blacoe et al., 2013] reported a quantum-inspired semantic space model for distributional semantics. This work formulates a formal quantum theoretical framework for capturing lexical meaning by using the language of quantum matrices. It tries

to represent the meaning of words by density matrices that encode dependency neighborhoods. Several quantum elements such quantum superposition and entanglement are also included into this model. The creative application of density matrices show the general potential of quantum theory as well as great power in empirical analysis.

[Stoudenmire and Schwab, 2016] reported supervised learning with quantum-inspired tensor networks. The most successful use of tensor networks in physics so far has been quantum many-body problem, where combining N independent systems corresponds to taking the tensor product of their individual state vectors. With the goal of applying similar tensor networks to machine learning, quantum tensor networks become an important approach to quantum-inspired neural networks learning. This work demonstrates how algorithms for optimizing such tensor networks can be adapted to supervised learning tasks by using matrix product states. Overall, quantum tensor networks are likely to become a useful framework for applying quantum theory to neural networks.

Chapter 3

Quantum-Inspired Regression Forest

3.1 Introduction

We interpret the ensemble learning process in several quantum physics concepts, and merge quantum-inspired techniques into the ensemble method naturally. We mainly focus on Tree Ensemble methods due to two truths. First, Tree Ensemble is a powerful and robust method that is widely used in multiple domain's tasks. An significant improvement on this popular method could make QIS very valuable. Second, base learners are constructed independently and in parallel. This indicates that we may take many elements of Tree Ensemble as black boxes except for generating the feature subsets. We make QI Forest and Random Forest only differ in generating feature subsets for individual learners. It provides the advantage that we can ensure any performance differences are purely caused by the proposed Quantum-Inspired Subspace method.

In Section 3.2, we present quantum interpretations and the proposed algorithms. We show the process how quantum mechanics inspires us to invent a novel ensemble method. In Section 3.3, we provide a solid mathematical proof for the advantage of the proposed algorithm. We prove that Quantum-Inspired Forest Regressors' advantage over Random Forest in case of the first order approximation. In our mathematical analysis, the Linear Regressor is nonlinear base regressors' first order approximation. In Section 3.4, we empirically compare Quantum-Inspired Forest

Regressors and Random Forest Regressors on date sets from UCI Repository [Lichman, 2013]. What's more, we perform one more empirical comparison, where we take Linear Regressors as base learners instead of Decision Tree Regressor. In Section 3.5, we discuss and summary our main work.

3.2 The Quantum-Inspired Approach

3.2.1 Quantum Interpretations

We believe quantum physics may provide a novel and valuable viewpoint for machine learning. Quantum physics shares similar forms with machine learning. And these similar forms or equations encourage us to think about machine learning in a quantum theoretical way. The motivation behind our works starts from Principal Component Analysis (PCA) and density matrices. [Nielsen and Chuang, 2010] introduced density matrix and operators in detail. In quantum mechanics, physicists often denote a pure state as a state vector $|\psi\rangle$. However, there exist mixed states, which cannot be written as a state vector. A mixed state corresponds to a probabilistic mixture of pure states, also called a quantum ensemble. A density matrix is a matrix that describes a quantum mixed state, an ensemble of several pure states. We show how to establish connections between density matrix and quantum operators to PCA as follows.

We interpret principal components as eigenstates in a mixed state. Suppose we are given a data set $X \in \mathbb{R}^{n\times m}, \vec{y} \in \mathbb{R}^n$ for a regression or classification problem. X, a $n \times m$ data matrix, contains n data samples, and each feature vector $\vec{x^i}$ has m features. The target variable vector $\vec{y}$ is a vector with a length of n. Singular Value Decomposition (SVD) is a widely used method to perform PCA [Wall et al., 2003]. For a data matrix X, there exist matrices U, S, V satisfying

$$X = USV^\top, \tag{3.1}$$

where U is a $n \times n$ unitary matrix, S is a $n \times m$ matrix with non-negative real numbers s_i on the diagonal line, and V is a $m \times m$ unitary matrix. We define the Gram matrix $\rho = XX^\top$ that is a symmetric and positive semi-definite $n \times n$

matrix. And S has r non-zero diagonal elements. And then we have

$$\rho = XX^\top = USV^\top VS^\top U^\top = U\Sigma U^\top, \tag{3.2}$$

where $\Sigma = SS^\top$ is a $n \times n$ diagonal matrix with diagonal elements $\sigma_i = s_i^2$. Column vectors of US are equal to principal components in PCA. And people often use first k column features US as dimension-reduced k-dimension feature vectors.

The quantum journey begins from here. As the density matrix of quantum mechanics is Hermitain, positive semi-definite and of trace 1, if we normalize the Gram matrix ρ by multiplying a factor $\frac{1}{Tr(\rho)}$, the Gram matrix can be regarded as a density matrix in quantum theory. For simplicity of our notation, we also denote the density matrix by ρ. So we redefine ρ and U with a normalization factor as

$$\rho = \frac{XX^\top}{Tr(XX^\top)} = U\Sigma U^\top. \tag{3.3}$$

Let $\vec{u_i}$ denote the ith column vector of matrix U, so $\vec{u_i}$ is also a pure state vector, which denotes $|u_i\rangle$ in quantum theory. As we have replaced the Gram Matrix by the normalized ρ, the sum of diagonal elements of Σ, $\sum_{i=1}^{n} s_i^2$, is equal to 1. The density matrix ρ describing the data matrix as a mixed state is also an operator of the form

$$\rho = \sum_{i=1}^{n} s_i^2 |u_i\rangle\langle u_i|. \tag{3.4}$$

The rank of matrix X indicates how many pure states we have in a quantum ensemble. As the rank of matrix X is r, we have

$$\rho = \sum_{i=1}^{r} s_i^2 |u_i\rangle\langle u_i|. \tag{3.5}$$

Physically, it means a data matrix X can be regarded as a mixed state or a quantum ensemble consisting of r pure states. In physics, an quantum ensemble, namely an ensemble of pure states, can reflect statistical expectations of quantum systems. And the variance s_i^2 is the fraction(weight probability) of the ensemble in each pure state $|u_i\rangle$.

On the one hand, the quantum interpretation treats PCA naturally as a dimensionality reduction process. In machine learning, researchers usually preserve the

first k components with largest variance values as dimensionality reduced features. In quantum mechanics, PCA means that we remove several non-principal eigenstates from the mixed state and preserve those principal eigenstates so that we prepare a new mixed state consisting of less eigenstates. The new state is exactly a low-rank approximated copy of the original mixed state. Obviously, PCA makes clear sense to us from a viewpoint of physics. But PCA is also a naive and biased operation that assigns uniform weights to principal eigenstates and weight 0 to non-principal eigenstate. If we preserve a large number of principal states, PCA will indeed provide us a deterministic low-dimensional feature set which can be used for training only one but relatively accurate learner. On the other hand, Random Subspace is used in Random Forest to produce diversified feature sets for base learners. Different from PCA, Random Subspace completely randomly assigns uniform weights to all principal components and non-principal components. Diversified low-dimensional feature sets can be repeatedly produced by Random Subspace, but the diversity comes at a large cost of accuracy. Randomly removing a lot of features of course unavoidably cause a large loss of information. PCA can be used for training a relatively accurate but deterministic base learner, while Random Subspace can be used for training relatively diversified but inaccurate base learners. So the problem is may we find a better way to balance accuracy and diversity? In scenarios of ensemble learning, we do have a reasonable way. Quantum mechanics naturally provides us the fraction probability of each eigenstate. The fraction probability $\frac{s_k^2}{\sum_{i=1}^r s_i^2}$ must possess a physics meaning. And we believe this physics meaning also indicates a valuable meaning in ensemble learning. We will mathematically prove the theoretical connection in Section 3.3. We prove that $\frac{s_k^2}{\sum_{i=1}^r s_i^2}$ is the optimal probabilistic distribution under Gaussian assumptions of model parameters.

And the second quantum interpretation is we can also regard regression as a state preparation process that we operate several pure states to approximate a target state $|y\rangle$. Translated in quantum theoretical language, it can be written as

$$\rho_y = |y\rangle\langle y| = \hat{A}\rho_x\hat{A}^\dagger, \tag{3.6}$$

where the state operation is noted by some quantum operator $\hat{A}$. So the quantum mechanism of regression tasks can be understood as we learn a Model Operator to operate eigenstates in a mixed to approximate a target pure state under some metrics. From a quantum theoretical viewpoint, the importance of an eigenstate

$|u_i\rangle$ is reflected by the Transition Probability from en eigenstate $|u_i\rangle$ jumping into the target state $|y\rangle$. We denote Transition Probability as t_i. Obviously, the Transition Probability is a parameter decided by model operator, the eigenstate, and the target state together. Aggregating fraction probabilities and transition probabilities together, the Fraction Transition Probability for the ith principal component is proportional to $s_i^2|\langle y|\hat{A}|u_i\rangle|^2$. So we take the the Fraction Transition Probability for the ith principal component as

$$p_k = \frac{s_k^2 t_k^2}{\sum_{i=1}^{r} s_i^2 t_i^2}. \tag{3.7}$$

In Section 3.3, we prove that Transition Probability Ampplitudes happen to equal to parameters of linear regression mapping from X to y in the first order approximated situation. According to the heuristical Fraction Transition Probabilities, we successfully propose Quantum-Inspired Subspace Method and Quantum-Inspired Forest.

3.2.2 Algorithm

Random Subspace is a fast and efficient ensemble method widely used in many algorithms, including Random Forest. Random Subspace randomly select a subset of features for training a base learner. But Quantum-Inspired Subspace can utilize the extra information inspired by quantum mechanics. We first preprocess the input data matrix X by using full-rank PCA. Different from either preserving principal components with largest eigenvalues or random subspace, QIS selects a component in a probability proportional to the corresponding Fraction Transition Probability. Under Gaussian assumptions of model parameters, we let $p_k = \frac{s_k^2}{\sum_{i=1}^{r} s_i^2}$ for the component k. When we replace Random Subspace by Quantum-Inspired Subspace for Random Forest, we obtain a novel algorithm, namely Quantum-Inspired Forest. We note that, in principle, full-rank PCA preprocessing generally can neither improve nor damage algorithm performance. The additional computational cost of the proposed algorithm is only brought by Principal Component Analysis and several matrix operations for computing Fraction Transition Probabilities. So it is a very low cost in practice.

Denote by $h_1, \ldots, h_T$ the regressors in the ensemble and by F, the feature set. As with most ensemble methods, we need to choose ensemble size T in advance. All

base regressors can be trained in parallel, which is also the case with Bagging and Random Forests. Algorithm 3 explains how to generate construct the training feature set F_i for regressor h_i. And we modify Random Forest into Quantum-Inspired Forest by employing Quantum-Inspired Subspace to generate ensemble feature subsets instead of Random Subspace.

Algorithm 3: Quantum-Inspired Subspace for generating feature subsets

function QuantumInspiredSubspace (X, y, F, T, K)
Input : **real** $[n \times m]$ X: the data matrix
Input : **real** $[n]$ $\vec{y}$: the target variable vector
Input : **int** T: the ensemble size
Input : **int** K: the target space dimensionality $K = \alpha m$
Output: feature subsets $\{F_i | i = 1, \ldots, T\}$
Preprocess data matrix $X_R \leftarrow PCA(X)$ by using full-rank PCA
Compute Fraction Probabilities $\vec{p_s} \leftarrow$ the diagonal elements of covariance matrix $X^\top X$
Compute Transition Probability Amplitudes $\vec{t} \leftarrow (X_R^\top X_R)^{-1} X_R^\top \vec{y}$ which are LR parameters
Compute Transition Probabilities $\vec{p_t} \leftarrow \vec{t} . * \vec{t}$
Compute Fraction Transition Probabilities $\vec{p} \leftarrow \frac{\vec{p_s} . * \vec{p_t}}{norm(\vec{p_s} . * \vec{p_t})}$
for $i \leftarrow 1$ ***to*** T **do**
    Select K unique random integers $a_1, \ldots, a_K$ from $[1, m]$ in probabilities of p_{a_i}
    $F_i \leftarrow \{a_1, \ldots, a_K\}$
return $\{F_i | i = 1, \ldots, T\}$

Algorithm 4: Quantum-Inspired Regression Forest

function QIForest (S, F, T, K)
Input : A training set $S = (x^1, y^1), ..., (x^n, y^n)$, features F, and the forest size T, the target space dimensionality K
Output: Quantum-Forest H
$H \leftarrow \emptyset$
$\{F_i | i = 1, \ldots, T\}$ generated by function QISubspace (X, y, F, T, K)
for $i \leftarrow 1$ ***to*** T **do**
    $S^i \leftarrow$ a bootstrap sample from S
    $F^i \leftarrow F_i$
    $h_i \leftarrow$ RegressionTreeLearn(S^i, F^i)
    $H \leftarrow H \cup \{h_i\}$
return H

It is worthy noting that Quantum-Inspired Subspace is a general method which can be easily applied with other ensemble methods and multiple base learners together. QIS also lend itself naturally to parallel processing, as ensemble feature

sets and individual learners can be built in parallel. QIS is not only naturally applicable to Tree Ensembles, but also makes sense for any ensemble regressors whose diversity is based random feature selections.

3.3 Theoretical Analysis and Proof

In this section, we prove the advantage of Quantum-Inspired Subspace through error-variance-covariance decomposition that combines error-ambiguity decomposition and bias-variance-covariance decomposition together. The proof states that the advantage of QIS theoretically increase ensemble ambiguity and decrease the individual error expectation in the first order approximation. And in our empirical analysis, the experimental results support the advantage is still approximately applicable to nonlinear models, such as Decision Tree. The mathematical proof for ensemble classification cannot hold in the same way, although our empirical analysis support that Quantum-Inspired Forest Classifiers can be favorably compared with Random Forest Classifiers.

3.3.1 Error-Variance-Covariance Decomposition

In this section, we show how to obtain Error-Variance-Covariance Decomposition. We organize several known conclusions together referring to derivations in Chapter 5.2 of [Zhou, 2012]. Assume that the task is to use an ensemble of T base regressors $h_1, h_2, ..., h_T$ to approximate a function $f : \mathbb{R}^m \to \mathbb{R}$. And a simple averaging policy is used for the final ensemble prediction

$$H(\vec{x}) = \frac{1}{T}\sum_{i=1}^{T} h_i(\vec{x}), \tag{3.8}$$

where $H(\vec{x})$ is the ensemble learner. And we define several notations here. The generalization error and ambiguity of a base learner is respectively defined as

$$err(h_i) = (h_i(x) - f(x))^2, \tag{3.9}$$

$$ambi(h_i) = (h_i(x) - H(x))^2. \tag{3.10}$$

And we also note the expectation prediction of a base learner h_i as

$$\mathbb{E}[h_i] = \int h_i(x)p(x)dx, \tag{3.11}$$

where $p(x)$ is the density function for data x. On the one hand, [Krogh et al., 1995] proposed the error-ambiguity decomposition of ensemble learning, and the generalization error of the ensemble can be written as

$$err(H) = \overline{err}(H) - \overline{ambi}(H), \tag{3.12}$$

where $\overline{err}(H) = \frac{1}{T}\sum_{i=1}^{T} err(h_i)$ is the average of individual generalization errors, and $\overline{ambi}(H) = \frac{1}{T}\sum_{i=1}^{T} ambi(h_i)$ is the average of ambiguities which is also called the ensemble ambiguity. A basic truth is that the larger the ensemble ambiguity, the better the ensemble.

On the other hand, [Ueda and Nakano, 1996] developed the bias-variance-covariance decomposition. The averaged bias, averaged variance, and averaged covariance of the individual learners are defined respectively as

$$\overline{bias}(H) = \frac{1}{T}\sum_{i=1}^{T}(\mathbb{E}[h_i] - f), \tag{3.13}$$

$$\overline{variance}(H) = \frac{1}{T}\sum_{i=1}^{T}\mathbb{E}[(h_i - \mathbb{E}[h_i])^2], \tag{3.14}$$

$$\overline{covariance}(H) = \frac{1}{T(T-1)}\sum_{i=1}^{T}\sum_{j\neq i, j=1}^{T}\mathbb{E}[(h_i - \mathbb{E}[h_i])(h_j - \mathbb{E}[h_j])]. \tag{3.15}$$

And then the bias-variance-variance decomposition of ensemble is written as

$$err(H) = \overline{bias}(H)^2 + \frac{1}{T}\overline{variance}(H) + (1 - \frac{1}{T})\overline{covariance}(H). \tag{3.16}$$

We may establish a bridge connecting the error-ambiguity decomposition and the bias-variance-covariance decomposition[Brown et al., 2005a,b] as

$$\overline{err}(H) - \overline{ambi}(H) = \overline{bias}(H)^2 + \frac{1}{T}\overline{variance}(H) + (1 - \frac{1}{T})\overline{covariance}(H). \tag{3.17}$$

And then we have

$$\overline{err}(H) = \mathbb{E}\left[\frac{1}{T}\sum_{i=1}^{T}(h_i - f)^2\right] = \overline{bias}^2(H) + \overline{variance}(H), \tag{3.18}$$

and

$$\begin{aligned}\overline{ambi}(H) &= \mathbb{E}\left[\frac{1}{T}\sum_{i=1}^{T}(h_i - H)^2\right] \\ &= \overline{variance}(H) - variance(H) \\ &= (1 - \frac{1}{T})\overline{variance}(H) - (1 - \frac{1}{T})\overline{covariance}(H).\end{aligned} \tag{3.19}$$

Finally, we obtain Error-Variance-Covariance Decomposition as

$$err(H) = \overline{err}(H) - (1 - \frac{1}{T})\overline{variance}(H) + (1 - \frac{1}{T})\overline{covariance}(H). \tag{3.20}$$

And the generalization error expectation is written as

$$\mathbb{E}[err(H)] = \mathbb{E}[err(h_i)] - (1 - \frac{1}{T})\mathbb{E}[\text{var}(h_i)] + (1 - \frac{1}{T})\mathbb{E}[\text{covar}(h_i, h_j)]. \tag{3.21}$$

Actually, there is no simple ensemble method that can minimize the expectation of $err(H)$. Fortunately, according to our following analysis, we find Quantum-Inspired Subspace method can decrease $\mathbb{E}[err(h_i)]$, $-\mathbb{E}[\text{var}(h_i)]$ and $\mathbb{E}[\text{covar}(h_i, h_j)]$ simultaneously, compared to Random Subspace method.

3.3.2 Ensemble Ambiguity

We decide to prove that Quantum-Inspired Subspace can improve ensemble ambiguity $\overline{ambi}(H)$ and decrease individual generalization errors $\overline{err}(H)$ simultaneously. And according to the Error-Variance-Covariance Decomposition relations, increasing ensemble ambiguity is equivalent to increasing $\mathbb{E}[\text{var}(h_i)] - \mathbb{E}[\text{covar}(h_i, h_j)]$. We want to figure out how to improve $\overline{variance}(H) - \overline{covariance}(H)$. We note that nonlinear regression models degenerate to Linear Regression (LR) in case of the first order approximation, just like how Taylor series expansion works. In the case of the first order approximation, we ignore all high order nonlinear terms. And we find the approximated case holds well for regression trees, as regressions tree also aim at finding linear relationships between features and target variables.

So in this subsection, what we decide to prove actually is, with Linear Regressors as base regressors, Quantum-Inspired Subspace Ensemble method can increase ensemble ambiguity strictly. Although it seems naive to consider ensemble linear regressors only, the mathematical analysis provides important theoretical insights about other nonlinear base learners. Assuming model parameters are independent distributed Gaussian random variables, we further know QIS can even decrease the averaged individual generalization errors. Given general data sets instead a certain data set, the Gaussian assumption that takes model parameters as Gaussian random variables is reasonable and realistic for most machine learning models. But the independence assumption only approximately holds for several linear models, luckily including Linear Regression. However, although what we prove only holds for most simplified cases, we find the proof still partly holds in more general situation. For simplicity, we use several new notations in proof. We denote the original data matrix as $X' = USV^\top$ and its linear regression parameters as $w'_k \sim \boldsymbol{N}(0, \sigma^2)$, where $k = 1, \ldots, m$. We can safely assume each parameter independently obeys normal distribution as we have no prior knowledge about the importance of features. Considering a certain data set, without training, we of course know nothing about each feature's importance. Considering model performance on general data sets, the independent Gaussian assumption is also realistic.

Let's turn to the full-rank PCA preprocessed data matrix $X = X'V$ and its linear regression parameters $\vec{w} = V^\top \vec{w'}$. As V is an orthogonal matrix, a model parameter w still obeys a Gaussian distribution, $w \sim \boldsymbol{N}(0, \sigma^2)$. We may regard columns vectors of preprocessed matrix X as input features. So we define individual learners as

$$h_i(\vec{x}) = \sum_{k \in F_i} w_k s_k u_k = \sum_{k \in F_i} w_k x_k, \tag{3.22}$$

where s_k is the kth-largest singular value, and F_i is the feature subset for the ith base learner. Benefitting from orthogonalized preprocessing and LR as base learners, model parameters stay invariant even trained by variant feature subsets. We call this characteristic as Parameter Invariance under variant feature subsets. In our proof, the Parameter Invariance of base learners is a key prerequisite for improving ensemble ambiguity. And besides Linearity, how Parameter Invariance is approximately applicable to nonlinear models is another key factor deciding how generally the proof may hold.

We first analyze the ensemble ambiguity $\overline{ambi}(H)$ which is equivalent to $(1 - \frac{1}{T})(\overline{variance}(H) - \overline{covariance}(H))$. According to Equation 3.22, we have

$$\mathbb{E}[\text{covar}(h_i, h_j)] = \mathbb{E}\left[\text{covar}(\sum_{k\in F_i} w_k s_k u_k, \sum_{k\in F_j} w_k s_k u_k)\right] = \sum_{k=1}^{r} w_k^2 s_k^2 p_k^2 \tag{3.23}$$

and

$$\mathbb{E}[\text{covar}(h_i)] = \mathbb{E}\left[\text{covar}(\sum_{k\in F_i} w_k s_k u_k, \sum_{k\in F_i} w_k s_k u_k)\right] = \sum_{k=1}^{r} w_k^2 s_k^2 p_k \tag{3.24}$$

with a constraint of $\sum_{k=1}^{r} p_k = 1$ and a statistical assumption that $w_k \sim \boldsymbol{N}(0, \sigma^2)$ is a normal random variable. We note that Random Subspace just naively sets $p_k = \frac{1}{r}$. We have a better solution to increase the ensemble ambiguity. We find the solution

$$p_k = \frac{w_k^2 s_k^2}{\sum_{i=1}^{r} w_i^2 s_i^2}, \tag{3.25}$$

which can exactly minimize $\mathbb{E}[\text{covar}(h_i, h_j)]$. What's more, it further increases $\mathbb{E}_{QI}[\text{var}(h_i)]$ compared with $\mathbb{E}_{RS}[\text{var}(h_i)]$,

$$\sum_{k=1}^{r} w_k^2 s_k^2 p_k > \sum_{k=1}^{r} \frac{w_k^2 s_k^2}{r}. \tag{3.26}$$

So we have

$$\mathbb{E}_{QI}[\text{var}(h_i)] > \mathbb{E}_{RS}[\text{var}(h_i)], \tag{3.27}$$

and

$$\mathbb{E}_{QI}[\text{covar}(h_i, h_j)] < \mathbb{E}_{RS}[\text{covar}(h_i, h_j)]. \tag{3.28}$$

The solution we find is in same forms as the Fraction Transition Probability that the density matrix interpretation indicates. The Transition Probabilities of Linear Regression Quantum Operator are exactly the linear regression model parameters. For a certain data set, we can get certain weights $\vec{w}$. For general data sets, we still have normal distribution assumption so that $\frac{w_k^2}{s^2} \sim \chi(1)$ is a chi-squared random variable. [Provost and Rudiuk, 1994] revealed the analytical probability density

function of p_k, and we know its expectation must be

$$\hat{p_k} = \frac{s_k^2}{\sum_{i=1}^{r} s_i^2}, \tag{3.29}$$

which are exactly Fraction Probabilities given by quantum interpretations. Our theoretical analysis of Quantum-Inspired Subspace shows that

$$\mathbb{E}_{QI}[\overline{ambi}(H)] > \mathbb{E}_{RS}[\overline{ambi}(H)]. \tag{3.30}$$

3.3.3 Individual Errors

In this subsection, we want to explain that Quantum-Inspired Subspace, $p_k = \frac{w_k^2 s_k^2}{\sum_{i=1}^{r} w_i^2 s_i^2}$, tends to decrease the averaged individual error, namely $\overline{err}(H)$. Actually this conclusion is trivial.

Although for a certain data set, we cannot conclude that each original feature equally contributes to the model performance. But, for general data sets, under the Gaussian assumption of model parameters $\vec{w}$, we can safely say that the expectation contribution of each original feature tends to be equal. As we have preprocessed data sets by using full-rank PCA, the widely accepted prior belief that principal components with larger variance carry more information supports the conclusion that QIS decreases the individual errors. As we know

$$\overline{err}(H) = \frac{1}{T}\sum_{1}^{T} err(h_i) = \mathbb{E}[(h_i - f)^2], \tag{3.31}$$

the widely accepted prior belief can be written as

$$\begin{aligned} \mathbb{E}_{QI}[(h_i - f)^2]] &> \mathbb{E}_{RS}[(h_i - f)^2]] \\ \mathbb{E}_{QI}[\overline{err}(H)] &> \mathbb{E}_{RS}[\overline{err}(H)]. \end{aligned} \tag{3.32}$$

According to Equation 3.12, 3.30 and 3.32, we finally prove the conclusion that

$$\mathbb{E}_{QI}[err(H)] < \mathbb{E}_{RS}[err(H)]. \tag{3.33}$$

The proof indicates that the correlation between base learners is decreased with an expectation enhancement in their strength. Statistically speaking, QIS can even improve base learners' performance and ensemble ambiguity simultaneously.

TABLE 3.1: QI Forest Regressors vs. Random Forest Regressors: $\alpha = 0.5$; ensemble size $T = 30$; training instances $N = 60\%$.

Data	**Instances**	**Dimension**	**QI-Forest**	**R-Forest**	+/−
Abalone	4177	8	$0.3204_{0.0055}$	$0.3350_{0.0073}$	++
Communities Crime	1994	122	$0.2763_{0.0025}$	$0.3016_{0.0080}$	++
Communities Crime Unnormalized 1	2215	140	$0.2515_{0.0053}$	$0.2766_{0.0112}$	++
Communities Crime Unnormalized 2	2215	140	$0.2125_{0.0052}$	$0.2697_{0.0073}$	++
Facebook Metrics	500	11	$0.1580_{0.0302}$	$0.1267_{0.0480}$	−
Forests Fire	517	8	$0.8296_{0.0175}$	$0.8369_{0.0231}$	+
Housing	505	13	$0.2011_{0.0089}$	$0.2492_{0.0171}$	++
Slump Test	103	9	$0.1704_{0.0103}$	$0.2678_{0.0276}$	++
Wine Quality Red	1599	11	$0.4379_{0.0060}$	$0.4622_{0.0118}$	++
Wine Quality White	4898	11	$0.4056_{0.0025}$	$0.4087_{0.0075}$	+

For the individual error expectation, the quantum-inspired weighted probabilistic selection strategy tends to work at least the same good as the uniform probabilistic selection strategy. We also note that this conclusion is statistically correct but not guaranteed on some certain data set.

Although Transition Probabilities of nonlinear models are quite difficult to derive, the Gaussian assumption is always realistic. We argue that Fraction Probabilities are at least approximately applicable to most machine learning models. We conjecture that, even if without Model Transition Probabilities, Fraction Probabilities are still very likely to improve ensemble learners, including classifiers.

Besides the simplified case of linear regression, we also need to discuss how Decision Tree may approximately preserve the first order linearity approximation and Parameter Invariance under variant feature subsets. On the one hand, the strategy to find the best split for constructing a regression tree is based on the criteria of mean square error reduction. So the feature split order can stay approximately invariant under variant feature subsets, whose mechanism is close to Parameter Invariance under variant feature subsets. On the other hand, the Decision Tree regressors learn linear relationships between features and target variables. The regression function based a tree regression mapping from X to $\vec{y}$ can be very simple like a combination of N step functions. In the limit of $N \to +\infty$, a combination of N step functions tends to become a approximately smooth function. The first order approximation makes sense in this situation.

TABLE 3.2: QI Ensemble Linear Regressor vs. Random Ensemble Linear Regressors: $\alpha = 0.5$; ensemble size $T = 30$; training instances $N = 60\%$.

Data	**Instances**	**Dimension**	**QIE-LR**	**RE-LR**	+/−
Abalone	4177	8	$0.3466_{0.0061}$	$0.4186_{0.0207}$	++
Communities Crime	1994	122	$0.2398_{0.0021}$	$0.3220_{0.0275}$	++
Communities Crime Unnormalized 1	2215	140	$0.0213_{0.0001}$	$0.1935_{0.0226}$	++
Communities Crime Unnormalized 2	2215	140	$0.1104_{0.0022}$	$0.2389_{0.0202}$	++
Facebook Metrics	500	11	$0.0044_{0.0004}$	$0.0675_{0.0196}$	++
Forests Fire	517	8	$0.7298_{0.0016}$	$0.7332_{0.0029}$	++
Housing	505	13	$0.2695_{0.0026}$	$0.3883_{0.0247}$	++
Slump Test	103	9	$0.1075_{0.0042}$	$0.2624_{0.0446}$	++
Wine Quality Red	1599	11	$0.4764_{0.0023}$	$0.4833_{0.0103}$	++
Wine Quality White	4898	11	$0.5245_{0.0010}$	$0.5334_{0.0073}$	++

TABLE 3.3: QI Forest Regressors vs. Random Forest Regressors: ensemble size $T = 30$; training instances $N = 60\%$; adjust α respectively as $0.125, 0.25, 0.5, 0.75, 1.0$. When $\alpha = 1.0$, QI Forest degenerates into Random Forest.

α	**QI-Forest**	**R-Forest**
0.125	$0.4251_{0.0154}$	$0.4932_{0.0208}$
0.25	$0.3411_{0.0082}$	$0.4186_{0.0182}$
0.5	$0.3263_{0.0094}$	$0.3544_{0.0168}$
0.75	$0.3253_{0.0099}$	$0.3313_{0.0118}$
1.0	$0.3377_{0.0095}$	−

TABLE 3.4: QI Forest Regressors vs. Random Forest Regressors: $\alpha = 0.5$; training instances $N = 60\%$; adjust ensemble size T respectively as $3, 10, 30, 100$.

T	**QI-Forest**	**R-Forest**
3	$0.4212_{0.0313}$	$0.4758_{0.0613}$
10	$0.3565_{0.0219}$	$0.3888_{0.0317}$
30	$0.3263_{0.0094}$	$0.3534_{0.0168}$
100	$0.3140_{0.0046}$	$0.3356_{0.0076}$

3.4 Empirical Analysis

Quantum-Inspired Subspace is easily incorporated into existing algorithms. In order to examine the benefit of QIS to ensemble performance, we modify standard Random Forest to incorporate Quantum-Inspired Subspace before the tree induction phase. In our empirical study of Quantum-Inspired Forest and Random Forest, we selected 10 UCI data sets that are commonly used in the machine learning literature in order to make the results easier to interpret and compare. As we take LR as base learners in our proof, we also compare Random Ensemble

TABLE 3.5: QI Forest Regressors vs. Random Forest Regressors: $\alpha = 0.5$; ensemble size $T = 30$; adjust training instances N respectively as $30\%, 40\%, 50\%, 60\%$.

Training Instances	QI-Forest	R-Forest
30%	$1.4372_{0.0359}$	$1.3462_{0.0555}$
40%	$0.8310_{0.0208}$	$0.8879_{0.0334}$
50%	$0.5551_{0.0131}$	$0.6209_{0.0243}$
60%	$0.3263_{0.0094}$	$0.3534_{0.0168}$

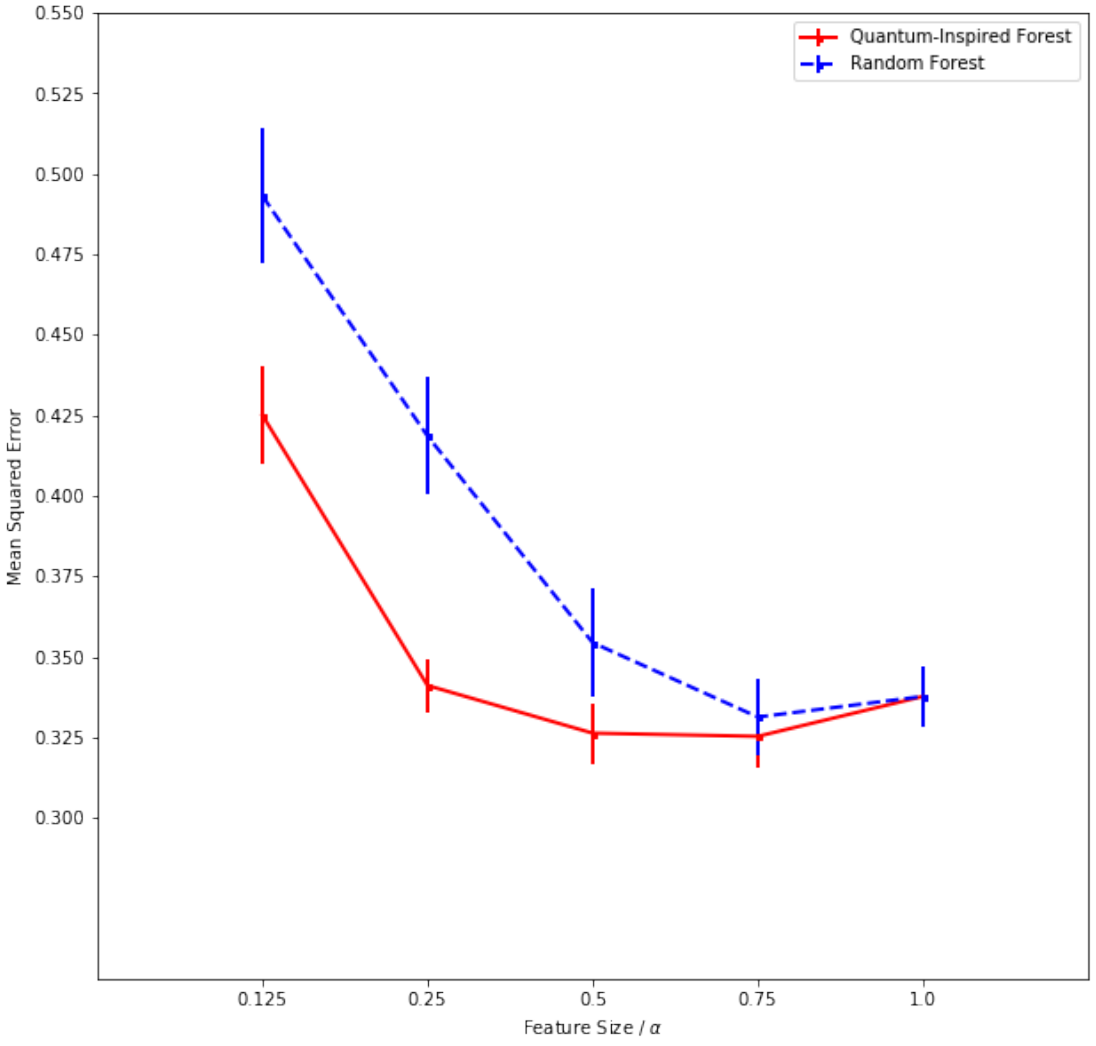

FIGURE 3.1: QI Forest Regressors vs. Random Forest Regressors: variant α

LR with Quantum-Inspired Ensemble LR in Table 3.2, where we replace Decision Tree by Linear Regression as base learners. Ensemble Linear Regressors are not useful in practice, but it can show how our proof holds.

We take the averaged mean square error (MSE) on 10 data sets as the metrics

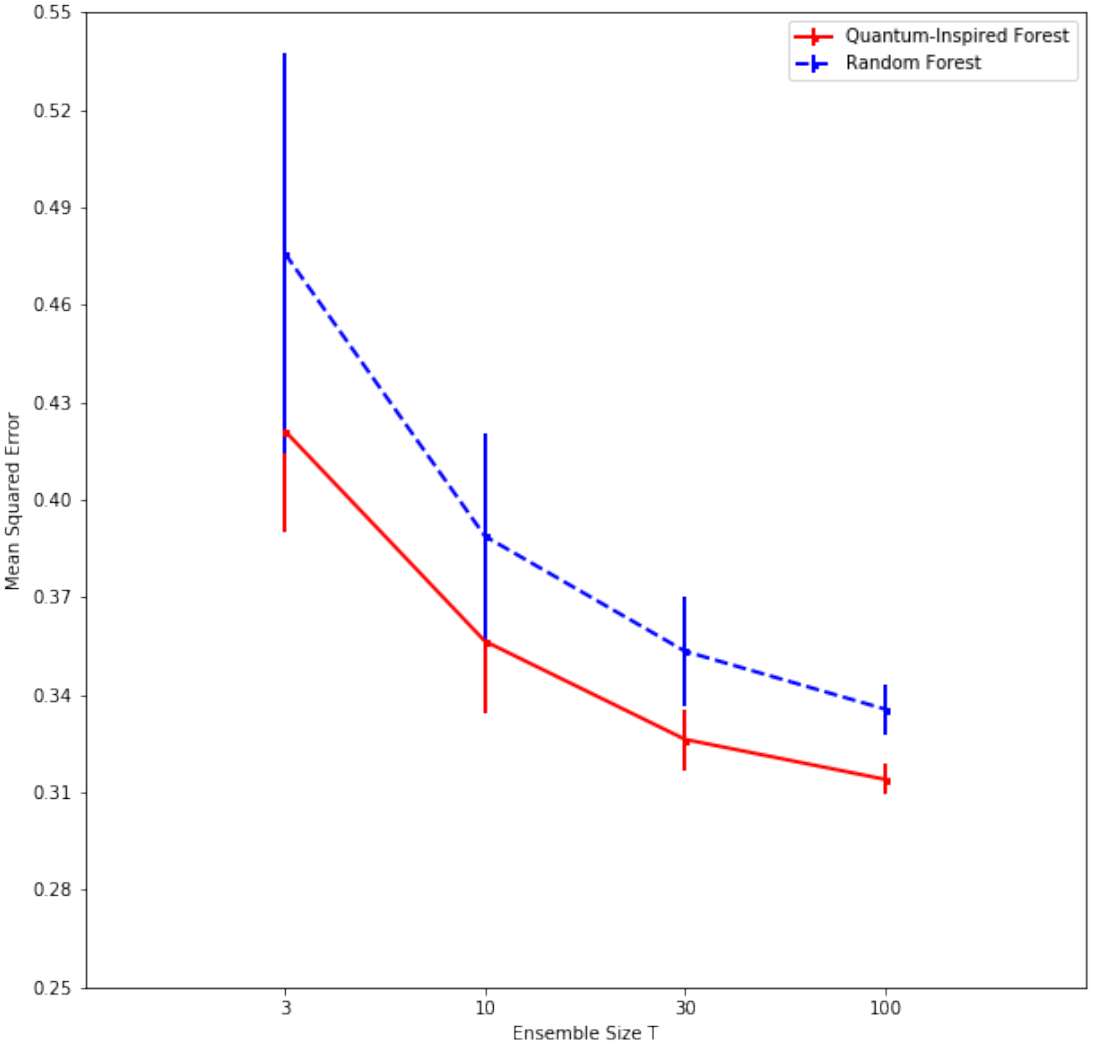

FIGURE 3.2: QI Forest Regressors vs. Random Forest Regressors: variant ensemble size T

in our empirical analysis. We decide to preprocess data sets, and take full-rank PCA preprocessed data matrix and mean normalized target variables y as preprocessed data sets. The first purpose is to ensure any performance differences are purely caused by the proposed Quantum-Inspired Subspace method rather than full-rank PCA preprocessing. We must leave the difference from full-rank PCA out. The second purpose is to remove the scale differences of different data sets so that we can fairly evaluate overall performance on 10 data sets. It's reasonable to start from full-rank PCA preprocessing because full-rank PCA is only an orthogonal transformation and causes no loss or distortion of information. As we mentioned above, in principle, full-rank PCA generally can neither improve nor damage algorithm performance. In practice, full-rank PCA usually brings in

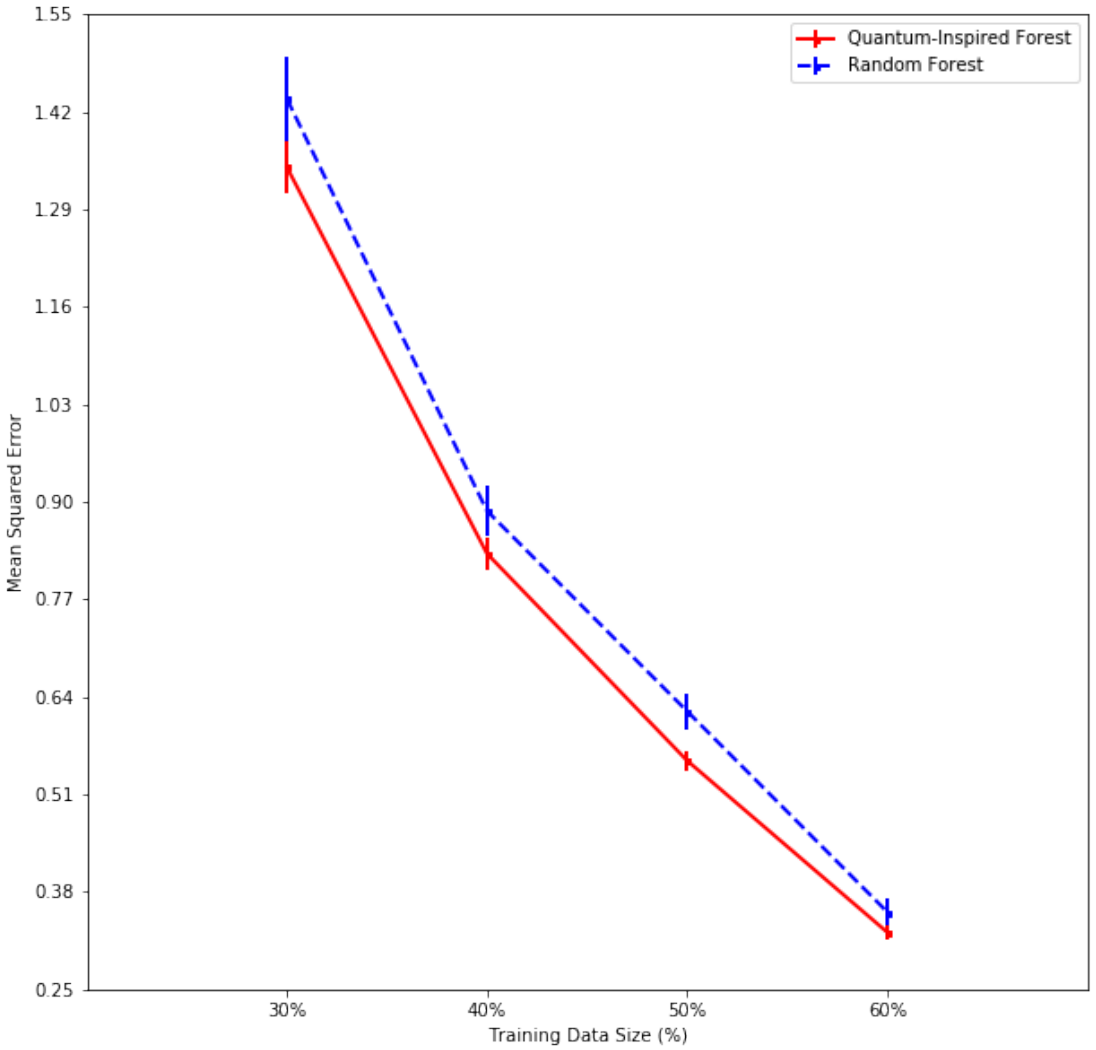

FIGURE 3.3: QI Forest Regressors vs. Random Forest Regressors: variant training data size

uncertain performance improvement or damage. So the full-rank PCA preprocessing is necessary for removing the uncertain performance differences from the orthogonal transformation.

We present mean square errors (with standard deviations as subscripts) on each data set or the averaged MSE on all 10 data sets in following tables. In Table 3.1 and 3.2, we denote better, significantly better, worse and significantly worse respectively as $+, ++, -$ and $--$. Instances is the data sample size. Dimension is the original data space dimensionality. We typically take 60% data instances as training data. As we notice the performance of Random Forest and Quantum-Inspired Forest adapt to hyperparameters in similar patterns, we decide to study two Forests' performance in multiple settings of forest hyperparameters.

We employ the strategy to compare Quantum-Inspired Forest and Random Forest counterpart in the same hyperparameter settings. This strategy removes the performance differences from tuning hyperparameters. And we repeat each experiment for 15 times to get statistically reliable results. The hyperparameter setting for Decision Tree base learners is always fixed in our experiments. The function to measure the quality of a split is mean square error. And a tree always find the best split at each node. And we also set no tree depth limit, and no minimum samples limit for splits and leaves. The default hyperparameter setting for forests is: ensemble size $T = 30$; select one half features to train base learners, which means $\alpha = 0.5$; the sub-sample size in Bagging is always the same as the original training sample size but the samples are drawn with replacements; and $N = 60\%$ samples are used as training data instances.

Both Table 3.1 and 3.2 share the defualt hyperparameter setting: $T = 30, \alpha = 0.5, N = 60\%$. We present MSE and standard deviations on each data set. Table 3.3 shares the default hyperparameter setting except that we set α respectively to $0.125, 0.25, 0.5, 1.0$. In this experiment, we want discover how QI Forest is compared to Random Forest with variant α settings. Table 3.4 shares the default hyperparameter setting except that we set ensemble size T respectively as $3, 10, 30, 100$. In this experiment, we want to discover how robustly QI Forest and Random Forest perform with small ensemble sizes. Table 3.5 shares the default hyperparameter setting except that we set training instances N respectively to $30\%, 40\%, 50\%, 60\%$. In this experiment, we want to how robustly QI Forest and Random Forest solve small data problems.

Table 3.1 shows the significant advantage of QI Forest Regressors in the default hyperparameter setting. QI Forest significantly outperform Random Forest on seven data sets; QI Forest slightly outperform Random Forest two data sets; and QI Forest perform slightly worse than Random Forest on only one data sets. The experimental result supports that Quantum-Inspired Forest Regressors outperform Random Forest Regressors in general situation. Table 3.2 further supports our theoretical analysis in first order approximation. QI Ensemble Linear Regressors significantly outperform Random Ensemble Linear Regressors on all 10 data sets. Table 3.3 supports that QI Forest not only outperforms Random Forest in one setting of α, but also beat Random Forest with multiple α settings. We notice that the smaller α is, the larger the advantage of QI Forest is. Especially when we select only a small number of features for training base learners, QI Forest can outperform

Random Forest significantly. Table 3.4 indicates that the performance difference of QI Forest and Random Forest increases as the ensemble size T decrease. It means QI Forest can perform significantly better than Random Forest with a limited forest size. Table 3.5 shows another advantage that QI Forest can solve small sample regression problems better than Random Forest. As we decrease training instances from 60% to 30%, the performance difference also increases. These experimental results show that given very limited computational resources or training data, QI Forest can outperform Random Forest.

As for classification tasks, our preliminary empirical analysis also finds similar advantage of QI Forest Classifiers. We discuss it in Chapter 4.

3.5 Discussion and Conclusion

From a heuristical viewpoint, we propose novel quantum interpretations for machine learning. On the one hand, we interpret eigenvalues of PCA as Fraction Probabilities in a mixed state. And it naturally indicates a generally accepted belief that eigenvalues / Fraction Probabilities can reflect the importance of each principal component. And we should let the probability of selecting a component be proportional to the corresponding Fraction Transition Probability. And we also interpret learning target variables as a state preparation process, that operates pure states to prepare a target state. So a machine learning model may be regarded as a Model Operator $\hat{A}$ in a physical sense, and determines the transition probability of an eigenstate jumping into a target state. We argue that both Fraction Probabilities and Transition Probabilities are beneficial to improve ensemble learning algorithms. However, considering our theoretical proof is only the first order approximately applicable to ensemble regressors, we only claim the advantage of QI Forest Regressors in this chapter.

From a viewpoint of theoretical analysis, we prove Fraction Probabilities and Transition Probabilities indeed can decrease ensemble errors in the simplified situation. According to our mathematical proof, in the case of Linear Regression as base learners, Transition Probabilities are exactly equal to model parameters of LR. For complex machine learning models, models' Transition Probabilities are quite difficult to derive. But for ensemble regressors, Transition Probabilities still make sense in the first order linearity approximation. As the Gaussian assumption of

model parameters is almost always realistic, we argue that Fraction Probabilities are approximately applicable to forest regressors. We conjecture that, even without Model Transition Probabilities, Fraction Probabilities are still likely to improve ensemble learning.

From a viewpoint of empirical analysis, our experiments strongly support the advantage of Quantum-Inspired Forest Regressors in multiple hyperparameter settings. In Table 3.2, we take Linear regression as base regressors, Quantum-Inspired Ensemble Linear Regressors significantly outperform Random Linear Regressors on all 10 data sets. In other tables, we take Decision Tree as base regressors, Quantum-Inspired Forest Regressors still outperform Random Forest Regressors significantly in variant hyperparameter settings. And we can ensure any performance differences are purely caused by the proposed Quantum-Inspired Subspace Method. Our empirical analysis concludes that Quantum-Inspired Forest perform more robustly than Random Forest, given very limited computational resources or training data. The observation provides QI Forest an extra advantage in extreme conditions.

In summary, we have two fold of contributions. First, we propose a novel ensemble method named Quantum-Inspired Subspace and Quantum-Inspired Regression Forest. Quantum-Inspired Subspace can be easily applied to diversified base learners and combined with other classical ensemble methods, such as Bagging. We incorporate Quantum-Inspired Subspace into Random Forest and propose Quantum-Inspired Forest. The additional computational cost is very cheap, equivalent to the cost of full-rank PCA preprocessing. Second, we propose quantum interpretations for several machine learning concepts, and successfully establish a theoretical bridge between quantum interpretations and ensemble learning. In future research, we consider two directions interesting. The first direction is to study Quantum-Inspired Subspace under complex circumstances, particularly classification tasks. The second direction is discover deeper theoretical connections between quantum mechanics and machine learning algorithms. May the mechanism of quantum entanglement be helpful for machine learning in some way? We also believe it will be very valuable to theoretically analyze quantum interpretations for neural networks.

Chapter 4

Quantum-Inspired Classification Forest

4.1 Quantum-Inspired Forest Classifiers

Inspired by the quantum interpretations in Chapter 3, we also proposed Quantum-Inspired Forest Classifiers. The mechanism of Quantum-Inspired Forest Classifiers is similar to Quantum-Inspired Forest Regressors. However, there are also differences between regression forest and classification forest. First, we use Classification Trees as base learners instead of Regression Trees. This is a trivial modification. Second, we remove transition probabilities from p_k, because we find that the influence of transition probabilities is quite unclear in classification tasks.

How may we treat "the transition probability amplitude" from a pure state to the target state in classification tasks? From a quantum theoretical viewpoint, the transition probability amplitude is a very natural quantum interpretation for the inner product of $\vec{x_k}$ and $\vec{y}$. The inner products are also used for marginal screening in high-dimensional regression. As for classification tasks, the inner products are not good metrics for marginal screening anymore. Worsely, neither the quantum interpretation nor the mathematical proof in Chapter 3 are directly applicable to Quantum-Inspired Forest Classifiers. Considering the unclear physical meaning for regression forest, we decide remove transition probabilities from p_k, and let p_k

stand for Fraction Probabilities only, which is

$$p_k = \frac{s_k^2}{\sum_{i=1}^{r} s_i^2}. \tag{4.1}$$

Algorithm 5: Quantum-Inspired Classification Forest

```
function QIForest (S, F, T, K)
Input  : A training set S = (x^1, y^1), ..., (x^n, y^n), features F, and the forest size
         T, the target space dimensionality K
Output: Quantum-Forest H
H ← ∅
{F_i | i = 1, ..., T} generated by function QISubspace (X, y, F, T, K)
for i ← 1 to T do
    S^i ← a bootstrap sample from S
    F^i ← F_i
    h_i ← ClassificationTreeLearn(S^i, F^i)
    H ← H ∪ {h_i}
return H
```

4.2 Empirical Analysis

In this section, we perform a empirical study for comparing Quantum-Inspired Forest Classifiers with the baseline Random Forest Classifiers. The experimental method and hyperparameter settings are similar to experiments in Chapter 3. We selected 9 UCI data sets that are commonly used in the machine learning literature in order to make the results easier to interpret and compare.

We take the averaged classification accuracy on 9 data sets as the metrics in our empirical analysis. Full-rank PCA preprocessing is also applied as before. We present classification accuracies (with standard deviations as subscripts) on each data set or the averaged accuracy on all 9 data sets in following tables. In Table 4.1, we denote better, significantly better, worse and significantly worse respectively as $+, ++, -$ and $--$. Instances is the data sample size. Dimension is the original data space dimensionality. We typically take 60% data instances as training data.

Due to the same reasons, we decide the experimental settings similar to the counterpart in Chapter 3. The hyperparameter setting for Decision Tree base learners is always fixed in our experiments. The function to measure the quality of a split is gini index. And a tree always find the best split at each node. And we also

TABLE 4.1: QI Forest Classifiers vs. Random Forest Classifiers: $\alpha = 0.5$; ensemble size $T = 30$; training instances $N = 60\%$.

Data	**Instances**	**Dimension**	**QI-Forest**	**R-Forest**	+/−
arcene	200	10000	$78.33_{2.12}$	$75.75_{3.35}$	+
breast	569	30	$94.21_{0.86}$	$94.16_{0.97}$	+
dexter	600	2600	$83.00_{2.23}$	$83.56_{1.35}$	−
glass	210	9	$79.68_{2.11}$	$77.62_{1.95}$	+
hill valley(noise)	600	100	$79.81_{1.47}$	$76.28_{2.94}$	+
hill valley	600	100	$92.58_{0.76}$	$92.08_{2.56}$	+
ionosphere	351	35	$91.33_{1.03}$	$96.90_{1.45}$	−−
madelon	2600	500	$63.06_{0.77}$	$51.89_{1.75}$	++
spambase	4600	57	$92.53_{0.13}$	$92.78_{0.28}$	−

TABLE 4.2: QI Forest Classifiers vs. Random Forest Classifiers: ensemble size $T = 30$; training instances $N = 60\%$; adjust α respectively as $0.125, 0.25, 0.5, 0.75$.

α	**QI-Forest**	**R-Forest**
0.125	81.03	72.21
0.25	83.15	78.38
0.5	83.84	82.34
0.75	83.42	83.33

TABLE 4.3: QI Forest Classifiers vs. Random Forest Classifiers: $\alpha = 0.5$; training instances $N = 60\%$; adjust ensemble size T respectively as $3, 10, 30$.

T	**QI-Forest**	**R-Forest**
3	79.32	75.83
10	78.18	75.80
30	83.84	82.34

set no tree depth limit, and no minimum samples limit for splits and leaves. The default hyperparameter setting for forests is: ensemble size $T = 30$; select one half features to train base learners, which means $\alpha = 0.5$; the sub-sample size in Bagging is always the same as the original training sample size but the samples are drawn with replacements; and $N = 60\%$ samples are used as training data instances. Table 4.2 shares the default hyperparameter setting except that we set α respectively to $0.125, 0.25, 0.5$. In this experiment, we want discover how QI Forest is compared to Random Forest with variant α settings. Table 4.3 shares the default hyperparameter setting except that we set ensemble size T respectively as $3, 10, 30$. In this experiment, we want to discover how robustly QI Forest

TABLE 4.4: QI Forest Classifiers vs. Random Forest Classifiers: $\alpha = 0.5$; ensemble size $T = 30$; adjust training instances N respectively as $20\%, 30\%, 60\%$.

Training Instances	**QI-Forest**	**R-Forest**
60%	83.84	82.34
30%	79.52	75.32
20%	75.90	72.14

and Random Forest perform with small ensemble sizes. Table 4.4 shares the default hyperparameter setting except that we set training instances N respectively to $20\%, 30\%, 60\%$. In this experiment, we want to how robustly QI Forest and Random Forest solve small data problems.

Table 4.1 shows the advantage of QI Forest Classifiers in the default hyperparameter setting. QI Forest significantly outperforms Random Forest on one data set; QI Forest slightly outperform Random Forest five data sets; QI Forest perform slightly worse than Random Forest on two data sets; and QI Forest perform significantly worse than Random Forest on one data set. The experimental result supports that Quantum-Inspired Forest Classifiers can be compared to Random Forest Classifiers in general situation. Table 4.2 supports that QI Forest not only outperforms Random Forest in one setting of α, but also beat Random Forest with multiple α settings. We notice that the smaller α is, the larger the advantage of QI Forest is. Especially when we select only a small number of features for training base learners, QI Forest can outperform Random Forest significantly. Table 4.3 indicates that the performance difference of QI Forest and Random Forest increases as the ensemble size T decrease. It means QI Forest can perform significantly better than Random Forest with a limited forest size. Table 4.4 shows another advantage that QI Forest can solve small sample regression problems better than Random Forest. As we decrease training instances from 60% to 20%, the performance difference increases from 1.50 to 3.76. These experimental results show that given very limited computational resources or training data, QI Forest can be favorably compared to Random Forest.

4.3 Discussion and Conclusion

The experimental observations for regression forest are basically close to the observations in Chapter 3. Overall, similar patterns and conclusions can be found easily. However, there are also important differences. The performance advantage of Quantum-Inspired Forest Classifiers over Random Forest Classifiers is weaker compared to the advantage for regression tasks. The advantage of Quantum-Inspired Classification Forest is not significant enough. As we currently have solid theoretical analysis of Quantum-Inspired Classification Forest, we guess two reasons here. First, it's at least partly caused by the removal of Transition Probabilities. How to obtain Transition Probabilities for classification tasks remains to be studied. Second, another important cause is the fact that the theoretical analysis for ensemble regression cannot be applied to ensemble classification directly. A different theoretical mechanism may lead a big difference between ensemble regression and ensemble classification. We are trying to establish the theoretical analysis of ensemble classification based information theory.

In summary, we propose a novel algorithm named Quantum-Inspired Classification Forest in this chapter. The heuristical idea also comes from density matrix as well as Quantum-Inspired Regression Forest. The difference between Random Classification Forest and Quantum-Inspired Forest is that we select components according to the corresponding Fractional Probabilities rather than uniformly randomly. We also perform empirical analysis on 9 UCI data sets. Its performance can be compared to Random Forest. We also note that the essential difference between ensemble regression and ensemble classification. We need to find another way to organize theoretical analysis of Quantum-Inspired Classification Forest. If so, we may know how to improve the ensemble classification algorithms further. This is a future direction.

Chapter 5

A Physical Perspective

In this chapter, we present several valuable viewpoints on machine learning, particularly deep learning, based on a physical perspective. This seems to be a relatively isolated chapter. We don't present specific algorithms or methods here, but present a kind of physical perspective extremely different from the philosophy of traditional machine learning communities. Information is the essential bridge connecting modern physics and machine learning. We believe this will become a important topic that will attract much attention of both physicists and machine learners soon in the future. A new amazing research field is forming without doubt.

At present, we think most researchers in the deep learning community underestimate or even ignore the close relationship between physics and machine learning. But a small number of researchers with both physics and machine learning background have started to pay attention to this field. They are trying very hard to explain or study deep networks and particularly its theoretical foundation. And some interesting results have be proposed. We decide to present the multiple connections of deep neural networks and physics based on our best knowledge. We also review a few recent works on physics-inspired machine learning. It helps us understand how physics may interact with machine learning. Physical prior knowledge for machine learning is an important content in this chapter. An interesting thing is that physical prior knowledge has been generally applied to machine learning in some sense, but few researchers are systematically discussing it. This chapter provides a nice supplement.

5.1 From Mathematics to Biology

Why does deep and cheap learning work so well? The success of deep neural networks (DNNs) is mainly due to the three important characteristics. Firstly, DNNs are highly expressive [Montufar et al., 2014]. DNNs have the potential to approximate continuous functions given many enough neurons. Recent research argues that linear increasing depth are expressive approximately as exponentially increasing width [Eldan and Shamir, 2016]. Secondly, deep learning methods have strong generalization properties [Hardt et al., 2015]. The amazing generalization performance of deep networks requires deeper study [Zhang et al., 2016]. Thirdly, DNNs are usually relatively easy to train [Choromanska et al., 2015, Goodfellow et al., 2014]. And training deep networks are actually different from classical optimization problems somehow. They both try to minimize some loss functions. And the concern of optimization is always about minimizing its loss function as efficiently. But the quality of training deep networks is measured by both computational cost and generalization errors. The solutions that bring us low generalization error rather training error are consider good solution.

From a mathematical perspective, we think the first point is totally understandable. We are not surprised by the representation power of deep neural networks. Mathematically, the representation power is very reasonable. Given enough neurons, we can prove that DNNs can approximate any continuous objective function. So theoretically speaking, DNNs have the ability to represent any probabilistic distribution. Given same amounts of neurons, deep networks generally have better representation power than wide networks [Eldan and Shamir, 2016]. Nevertheless, the second point is very amazing and surprising. The third point is closely related to the second point, because the purpose of optimization is exactly to enhance the generalization performance of deep networks. In probability theory and statistics, we generally require at least 10 samples for estimating a model parameter. Terribly, in deep learning, models parameters are often much more than training samples. However, deep networks still work so well in test data sets. Why? The strange generalization ability is a big challenge of deep learning theory.

From a biological perspective, we try to find some good explanations. The key lies in the brain, particularly the architecture of neural network in the brain. Some researchers believe artificial intelligence is mathematics, while some other researchers

believe artificial intelligence is engineering. They both seems reasonable in special directions. But we strongly prefer the claim that artificial intelligence are science, because intelligence itself is a natural phenomenon that is consistent with the physical laws in the universe and the physical environment in the earth.

Brain scientists present that we have already known the bottom-level principle and architecture, though neural networks in brain are still quite complex for us. Neural systems in human brains share several essential architecture characteristics with DNNs. First, they both have deep hierarchical structures. Deep learning have many neurons, but the connections between neurons in one layer are forbidden. In human brains, the signals from one kind of cell are often transmitted into neurons with different functions. Second, local connections are preferred, while direct long-term connections between far neurons are very inefficient. This is not strange. In deep learning, only layer-wised connections are allowed, because full connections among all neurons can take terrible trouble when we train the deep networks. The training dynamics can become so complex and inefficient that we cannot train deep networks at an acceptable cost. At same time, layer-wise connections reduce the complexity of deep networks remarkably, and still preserve good representation power. Third, their architecture design are strongly driven by environments or data that obey physical laws. A good example is the close relation between convolution kernels and visual cortex. Convolution kernels take a biological inspiration from our visual system [Tovée, 1996]. The visual cortex has small regions of cells that are sensitive to specific visual regions. [Hubel and Wiesel, 1962] showed that some individual neurons in the brain only responded to edges of a certain orientation. For example, some neurons activate when exposed to vertical edges and some when shown horizontal or diagonal edges. This is exactly what a convolution kernel do. [Hubel and Wiesel, 1962] found out that all of these neurons were organized in a hierarchical architecture able to produce our visual perception. So we can find a clear biological basis behind Convolutional Neural Networks (CNNs). Human can do pretty well with the biological neural network architecture, so deep networks that employ a similar network architecture are also very likely to do well. We believe this biological prior is very useful prior knowledge, although some researchers believe that recent developments of deep learning have nothing to do with neuroscience or brains. After a long-term evolution, human brains have become very good at sensing and precessing the environment information. Or we had become extinct already. This is a kind of Anthropic Principle about intelligence and evolution. So the brain architecture is

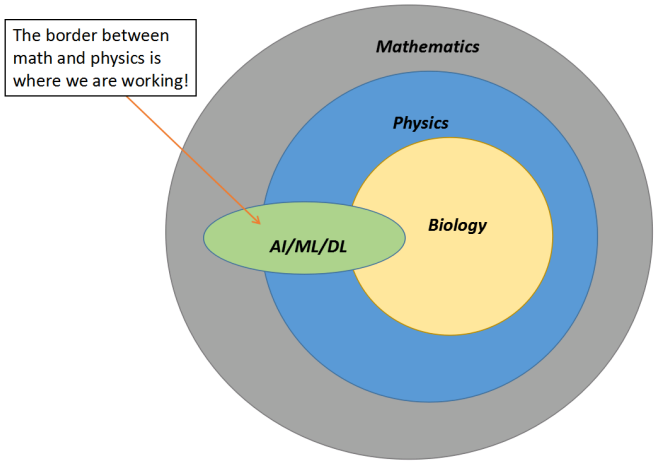

FIGURE 5.1: Intelligence: From Mathematics, to Physics, to Biology

a good prior architecture optimized by natural selections. Obviously, it becomes reasonable to consider DNNs happen to have a architecture similar to the human brain architecture.

5.2 Physical laws as prior knowledge

Biology is ruled by physics, while physics is ruled by mathematics. Physics is the essential bridge that connects mathematics and biology, and also the bridge that connects information and physical world. From a physical perspective, we can have a deeper understanding about learning and intelligence. We have shown that the architecture design of natural intelligence is good biological prior for artificial intelligence. The next question is, why do CNNs well? Mathematically, CNNs reduced weight parameters, and achieve low-level feature detection at a relatively low cost. Biologically, CNNs have a similar architecture as our visual system. But we also know that reducing model complexity and architecture similarity to the brain don't necessarily. We can find a deeper physical foundation that connects the mathematical explanation and the biological explanation.

A key perspective is that intelligence is designed for solving problems in the physical space rather than a mathematical space. How to discriminate the physical space and the mathematical space. Again, we take computer vision as an example. A representative sample in the physical space looks like Figure 5.2 and Figure 5.3. And a representative sample in the mathematical space looks like Figure 5.4, almost like only noises. The key lies in here. The mathematical space

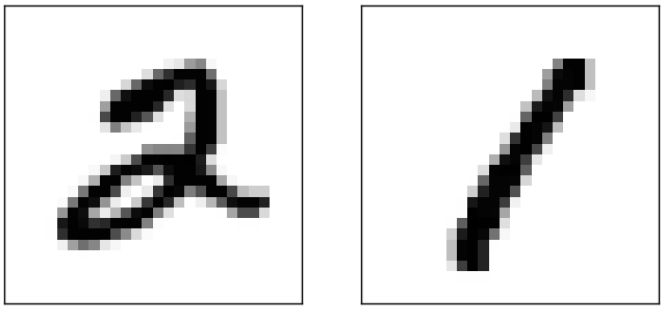

FIGURE 5.2: MNIST: Handwritten Digits Recognition. Each image has 28×28 pixels and a label ranging from 0 to 9.

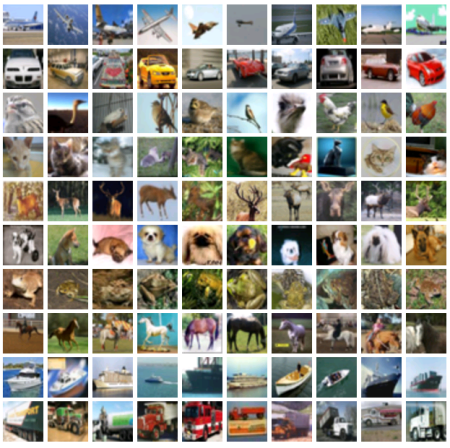

FIGURE 5.3: Random samples in physical space. Cifar-10.

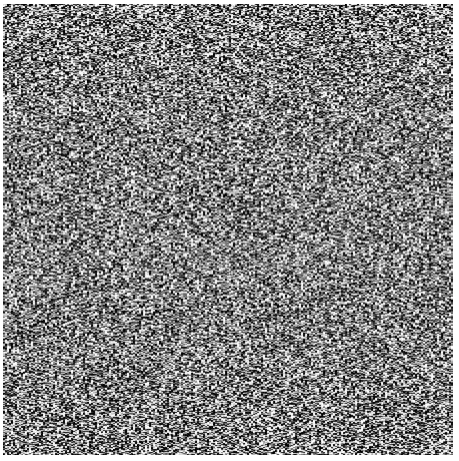

FIGURE 5.4: A random sample in mathematical space.

is too large to be compared with the physical space. The probability of which a random sample from the mathematical space also appears in the physical space is nearly 0. When we build artificial intelligence, we should only focus on physically possible circumstances. This does be what natural intelligence is doing. And if we translate those physically possible circumstances in the language of machine learning, we will get physical prior knowledge for machine learning.

Actually, two physical laws matter a lot in the situation of CNNs. One is Space Translation Symmetry, the other is the principle of locality in classical physics. What is Space Translation Symmetry? In physical language, Space Translation Symmetry is the invariance of physical laws under space translation. In daily language, if we move a cat from one location to another location, the cat is still a cat. It's also true for cats in a picture. In the language of machine learning, if a convolution kernel is useful for visual detection in one location, the convolution kernel should also be useful for visual detection in any other locations. This is a physical principle but not necessarily a physical principle. What is the principle

of locality? In physical language, an object only directly interacts with its immediate surroundings, and Einstein's "Spooky Action at a Distance" is forbidden in classical physics. In daily language, a living cat cannot be separated into parts at multiple locations. Again, it's also true for cats in a picture. In the language of machine learning, we don't need detect the pattern or knowledge of distant locations, and small convolutions kernels that detect local features can work well. We also need note that the principle of locality is not true in quantum physics. The entanglement at a distance is not rare for quantum systems. If we train deep networks to solve a quantum physical problem, such as predicting the free energy, the localized convolution kernel will harm the generalization performance.

We emphasis that although physical prior knowledge can be learned from training data, the learning takes more training data and computational time. Worse, the model has higher risk of overfitting and knowledge get lost easily due to new data flows. Hard encoding should be a better way to encoding physical prior knowledge. The prior knowledge of two physical laws is hard encoded in the network architecture by adding convolution layers. And data knowledge is soft encoded by training network weights. Hard encoded learning requires less resources and is lasting, while soft encoded learning requires more resources and is adaptive. Proper hard encoding can further reduce the model complexity and avoid overfitting. We should encode physical laws into model architecture in a hard encoding way as much as possible. A recent work [Stewart and Ermon, 2017] provides a nice example of hard encoding physical laws of motion into deep learning. This prior knowledge of motion laws successfully relax the requirement for training data size and get good experimental results.

5.3 A Statistical Physical Perspective

In very early period, [Jaynes, 1957] discussed the topic of statistical physics and information theory. This is a wonderful pioneer work that pointed us a bright direction. Statistical physics is more than a physical theory, and also an advanced mathematical framework that uses methods of probability theory and statistics, and deals with large populations and approximations. It is good at describing a wide variety of fields with an inherently stochastic nature. Information or information theory is a natural bridge between statistical physics and statistics/machine learning. Some researchers think Maximum Likelihood Estimation and Bayesian

Inference can be regarded special cases of statistical physics. But, overall speaking, statistical physical learning is still at the very beginning phase. We have obtained some interesting results when we try to machine learning in the statistical physical framework. We will present these works in near future.

5.3.1 Minimum Hamiltonian Estimation

We only present an interesting preliminary result in this subsection. We successfully propose Minimum Hamiltonian Estimation which is equivalent to Maximum Likelihood Estimation (MLE) according to our theoretical analysis. We also prove that the sample size for training in machine learning is exactly equal to the inverse temperature in thermodynamics.

Suppose there is a training sample set $\{x\}$ of m independent and identically distributed observations, coming from a distribution with an unknown probability density function $p(x)$. It is however surmised that the density function $p(x)$ belongs to a certain family of distributions $\{p(x|\theta), \theta \in \Theta\}$, where θ is a vector of parameters for this family, called the parametric model. We decide to estimate θ by maximizing $p(\theta|\{x\})$, written as

$$\theta_0 = \operatorname{argmax} p(\theta|\{x\}) = \frac{p(\{x\}|\theta)}{\int D(\theta)p(\{x\}|\theta)d\theta}, \tag{5.1}$$

where $D(\theta)p(\{x\}|\theta)$ is a normalization factor, $D(\theta)$ is the density function of θ. So we also have

$$\theta_0 = \operatorname{argmax} p(\{x\}|\theta). \tag{5.2}$$

As

$$\begin{aligned} p(\{x\}|\theta) &= \prod_{x^{(i)}\in\{x\}} p(x^{(i)}|\theta) \\ &= \prod_{x^{(i)}\in\{x\}} e^{\ln p(x^{(i)}|\theta)} \\ &= e^{\sum_{x^{(i)}\in\{x\}} \ln p(x^{(i)}|\theta)} \\ &= e^{-m(-\sum_{i=1}^{m} p(x^{(i)}) \ln p(x^{(i)}|\theta))} \\ &= e^{-mH(\theta)}, \end{aligned} \tag{5.3}$$

the likelihood $l(\theta) = \ln p(\{x\}|\theta) = -mH(\theta)$. We define $H(\theta) = -\sum_{i=1}^{m} p(x^{(i)}) \ln p(x^i|\theta)$ as the Hamiltonian. According to the analysis above, we have successfully transformed Maximum Likelihood Estimation into a new equivalent estimation method, named Minimum Hamiltonian Estimation,

$$\theta_0 = \operatorname{argmin} H(\theta). \tag{5.4}$$

And it is a very interesting approach to considering Maximum Likelihood Estimation and Minimum Hamiltonian Estimation in a statistical physical framework. Maximum Likelihood Estimation is a process that a thermodynamical system reaches its ground state. And the optimization algorithm decide how the thermodynamical system evolves. In thermodynamics, the probability of a state s with a energy of $E(s)$

$$p(s) = \frac{e^{-\beta E}}{Z}, \tag{5.5}$$

where $Z = \sum_a e^{-\beta E(a)}$ is called the partition function and $\beta = \frac{1}{T}$ is called the inverse temperature. Let's compare Equation 5.3 with Equation 5.5. If we analyze MLE in a statistical physical framework, we can easily notice that the training sample size m is exactly equal to the inverse temperature β. If the training sample size m is very large, we can regard the learning model as a low-temperature system. In the limit of absolute zero $T \to 0$, the low-temperature system certainly tends to occupy the ground state. In this case, minimizing its hamiltonian is a very good method modeling the low-temperature system, which indicates good generalization performance in language of machine learning. If the training sample size m is very small, we can regard the learning model as a high-temperature system. In the limit of high temperature $T \to +\infty$, the high-temperature system tends to occupy all possible state uniformly. In this case, minimizing its hamiltonian is bad for modeling the high-temperature system, which indicates bad generalization performance in language of machine learning.

See. We have got a very reasonable physical quantity describing the generalization performance. Maximum Likelihood Estimation and Minimum Hamiltonian Estimation use the ground state to model the data set $\{x\}$. But a system only occupy the ground state in a probability of $\frac{e^{-\beta E_{min}}}{Z}$. So the algorithm only works in case of low temperature. More importantly, the concept of hamiltonian and temperature has been introduced into machine learning naturally.

Chapter 6

Conclusion

In this thesis, we have introduced several physical concepts in ensemble learning, and proposed a physics-inspired machine learning algorithm, Quantum-Inspired Forest. The physical perspective on machine learning reveals an important new approach to advancements of machine learning. We summarize the main contributions of this thesis in the following.

In Chapter 3, we mainly propose two-fold contribution. First, we propose a novel ensemble method named Quantum-Inspired Subspace and Quantum-Inspired Regression Forest. Quantum-Inspired Subspace can be easily applied to diversified base learners and combined with other classical ensemble methods, such as Bagging. We incorporate Quantum-Inspired Subspace into Random Forest and propose Quantum-Inspired Forest. The additional computational cost is very cheap, equivalent to the cost of full-rank PCA preprocessing. Second, we propose quantum interpretations for several machine learning concepts, and successfully establish a theoretical bridge between quantum interpretations and ensemble learning. In future research, we consider two directions interesting. The first direction is to study Quantum-Inspired Subspace under complex circumstances, particularly classification tasks. The second direction is discover deeper theoretical connections between quantum mechanics and machine learning algorithms. May the mechanism of quantum entanglement be helpful for machine learning in some way? We also believe it will be very valuable to theoretically analyze quantum interpretations for neural networks.

In Chapter 4, we propose a novel algorithm named Quantum-Inspired Classification Forest in this chapter. The heuristical idea also comes from density matrix

as well as Quantum-Inspired Regression Forest. The difference between Random Classification Forest and Quantum-Inspired Forest is that we select components according to the corresponding Fractional Probabilities rather than uniformly randomly. We also perform empirical analysis on 9 UCI data sets. Its performance can be compared to Random Forest. We also note that the essential difference between ensemble regression and ensemble classification. We need to find another way to organize theoretical analysis of Quantum-Inspired Classification Forest. If so, we may know how to improve the ensemble classification algorithms further.

In Chapter 5, we mainly present a physical perspective on deep neural networks. How can machine learning interact with theoretical physics? We carefully discuss several valuable viewpoints on neural networks based on a physical perspective. We believe this will become a important topic that will attract much attention of both physicists and machine learners soon in the future. This is absolutely a forming research field. And it's very meaningful to discuss several valuable points at this early beginning of this new field.

In the future, we will care about two approaches. The first one is we plan to completed the theoretical analysis of Quantum-Inspired Classification Forest in a information theoretical way, and study the empirical performance of Quantum-Inspired Forest under complex circumstances. The error decomposition method of ensemble classification is incomplete now, but information entropy and mutual information of features may explain or even improve Quantum-Inspired Subspace further. The second approach, the most important one in our opinion, is applying the method of quantum physics and statistical physics to machine learning and neural networks. Not only deep networks can help physicists solve analytically hard problems, but also physical methods can be very helpful for deep networks.

Bibliography

Amara, S. G. and Kuhar, M. J. (1993). Neurotransmitter transporters: recent progress. *Annual review of neuroscience*, 16(1):73–93.

Blacoe, W., Kashefi, E., and Lapata, M. (2013). A quantum-theoretic approach to distributional semantics. In *HLT-NAACL*, pages 847–857.

Bohachevsky, I. O., Johnson, M. E., and Stein, M. L. (1986). Generalized simulated annealing for function optimization. *Technometrics*, 28(3):209–217.

Boixo, S., Rønnow, T. F., Isakov, S. V., Wang, Z., Wecker, D., Lidar, D. A., Martinis, J. M., and Troyer, M. (2014). Evidence for quantum annealing with more than one hundred qubits. *Nature Physics*, 10(3):218–224.

Breiman, L. (1984). *Classification and regression trees*. Chapman & Hall/CRC.

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Brooke, J., Bitko, D., Aeppli, G., et al. (1999). Quantum annealing of a disordered magnet. *Science*, 284(5415):779–781.

Brown, G., Wyatt, J., Harris, R., and Yao, X. (2005a). Diversity creation methods: a survey and categorisation. *Information Fusion*, 6(1):5–20.

Brown, G., Wyatt, J. L., and Tiňo, P. (2005b). Managing diversity in regression ensembles. *Journal of Machine Learning Research*, 6(Sep):1621–1650.

Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B., and LeCun, Y. (2015). The loss surfaces of multilayer networks. In *Artificial Intelligence and Statistics*, pages 192–204.

Cortes, C. and Vapnik, V. (1995). Support vector machine. *Machine learning*, 20(3):273–297.

Cutler, A. and Zhao, G. (2001). Pert-perfect random tree ensembles. *Computing Science and Statistics*, 33:490–497.

Das, A. and Chakrabarti, B. K. (2008). Colloquium: Quantum annealing and analog quantum computation. *Reviews of Modern Physics*, 80(3):1061.

Dong, D., Chen, C., Chu, J., and Tarn, T.-J. (2012). Robust quantum-inspired reinforcement learning for robot navigation. *IEEE/ASME Transactions on Mechatronics*, 17(1):86–97.

Eldan, R. and Shamir, O. (2016). The power of depth for feedforward neural networks. In *Conference on Learning Theory*, pages 907–940.

Fan, W., McCloskey, J., and Yu, P. S. (2006). A general framework for accurate and fast regression by data summarization in random decision trees. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 136–146. ACM.

Farhi, E., Goldstone, J., Gutmann, S., Lapan, J., Lundgren, A., and Preda, D. (2001). A quantum adiabatic evolution algorithm applied to random instances of an np-complete problem. *Science*, 292(5516):472–475.

Fawagreh, K., Gaber, M. M., and Elyan, E. (2014). Random forests: from early developments to recent advancements. *Systems Science & Control Engineering: An Open Access Journal*, 2(1):602–609.

Finnila, A., Gomez, M., Sebenik, C., Stenson, C., and Doll, J. (1994). Quantum annealing: a new method for minimizing multidimensional functions. *Chemical physics letters*, 219(5):343–348.

Geurts, P., Ernst, D., and Wehenkel, L. (2006). Extremely randomized trees. *Machine learning*, 63(1):3–42.

Goodfellow, I. J., Vinyals, O., and Saxe, A. M. (2014). Qualitatively characterizing neural network optimization problems. *arXiv preprint arXiv:1412.6544*.

Hardt, M., Recht, B., and Singer, Y. (2015). Train faster, generalize better: Stability of stochastic gradient descent. *arXiv preprint arXiv:1509.01240*.

Heim, B., Rønnow, T. F., Isakov, S. V., and Troyer, M. (2015). Quantum versus classical annealing of ising spin glasses. *Science*, 348(6231):215–217.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence*, 20(8):832–844.

Huang, H., Qin, H., Yoo, S., and Yu, D. (2012). A new anomaly detection algorithm based on quantum mechanics. In *2012 IEEE 12th International Conference on Data Mining*, pages 900–905. IEEE.

Hubel, D. H. and Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154.

Jaynes, E. T. (1957). Information theory and statistical mechanics. *Physical review*, 106(4):620.

Johnson, D. S., Aragon, C. R., McGeoch, L., and Schevon, C. (1992). Optimization by simulated annealing: an experimental evaluation; part iii, the traveling salesman problem. *Opns. Res.*

Johnson, D. S., Aragon, C. R., McGeoch, L. A., and Schevon, C. (1989). Optimization by simulated annealing: an experimental evaluation; part i, graph partitioning. *Operations research*, 37(6):865–892.

Johnson, D. S., Aragon, C. R., McGeoch, L. A., and Schevon, C. (1991). Optimization by simulated annealing: an experimental evaluation; part ii, graph coloring and number partitioning. *Operations research*, 39(3):378–406.

Kadowaki, T. and Nishimori, H. (1998). Quantum annealing in the transverse ising model. *Physical Review E*, 58(5):5355.

Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., et al. (1983). Optimization by simmulated annealing. *science*, 220(4598):671–680.

Krogh, A., Vedelsby, J., et al. (1995). Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, 7:231–238.

Lanting, T., Przybysz, A., Smirnov, A. Y., Spedalieri, F., Amin, M., Berkley, A., Harris, R., Altomare, F., Boixo, S., Bunyk, P., et al. (2014). Entanglement in a quantum annealing processor. *Physical Review X*, 4(2):021041.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

Leifer, M. S. and Poulin, D. (2008). Quantum graphical models and belief propagation. *Annals of Physics*, 323(8):1899–1946.

Lichman, M. (2013). UCI machine learning repository.

Liu, F., Ting, K., and Fan, W. (2005). Maximizing tree diversity by building complete-random decision trees. *Advances in Knowledge Discovery and Data Mining*, pages 350–368.

Martoňák, R., Santoro, G. E., and Tosatti, E. (2002). Quantum annealing by the path-integral monte carlo method: The two-dimensional random ising model. *Physical Review B*, 66(9):094203.

Melucci, M. and van Rijsbergen, K. (2011). Quantum mechanics and information retrieval. In *Advanced topics in information retrieval*, pages 125–155. Springer.

Messenger, R. and Mandell, L. (1972). A modal search technique for predictive nominal scale multivariate analysis. *Journal of the American statistical association*, 67(340):768–772.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533.

Montufar, G. F., Pascanu, R., Cho, K., and Bengio, Y. (2014). On the number of linear regions of deep neural networks. In *Advances in neural information processing systems*, pages 2924–2932.

Morita, S. and Nishimori, H. (2007). Convergence of quantum annealing with real-time schrödinger dynamics. *Journal of the Physical Society of Japan*, 76(6):064002.

Nielsen, M. A. and Chuang, I. L. (2010). *Quantum computation and quantum information*. Cambridge university press.

Provost, S. B. and Rudiuk, E. M. (1994). The exact density function of the ratio of two dependent linear combinations of chi-square variables. *Annals of the Institute of Statistical Mathematics*, 46(3):557–571.

Rédei, M. and Summers, S. J. (2007). Quantum probability theory. *Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics*, 38(2):390–417.

Santoro, G. E., Martoňák, R., Tosatti, E., and Car, R. (2002). Theory of quantum annealing of an ising spin glass. *Science*, 295(5564):2427–2430.

Sarandy, M. S., Wu, L.-A., and Lidar, D. (2004). Consistency of the adiabatic theorem. *Quantum Information Processing*, 3(6):331–349.

Schuld, M., Sinayskiy, I., and Petruccione, F. (2015). An introduction to quantum machine learning. *Contemporary Physics*, 56(2):172–185.

Shin, S. W., Smith, G., Smolin, J. A., and Vazirani, U. (2014). How" quantum" is the d-wave machine? *arXiv preprint arXiv:1401.7087*.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489.

Stewart, R. and Ermon, S. (2017). Label-free supervision of neural networks with physics and domain knowledge. In *AAAI*, pages 2576–2582.

Stoudenmire, E. M. and Schwab, D. J. (2016). Supervised learning with quantum-inspired tensor networks. *arXiv preprint arXiv:1605.05775*.

Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708.

Tovée, M. J. (1996). *An introduction to the visual system*. Cambridge University Press.

Ueda, N. and Nakano, R. (1996). Generalization error of ensemble estimators. In *Neural Networks, 1996., IEEE International Conference on*, volume 1, pages 90–95. IEEE.

Von Neumann, J. (1927). Wahrscheinlichkeitstheoretischer aufbau der quantenmechanik. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, 1927:245–272.

Wall, M. E., Rechtsteiner, A., and Rocha, L. M. (2003). Singular value decomposition and principal component analysis. In *A practical approach to microarray data analysis*, pages 91–109. Springer.

Weinstein, M. and Horn, D. (2009). Dynamic quantum clustering: A method for visual exploration of structures in data. *Physical Review E*, 80(6):066117.

Wittek, P. (2014). *Quantum machine learning: what quantum computing means to data mining*. Academic Press.

Wolf, L. (2006). Learning using the born rule.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*.

Zhou, Z.-H. (2012). *Ensemble methods: foundations and algorithms*. CRC press.

Publications

1. Zeke Xie, Issei Sato. A Quantum-Inspired Ensemble Method and Quantum-Inspired Forest Regressors. The 9th Asian Conference on Machine Learning (ACML 2017).