โครงการ

การเรียนการสอนเพื่อประสบการณ์

ชื่อโครงการ มุมมองที่คล้ายกันของโครงข่ายตกค้างและสมการเชิงอนุพันธ์ย่อย

Analogous View Between Residual Networks and Partial Differential Equations

ชื่อนิสิต นายวิชยุตม์ อิ่มจิตร **เลขประจำตัว** 6033435523

ภาควิชา ฟิสิกส์

ปีการศึกษา 2563

คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ANALOGOUS VIEW BETWEEN RESIDUAL NETWORKS AND PARTIAL DIFFERENTIAL EQUATIONS

Mr. Vichayuth Imchitr

A report submitted to the Department of Physics of Chulalongkorn
University in partial fulfillment of the requirements for the degree of
Bachelor of Science in Physics
Academics Year 2020

Project Title	Analogous View Between Residual Networks and Partial Differential Equations
By	Vichayuth Imchitr
Field of Study	Physics
Project Advisor	Assoc.Prof.Dr.Udomsilp Pinsook
Academic Year	2020

This report is submitted to the Department of Physics, Faculty of Science, Chulalongkorn University, in partial fulfillment of the requirements for the degree of Bachelor of Science.

This report has been approved by the committee:

……………….. Chairman

(Assoc.Prof.Dr.Nakorn Phaisangittisakul)

……………….. Project Advisor

(Assoc.Prof.Dr.Udomsilp Pinsook)

……………….. Committee

(Dr.Thiparat Chotibut)

Project Title	Analogous View Between Residual Networks and Partial Differential Equations
By	Vichayuth Imchitr
Field of Study	Physics
Project Advisor	Assoc.Prof.Dr.Udomsilp Pinsook
Academic Year	2020

Abstract

In this project we try to explain dynamic of hidden unit in residual network with knowledge of partial differential equation and quantum mechanics. By consider the change of input information over one layer of residual block as dynamics over a unit of time and consider each tensor element of input as function in the position space we see that the residual block is mathematical equivalence to partial differential equation depend on time and position, this allows us to derive Hamiltonian-like object describe dynamics of the input in the similar fashion to the Schrödinger equation. We also show that output from the hidden layer of residual network can be write as sum of contribution from all paths the input has travelled in the previous layer like Feynman path integral formulation of quantum mechanics. The experiment of neural network architecture from PDE is created and compared with residual network in the case that residual block is made up of skip connection over one convolution kernel. Example of further development of neural network architecture from mathematical understanding of residual block is indicated here as neural ordinary differential equation.

Acknowledgement

I would like to express my gratitude to my project advisor Assoc.Prof.Dr.Udomsilp Pinsook for the precious advice, being supportive and attentively take care of me through the time of research and writing this thesis. Secondly, I would like to thank the committees of this project Assoc.Prof.Dr.Nakorn Phaisangittisakul and Dr.Thiparat Chotibut for helpful critics and question which helps me improve my understanding on the topic of my research and advice help me on my way of this project. Thirdly, I thank the department of Physics, Faculty of science, Chulalongkorn university for financial support of this project. Finally, I would like to thank my family and friends for supporting and encouraging me through the entire time.

Content

CHAPTER I

INTRODUCTION

Artificial neural network is a machine learning algorithm inspired by biological neural network located in animal brain. By receiving a set of observational data $\{\vec{X}, \vec{y}\}$ where $\vec{X}$ is a matrix of independent variable and $\vec{y}$ is vector of dependent variables, the network itself can learn relation between each pair of X_i, y_i and predict $\vec{y}$ when unseen $\vec{X}$ is given. This fascinating ability cause neural network to be used in various tasks.

Activity of artificial neural network starts by receiving dataset into the network, next the dataset is divided into two groups: training dataset $\mathcal{D}_{train}$ and test dataset $\mathcal{D}_{test}$, training dataset is used to create model $f(\vec{X};\vec{\theta})$ which is function of $\vec{X}$ and parameters $\vec{\theta}$ used in prediction of $\vec{y}$. The model is created by passing the operation of parameters into some activation function. Then the performance of network is evaluated using cost function $\mathcal{L}(\vec{y}, f(\vec{X};\vec{\theta}))$, the function tells us how $f(\vec{X};\vec{\theta})$ different from the ground truth $\vec{y}$. Learning of the network is to minimizing cost function of training dataset $\mathcal{L}(\vec{y}_{train}, f(\vec{X}_{train};\vec{\theta}))$ by adjusting value of $\vec{\theta}$s, usually by back propagation in wish gradient of loss function with respect to each parameter is calculated, then it used to adjust the learning parameter in the way that value of loss function is decreased. After the network is finished learning final evaluation of network is performed by calculating loss function of test dataset $\mathcal{L}(\vec{y}_{test}, f(\vec{X}_{test};\vec{\theta}))$ [1].

At the present time, numerous architectures of neural network are proposed with goal of improve performance of neural network on vast majorities of tasks, the one that caught our attention is the network architecture called residual neural network (ResNet) which its hidden layer made up of residual block, the network shows high performance on image classification task in term of reduce error caused by vanishing gradient which drops accuracy of network as the architecture goes deeper.

1.1 Brief idea of residual network

What make residual network special is the structure that build up the hidden layer called residual block, residual block consists of two components: convolution kernel and skip connection.

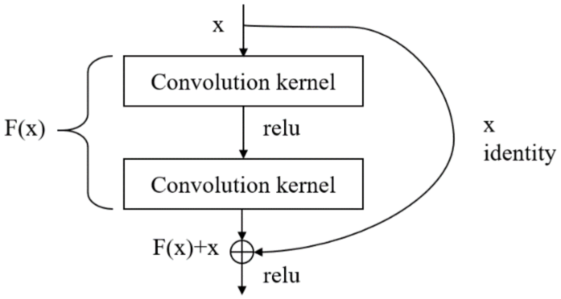


Figure 1.1 Residual block

Convolution kernel is a tensor of parameter called weight w, the kernel is overlay on the tensor of input data and usually smaller than the input. Convolution kernel work by convolve the element of tensor $u(x,t)$ in the interception region $x \in [-a,a]$ and create an element of activation map $u(x,t+1)$ as:

$$u(x,t+1) = \sum_{dx=-a}^{dx=a} w(dx)u(x+dx) = w(x) * u(x,t) \tag{1}$$

Then further process by slide along the width of input data with fixed stride to the right and hops down to the left-end of tensor with same stride value, the process repeated until entire tensor is process, completing the activation map [2]. The use of convolution kernel can be modified by adjust hyperparameters like stride which controls step the kernel move, or zero-padding which create zero value boarder on the input tensor changing size of activation map on the progress.

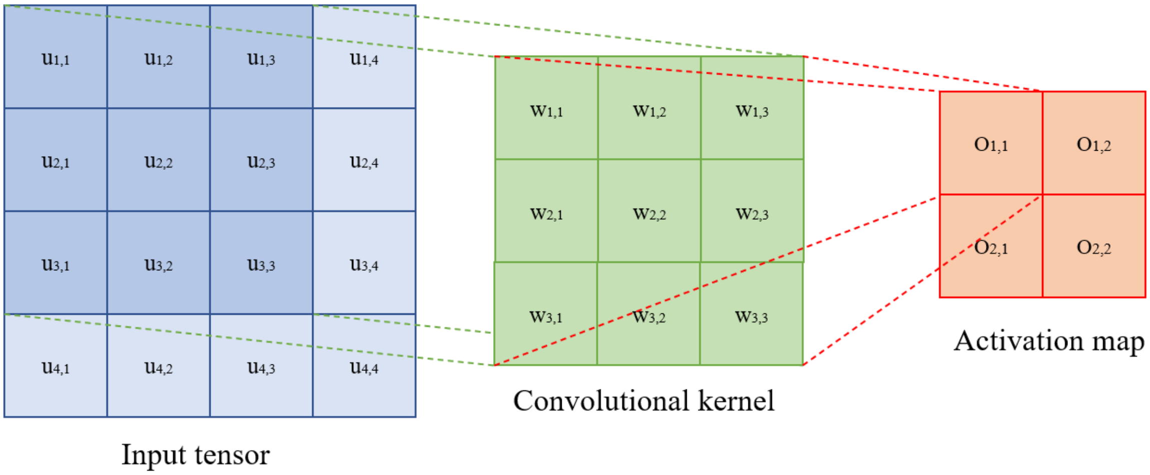
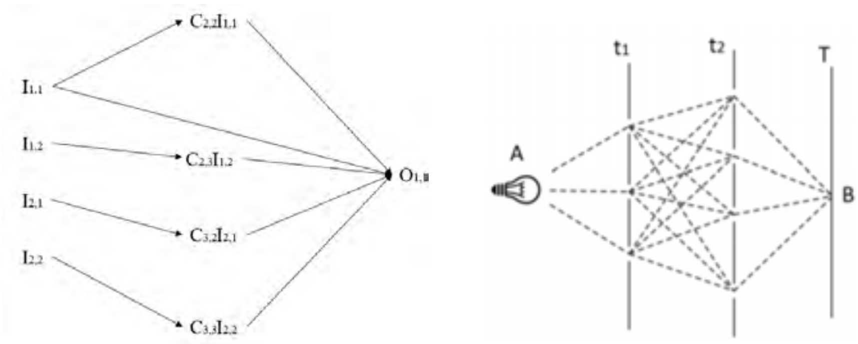


Figure 1.2 The process of 2-dimensional convolutional kernel on the 2-dimensional input tensor, the kernel creates first element of activation map $O_{1,1}$.

With proper value of weight, convolution kernel can be used in classification task of neural network by help the network extract high-level features of data i.e., features that display the characteristic of $\vec{X}$ which help the network to determine its label $\vec{y}$ [2], hence enhance the performance of classification.

The network of convolutional kernel is the architecture of hidden layer of convolutional neural network (CNN) which create model by create activation map and processing with activation function to make output of the layer then repeated until the final layer, before passing into traditional fully connected ANN. The network shows high performance on image classification task [3]. However, as hidden layer of network goes deeper, the accuracy of prediction become saturated and suddenly drop in some point. The degradation of accuracy is caused by vanishing gradient where gradient with respect to weight at the previous layer is decreased due to the chain rule multiplication of gradients until it vanished eventually [4]. Skip connection of ResNet can solve such problem by adding elements of tensor from previous layer to activation map of current layer hence higher value of gradient is obtained [5]. The strategy to prevent vanishing gradient is sometimes include applying the suitable activation function like rectified linear unit (ReLU) which return absolute value of the activation map element if it has value more than zero and return zero if the input element is negative.

1.2 Motivation



(a) (b)

Figure 1.4: (a) The path of input information caused by residual block, the output information is seen as sum of contribution from each path, (b) path of photon in slit experiment.

To design further neural network architecture from the previous version, the mathematical knowledge of how input tensor changed as it passes through the hidden layer of network is strongly required. By comparison we see the implicit similarity between the dynamics of hidden state through the hidden layer and particle in quantum mechanics such as photon in slit experiment where the final output comes from summation of contributes from each path it was travelled. This implicit similarity gives up a question: “Is it possible to explain the non-physical phenomena with knowledge of physics?” in which we strongly believe that it is possible.

1.3 Objective and scope of this project

In this project we aim to explain the dynamics of the hidden state of residual network with knowledge of non-relativistic quantum mechanics pointing similarity between formulation of non-relativistic quantum mechanics and structure of residual network. In chapter II we review the formulation of quantum mechanics Schrödinger equation, discuss its special case where Hamiltonian is time-independent and expand to Feynman path integral formulation. In chapter III the structure of residual block is shown to be equivalent to the partial differential equation (PDE) and define Hamiltonian-like object which cause dynamics of information in the hidden layer and finally write output from hidden layer of residual network in form of path integral using knowledge of Fourier transformation and Hamiltonian-like object we obtained.

CHAPTER II

FORMULATION OF NON-RELATIVISTIC QUANTUM MECHANICS

2.1 The Schrödinger equation

Time evolution of a quantum system is described by the Schrödinger equation:

$$i\hbar\frac{\partial\psi(x,t)}{\partial t} = \widehat{H}\psi(x,t) = -\frac{\hbar^2}{2m}\frac{\partial^2\psi(x,t)}{\partial x^2} + V\psi(x,t) \tag{2}$$

Where the solution of (2) $\psi(x,t)$ is called wave function which contain all information about the system [6]. In case that Hamiltonian operator $\widehat{H}$ dose not explicitly depend on time, wave function can be calculated by the method of separable solution [7]. With this method (2) can be separate into two independent ordinary differential equation (ODE).

$$i\hbar\frac{d\varphi(t)}{dt} = E \tag{3}$$

$$-\frac{\hbar^2}{2m}\frac{\partial^2\chi(x)}{\partial x^2} + V\chi(x) = E\chi(x) \tag{4}$$

By solving (3) we have $\varphi(t) = e^{-\frac{iEt}{\hbar}}$ while (4) depending on what form of Hamiltonian is the Hamiltonian dependent ODE is called time-dependent Schrödinger equation (TISE). The wave function is the product of solution from the two ODEs, so we may see time evolution of quantum particle subjected to the time-dependent Hamiltonian is determined by the exponential $e^{-\frac{iEt}{\hbar}}$. This type of Schrödinger equation is shown to be equivalent to Feynman path integral formulation in the original paper of the formulation and as we will see in the next few sections. The wave function can be considered as vector in Hilbert space of position that interchangeable with Hilbert space of momentum where momentum can be seen as real number operator p while position is instead considered as some derivative operator. The interchange of two Hilbert spaces can be done by applying inverse Fourier transformation and Fourier transformation.

$$\phi(p) = \frac{1}{\sqrt{2\pi\hbar}}\int_{-\infty}^{\infty}\psi(x,0)e^{-ipx/\hbar}dx \iff \psi(x,0) = \frac{1}{\sqrt{2\pi\hbar}}\int_{-\infty}^{\infty}\phi(p)e^{ipx/\hbar}dx \tag{5}$$

The use of Fourier transformation is useful in various case, in particular the case that particle in considered to be free particle in which calculation in position space cause paradox [6].

2.2 The Dirac notation

In Dirac notation, quantum mechanics is described using abstract object $|\quad\rangle$ serves as vector in a Hilbert space [8] and its complex conjugate defined as ket $\langle\quad|$ and the inner product between two vectors in Hilbert space is called bracket written as $\langle\quad|\quad\rangle$. In comparison to previous formulation the basis vector ψ in Hilbert space can be write in form of $|\psi\rangle$.

The inner product of the vector and basis vector of position space of momentum space create wave function in the Hilbert space.

$$\psi(x) = \langle x|\psi\rangle$$
$$\tilde{\psi}(p) = \langle p|\psi\rangle \tag{6}$$

And inner product between two bases represents inverse Fourier transformation and Fourier transformation.

$$\langle x'|x\rangle = \frac{1}{\sqrt{2\pi\hbar}} e^{\frac{ip(x'-x)}{\hbar}} = \delta(x'-x)$$
$$\langle p'|p\rangle \frac{1}{\sqrt{2\pi\hbar}} e^{\frac{ix(p'-p)}{\hbar}} = \delta(p'-p)$$
$$\langle x|p\rangle = \frac{1}{\sqrt{2\pi\hbar}} e^{\frac{ipx}{\hbar}} \tag{7}$$
$$\langle p|x\rangle = \frac{1}{\sqrt{2\pi\hbar}} e^{-\frac{ipx}{\hbar}}$$

The notation also contains completeness relation:

$$\int_{-\infty}^{\infty} dx|x\rangle\langle x| = \mathbb{I}$$
$$\int_{-\infty}^{\infty} dp|p\rangle\langle p| = \mathbb{I} \tag{8}$$

With this notation, we can rewrite (2) as

$$i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle = \hat{H}|\psi(t)\rangle \tag{9}$$

2.2 Feynman Path Integral

In 1948, Richard Feynman proposed a new formulation of quantum mechanics by considering the probability that particle will occupy each trajectory in space, with this idea he thinks of wavefunction as probability amplitude that particle is in a space-time point (x,t) which obtained from the integration of all contributions from possible paths the particle subjected to in the past, where those contributions are the same in amplitude but their phase are depend on the classical action of the path [9]:

$$\psi(x,t_f) = \int_{x_{path}} dx_{path} e^{\left[\frac{i}{\hbar}S_{path}\right]} \psi(x',t_i) \tag{10}$$

The formulation also able to be derived from the previous formulations, as we mention earlier this can be done in the case of time-independent Hamiltonian. In his original paper, Feynman derive

TISE from (10) with some approximations [9]. The derivation of (10) from previous formulation can be done by consider solution of TISE.

As we mentioned earlier, time-evolution of wave function is dictated by the exponential $e^{-\frac{iEt}{\hbar}}$, with this fact we can write time evolution of wave function as

$$\begin{aligned} \psi(x,t) &= e^{-\frac{i}{\hbar}t\hat{H}}\psi(x,0) \\ &= \left\langle x\middle|e^{-i(t)\hat{H}}\middle|\psi(0)\right\rangle \end{aligned} \tag{11}$$

By applying completeness relation in position basis

$$\begin{aligned} \psi(x,t) &= \int dx \left\langle x\middle|e^{-i(t)\hat{H}}\middle|x'\right\rangle\langle x'|\psi(0)\rangle \\ &= \int dx \left\langle x\middle|e^{-i(t)\hat{H}}\middle|x'\right\rangle\psi(x',0) \end{aligned} \tag{12}$$

We see that $\psi(x,t)$ becomes the sum of all products between the initial wave function and the propagator $\left\langle x\middle|e^{-i(t)\hat{H}}\middle|x'\right\rangle$ which specify the probability amplitude that particle that was in the point $(x',0)$ to the point (x,t). To specify what exactly the propagator is we might want to simplify the exponential of Hamiltonian since two component operators of Hamiltonian $\hat{H} = \hat{T} + \hat{V}$ depend on different variables which cause complexity in calculation, to cope with such complexity we may wish to separate the exponential of Hamiltonian into the term of $\hat{T}$ and $\hat{V}$. However, the separation caused more problem due to the non-commute nature of the variables they depend on. So instead of considering the effect of $e^{-i(t)\hat{H}}$ at once we instead divide the exponential of t into N pieces of $e^{-i\left(\frac{t}{N}\right)\hat{H}}$ and approximate the exponential,

$$e^{-i\frac{t}{N}(\hat{T}+\hat{V})} \approx e^{-i\frac{t}{N}\hat{T}}e^{-i\frac{t}{N}\hat{V}}\left(1+\frac{\left[\hat{T},\hat{V}\right]}{N^2}+\left(\frac{\mathcal{O}}{N^3}\right)\right) \tag{13}$$

under limit $N \to \infty$ we can neglect all the terms in (12) except the term with zeroth order of N hence the exponential of Hamiltonian is now separable. Then we apply N completeness relation of position basis, the relations become continuous, let $x = x_N$ and $x' = x_0$ our propagator becomes:

$$\begin{aligned} \left\langle x\middle|e^{-i(t)\hat{H}}\middle|x'\right\rangle &= \int dx_{N-1}\int dx_{N-2}\,\ldots.\int dx_1 \left\langle x_N\middle|e^{-i\frac{t}{N}\hat{H}}\middle|x_{N-1}\right\rangle\left\langle x_{N-1}\middle|e^{-i\frac{t}{N}\hat{H}}\middle|x_{N-2}\right\rangle \\ &\times \left\langle x_{N-1}\middle|e^{-i\frac{t}{N}\hat{H}}\middle|x_{N-2}\right\rangle\langle x_{N-2}|\,\ldots.|x_1\rangle\left\langle x_1\middle|e^{-i\frac{t}{N}\hat{H}}\middle|x_0\right\rangle \end{aligned} \tag{14}$$

Since $e^{-i\frac{t}{N}\hat{H}}$ can be separate into two terms, one depended on x and one depended on p we add another completeness relation, this time on momentum basis to calculate the term of kinetic energy.

$$\begin{aligned}\left\langle x\middle|e^{-i(t)\hat{H}}\middle|x'\right\rangle = \int dx_{N-1} \ldots . \int dx_1 \int dp_{N-1} \ldots . \int dp_0 \, \langle x_N|p_{N-1}\rangle \\ \times \left\langle p_{N-1}\middle|e^{-i\frac{t}{N}\frac{\hat{p}^2}{2m}}e^{-i\frac{t}{N}\hat{V}}\middle|x_{N-1}\right\rangle \langle x_{N-1}|p_{N-2}\rangle\langle p_{N-2}| \ldots \\ \ldots \langle x_1|p_0\rangle \left\langle p_0\middle|e^{-i\frac{t}{N}\frac{\hat{p}^2}{2m}}e^{-i\frac{t}{N}\hat{V}}\middle|x_0\right\rangle \end{aligned} \tag{15}$$

From (6) we see that bracket between position basis and momentum basis is Fourier transformation

$$\begin{aligned}\left\langle x\middle|e^{-i(t)\hat{H}}\middle|x'\right\rangle &= \int dx_{N-1} \ldots . \int dx_1 \int dp_{N-1} \ldots . \int dp_0 \frac{1}{\sqrt{2\pi\hbar}} e^{\frac{i}{\hbar}p_{N-1}x_N} e^{-\frac{i p_{N-1}^2}{\hbar}\frac{t}{2m}\frac{t}{N}} \\ &\quad \times e^{-\frac{i}{\hbar}V(x_{N-1})\frac{t}{N}} \frac{1}{\sqrt{2\pi\hbar}} e^{\frac{i}{\hbar}P_{N-1}x_{N-1}} \ldots . \frac{1}{\sqrt{2\pi\hbar}} e^{\frac{i}{\hbar}p_0x_1} e^{-\frac{i}{\hbar 2mN}\frac{p_0^2\, t}{}} e^{-\frac{i}{\hbar}V(x_0)\frac{t}{N}} \\ &= \int dx_{N-1} \ldots . \int dx_1 \int dp_{N-1} \ldots . \int dp_0 \frac{1}{2\pi\hbar} e^{\frac{i}{\hbar}p_{n-1}(x_N - x_{N-1})} e^{-\frac{i p_{N-1}^2}{\hbar}\frac{t}{2m}\frac{t}{N}} \\ &\quad \times e^{-\frac{i}{\hbar}V(x_{N-1})\frac{t}{N}} \ldots . \frac{1}{2\pi\hbar} e^{\frac{i}{\hbar}p_{0-1}(x_1 - x_0)} e^{-\frac{i}{\hbar 2mN}\frac{p_0^2\, t}{}} e^{-\frac{i}{\hbar}V(x_0)\frac{t}{N}} \end{aligned} \tag{16}$$

The sequence of xs with respect to time define the possible path of the particle, with this (12) and (16) tell us that $\psi(x,t)$ is the result of integral of contributions from all possible paths the particle in the past like we mention earlier. Furthermore (16) allow us to separate the term of potential energy from the rest momentum-dependent terms. Unless we know what exactly what potential energy takes form, we cannot calculate any further than that. Momentum-dependent terms, however, can be integrate out and make $\psi(x,t)$ in form of (10) which is the sum of position only, this can be done by defining imaginary time $\tau = -it$ and $\triangle\, \tau = \frac{\tau}{N}$ and consider a momentum term at time N

$$\begin{aligned}\int_{-\infty}^{\infty} dp_{N-1} \frac{1}{2\pi\hbar} e^{-\frac{i p_{N-1}^2}{\hbar}\frac{t}{2m}\frac{t}{N}+\frac{i}{\hbar}p_{n-1}(x_N - x_{N-1})} &= \int_{-\infty}^{\infty} dp_{N-1} \frac{1}{2\pi\hbar} e^{-\frac{\Delta\tau p_{N-1}^2}{\hbar\ 2m}+\frac{i}{\hbar}p_{n-1}(x_N - x_{N-1})} \\ &= \int_{-\infty}^{\infty} dp_{N-1} \frac{1}{2\pi\hbar} e^{-\frac{\Delta\tau}{2m\hbar}(p_{N-1}^2 - \frac{2im}{\Delta\tau}p_{N-1}(x_N - x_{N-1}))} \\ &= \frac{1}{2\pi\hbar} e^{-\frac{m}{2\hbar\Delta\tau}(x_N - x_{N-1})^2} \\ &\quad \times \int_{-\infty}^{\infty} dp_{N-1} e^{-\frac{\Delta\tau}{2m\hbar}(p_{N-1} - \frac{im}{\Delta\tau}(x_N - x_{N-1})^2)} \end{aligned} \tag{16}$$

Now let $u=\sqrt{\frac{m}{2\hbar\Delta\tau}}\left[p_{N-1}-\frac{im}{\Delta\tau}(x_N-x_{N-1})\right]$, (16) becomes Gaussian integral:

$$\int_{-\infty}^{\infty}dp_{N-1}\frac{1}{2\pi\hbar}e^{-\frac{ip_{N-1}^2}{\hbar}\frac{t}{2m}\frac{t}{N}+\frac{i}{\hbar}p_{n-1}(x_N-x_{N-1})}=\frac{1}{2\pi\hbar}\sqrt{\frac{2m\hbar}{\Delta\tau}}e^{-\frac{m}{2\hbar\Delta\tau}(x_N-x_{N-1})^2}\int_{-\infty}^{\infty}e^{-u^2}$$
$$=\sqrt{\frac{m}{2\pi\hbar\Delta\tau}}e^{-\frac{m}{2\hbar\Delta\tau}(x_N-x_{N-1})^2}\qquad(17)$$

The same goes for all momentum integrals, thus (15) becomes,

$$\langle x|e^{-i(t)\hat{H}}|x'\rangle=\int dx_{N-1}\ldots.\int dx_1\left(\frac{m}{2\pi\hbar\Delta\tau}\right)^{\frac{N}{2}}e^{\left(-\frac{m}{2\hbar}\frac{(x_N-x_{N-1})^2}{\Delta\tau^2}\Delta\tau-\frac{\Delta\tau}{\hbar}V(x_{N-1})\right)}\ldots.$$
$$\times e^{\left(-\frac{m}{2\hbar}\frac{(x_1-x_0)^2}{\Delta\tau^2}\Delta\tau-\frac{\Delta\tau}{\hbar}V(x_0)\right)}$$
$$=\int dx_{N-1}\ldots.\int dx_1\left(\frac{m}{2\pi\hbar\Delta\tau}\right)^{\frac{N}{2}}e^{\left(\frac{\Delta\tau}{\hbar}\left[-\frac{m}{2}\frac{(x_N-x_{N-1})^2}{\Delta\tau^2}-V(x_{N-1})\right]\right)}\ldots.$$
$$\times e^{\left(\frac{\Delta\tau}{\hbar}\left[-\frac{m}{2}\frac{(x_1-x_0)^2}{\Delta\tau^2}-V(x_0)\right]\right)}\qquad(18)$$

Now we convert time back to its real version,

$$\langle x|e^{-i(t)\hat{H}}|x'\rangle=\int dx_{N-1}\ldots.\int dx_1\left(\frac{m}{2\pi\hbar\Delta\tau}\right)^{\frac{N}{2}}e^{\left(\frac{\mathrm{i}}{\hbar}\Delta t\left[\frac{m}{2}\frac{(x_N-x_{N-1})^2}{\Delta t^2}-V(x_{N-1})\right]\right)}\ldots.$$
$$\times e^{\left(\frac{\mathrm{i}}{\hbar}\Delta t\left[\frac{m}{2}\frac{(x_1-x_0)^2}{\Delta t^2}-V(x_0)\right]\right)}\qquad(19)$$

Now we obtain the exponential of discrete Lagrangian $L=T(\dot{x})-V(x)$. Finally, we take limit $N\to\infty$ the propagator becomes:

$$\langle x|e^{-i(t)\hat{H}}|x'\rangle=\int_{x_{paths}}dxe^{\frac{i}{\hbar}S_{path}}\qquad(20)$$

Combine (20) with (11) we can write the wave function at position x and time t in form of (9) hence path integral formulation is derived from the solution of TISE proof the equivalence between three formulations.

CHAPTER III

PDE AND QUANTUM MECHANICS EXPLANATION OF RESIDUAL NETWORK

3.1 Hamiltonian of the hidden state

Consider a second order PDE which depend on two independent variables t and x:

$$\frac{\partial\psi(x,t)}{\partial t}=\frac{\alpha^2}{2}\frac{\partial^2\psi(x,t)}{\partial x^2}+\beta\frac{\partial\psi(x,t)}{\partial x}+\gamma\psi(x,t) \tag{21}$$

The time derivative and spatial derivative can be discretized using Euler discretization.

$$\begin{aligned}\frac{\partial\psi(x,t)}{\partial t}&=\frac{\psi(x,t+\Delta t)-\psi(x,t)}{\Delta t}\\ \frac{\partial\psi(x,t)}{\partial x}&=\frac{\psi(x+\Delta x,t)-\psi(x-\Delta x,t)}{2\Delta x}\\ \frac{\partial^2\psi(x,t)}{\partial x^2}&=\frac{\psi(x+\Delta \mathrm{x},t)-2\psi(x,t)+\psi(x-\Delta \mathrm{x},t)}{\Delta x^2}\end{aligned} \tag{22}$$

(21) is now can be rewritten in discrete form:

$$\begin{aligned}\frac{\psi(x,t+\Delta t)-\psi(x,t)}{\Delta t}=\frac{\alpha^2}{2}\frac{\psi(x+\Delta \mathrm{x},t)-2\psi(x,t)+\psi(x-\Delta \mathrm{x},t)}{\Delta x^2}\\ +\beta\frac{\psi(x+\Delta x,t)-\psi(x-\Delta x,t)}{2\Delta x}+\gamma\psi(x,t)\end{aligned} \tag{23}$$

By setting $\Delta t=\Delta x=1$, we can rewrite (23) in form of matrix multiplication.

$$\psi(x,t+1)-\psi(x,t)=\left[\frac{1}{2}(\alpha^2+\beta),(\gamma-\alpha^2),\frac{1}{2}(\alpha^2-\beta)\right]\begin{bmatrix}\psi(x+1,t)\\ \psi(x,t)\\ \psi(x-1,t)\end{bmatrix} \tag{24}$$

By define the convolution kernel $f(x)=\left[\frac{1}{2}(\alpha^2+\beta),(\gamma-\alpha^2),\frac{1}{2}(\alpha^2-\beta)\right]$, we can rewrite (24) in convolution form:

$$\begin{aligned}\psi(x,t+1)-\psi(x,t)=f(x)*\psi(x,t)\\ \psi(x,t+1)=f(x)*\psi(x,t)+\psi(x,t)\end{aligned} \tag{25}$$

As we see in (25) any second order PDE can be rewritten into the form of residual block [10] with convolution kernel $f(x)$ and input $\psi(x,t)$ showing equivalence between the two mathematical objects. In comparison to quantum mechanics, each element of the input of the t^{th} residual block resembles wave function at some point in position space and time t. With this comparison we can see that an operation of convolutional kernel on a hidden unit is equal to the operation of Hamiltonian operator on a wave function which is not wrong, but not completely right either since the hidden unit on the skip connection may also multiplied or even convolved by another convolution kernel before adding up to the convolved hidden unit. With this we can generalize our Hamiltonian-like object of hidden unit to be the quotient of convolution kernel and weight tensor on the skip connection under condition that the weight tensor is invertible.

3.2 Path integration form of residual network hidden unit

Consider the output of a residual block consisting of a convolutional layer and skip connection $f(x)$ with weight tensor w with ReLU activation function:

$$\psi(x_t) = relu[w \cdot \psi(x_{t-1}) + f(x_{t-1}) * \psi(x_{t-1})] \tag{26}$$

Since ReLU activation function can be applied on the activation map and skip connection separately (26), can be rewritten in form of

$$\psi(x_t) = \sum_{x_{t-1}} \kappa \cdot w\big(\delta(x_t - x_{t-1}) + \Omega(x_t - x_{t-1})\big)\psi(x_{t-1}) \tag{27}$$

Where $\kappa \cdot w$ is the activation function on weight tensor. By adopt notation of space-time of the network established in previous section we can see the delta function as inner product between two position bases.

$$\delta(x_t - x_{t-1}) = \sum_{p_{t-1}} e^{ip_{t-1}(x_t - x_{t-1})} \tag{28}$$

And the convolutional term, which act on the same delta function can be considered in momentum space that we defined as inverse Fourier transformation on the position space object.

$$\Omega(x_t - x_{t-1}) = \frac{1}{M}\sum_{k=0}^{M-1} e^{i\frac{2\pi}{M}k(x_t - x_{t-1})}\,\widetilde{\Omega}(p_{t-1}) \tag{29}$$

Then we define quantity $h_t = \log{(\kappa \cdot w)}$ we can rewrite (27) as

$$\psi(x_t) = \sum_{x_{t-1}}\sum_{p_{t-1}} e^{ip_{t-1}(x_t - x_{t-1})} e^{h_t}\left(1 + \widetilde{\Omega}(p_{t-1})\right)\psi(x_{t-1}) \tag{30}$$

Assume that the skip connection weights are much smaller than the convolutional kernel $w \ll f$, this allows us to approximate $1 + \widetilde{\Omega}(p_{t-1}) = e^{\widetilde{\Omega}(p_{t-1})}$, so we have:

$$\psi(x_t) = \sum_{x_{t-1}}\sum_{p_{t-1}} e^{ip_{t-1}(x_t - x_{t-1})} e^{h_t + \widetilde{\Omega}(p_{t-1})}\,\psi(x_{t-1}) \tag{31}$$

Now we define Hamiltonian $H(p_{t-1}) = h_t + \widetilde{\Omega}(p_{t-1})$ and consider the output from N residual blocks,

$$\psi(x_N) = \sum_{x_{N-1}}\sum_{x_{N-2}} \dots . \sum_{x_0}\sum_{p_{N-1}}\sum_{p_{N-2}} \dots . \sum_{p_0} e^{ip_{N-1}(x_N - x_{N-1}) + H(p_{N-1})}\, e^{ip_{N-2}(x_{N-1} - x_{N-2}) + H(p_{N-2})} \dots . \times e^{ip_0(x_1 - x_0) + H(p_0)}\psi(x_0) \tag{32}$$

The form of Hamiltonian used in this calculation is equivalent to what we mentioned in previous section since the term $\widetilde{\Omega}(p_{t-1})$ will be far larger than the term h_t which may neglectable at this point. The sequence of x_n and p_n successively in time define the path of the hidden unit so we may rewrite (31) in form of:

$$\psi(x_N) = \sum_{x_{path}} \sum_{p_{path}} \prod_t e^{ip_{t-1}(x_t - x_{t-1}) + H(p_{t-1})} \psi(x_0) \tag{33}$$

(33) tells us that the output from N hidden layers of ResNet is the summation of contributions from all paths the feature was travelled in the past [10]. Since we know that the residual block and PDE are equivalent, we can rewrite our Hamiltonian with a second order PDE. Since the current calculation is done in momentum space the spatial differential operator is replace by imaginary number ip, for simplicity we use equation (21) which can be rewritten in momentum space as $H(p_{t-1}) = -\frac{1}{2}\alpha^2 p_{t-1}^2 + ibp_{t-1} + c$, so we have:

$$\begin{aligned}\psi(x_N) &= \sum_{x_{path}} \sum_{p_{path}} \prod_t e^{ip_{t-1}(x_t - x_{t-1}) - \frac{1}{2}\alpha^2 p_{t-1}^2 + i\beta p_{t-1} + \gamma} \psi(x_0) \\ &= \sum_{x_{path}} \prod_t \sum_{p_{path}} e^{\gamma}\, e^{-\frac{1}{2}\alpha^2 p_{t-1}^2 + ip_{t-1}(x_t - x_{t-1} + \beta)} \psi(x_0)\end{aligned} \tag{34}$$

The momentum term in (33) can be integrate out like what have done in (19) hence we obtain:

$$\begin{aligned}\psi(x_N) &= \sum_{x_{path}} \prod_t \frac{e^{\gamma}}{\alpha} e^{-(x_t - x_{t-1} + \beta)^2 / 2\alpha^2} \psi(x_0) \\ &= \sum_{x_{path}} \prod_t e^{\gamma - \log(\alpha) - \frac{(x_t - x_{t-1} + \beta)^2}{2\alpha^2}} \psi(x_0)\end{aligned} \tag{35}$$

We can also define velocity over a time step as $\dot{x} = x_t - x_{t-1}$ thus we obtained kinetic energy $T(\dot{x})$ i.e., the term containing velocity and potential energy V as

$$T(\dot{x}) = \frac{(\dot{x} + \beta)^2}{2\alpha^2} \tag{36}$$

$$V = \gamma - \log(\alpha) \tag{37}$$

In the end we use definition of the Lagrangian $L = T - V$ and the classical action $S = \sum_t L_t$, (35) can now rewritten in more physical form:

$$\begin{aligned}\psi(x_N) &= \sum_{x_{path}} e^{\sum_t V - T(\dot{x})}\,\psi(x_0)\\ &= \sum_{x_{path}} e^{-\sum_t L_{path}^t}\psi(x_0)\\ &= \sum_{x_{path}} e^{-S_{path}}\psi(x_0) \qquad (38)\end{aligned}$$

Hence the output of multilayer residual block is the sum of contribution from all paths the input travelled in the past, where each contribution is an exponential of negative classical action of the path in the similar fashion to the Feynman path integral formulation of quantum mechanics.

CHAPTER IV

EXPERIMENT ON PDE NETWORK

The mathematical equivalence between PDE and residual block leads to the explanation of residual block in notation of Hamiltonian and path integral formulation of neural network as shown in chapter III. In this chapter we aim to proof the equivalence experimentally, to do that we create PDE network based on the second order PDE and compare with residual network in the control case which all residual blocks contain a skip connection over a convolutional layer. The comparison is showed in form of accuracy acquired from each network with varied depth of 5 layers, 15 layers, 27 layers, and 50 layers. The logistic regression for classifier is done by using 3 layers of linear neural network and using SoftMax function, and optimizer used in learning of the models is ADAM optimizer.

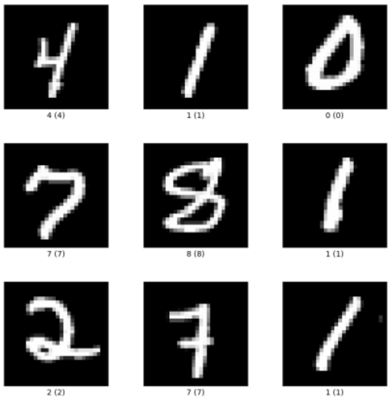


Figure 4.1: example of MNIST data, the dataset containing 60,000 pairs of handwriting image and labels for training the model and another 10,000 pairs for testing the model.

4.1 Structure of the PDE neural networks

Partial differential neural network created by replacing hidden layer of ResNet with PDE operator created by applying finite different method on the second order PDE, since convolutional kernel is two-dimensional tensor the PDE used in creation of network depends on three variables, by space-time notation in chapter II the PDE depends on time i.e., layer of the hidden layer and position in two cartesian coordinate x, y which is the width and length of the input tensor. The PDE used in this experiment is defined as:

$$\frac{\partial\psi(x,t)}{\partial t} = \frac{\alpha^2}{2}\frac{\partial^2\psi(x,t)}{\partial x^2} + \beta\frac{\partial\psi(x,t)}{\partial x} + \frac{\sigma^2}{2}\frac{\partial^2\psi(x,t)}{\partial y^2} + \rho\frac{\partial\psi(x,t)}{\partial y} + \gamma\psi(x,t) \tag{39}$$

Where $\alpha, \beta, \sigma, \rho,$ and γ are learning parameters. By discretization, the spatial derivatives in x coordinate become 1×3 tensors: $\frac{\partial^2}{\partial x^2} \to [1, -2, 1]$, $\frac{\partial}{\partial x} \to \left[\frac{1}{2},\ 0,\ -\frac{1}{2}\right]$, and spatial derivatives in y

coordinate become 3 × 1 tensors: $\frac{\partial^2}{\partial y^2} \rightarrow \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}, \frac{\partial}{\partial y} \rightarrow \begin{bmatrix} \frac{1}{2} \\ 0 \\ -\frac{1}{2} \end{bmatrix}$. When input tensor is sent to the first layer, it will be convoluted with product of kernel and learning parameters separately creating two activation maps, then the two activation maps along with the skip connection tensor are add up and pass to ReLU function and send value out as output of the layer, since normally convolution kernel are different in each layer, the PDE in each layer is also created differently, the procedure of creating activation maps and combine together with skip connection occurs repeatedly for all layers. Then the final output is sent to linear layers before send to linear network and use SoftMax classifier for logistic regression which return probability of each class of labels.

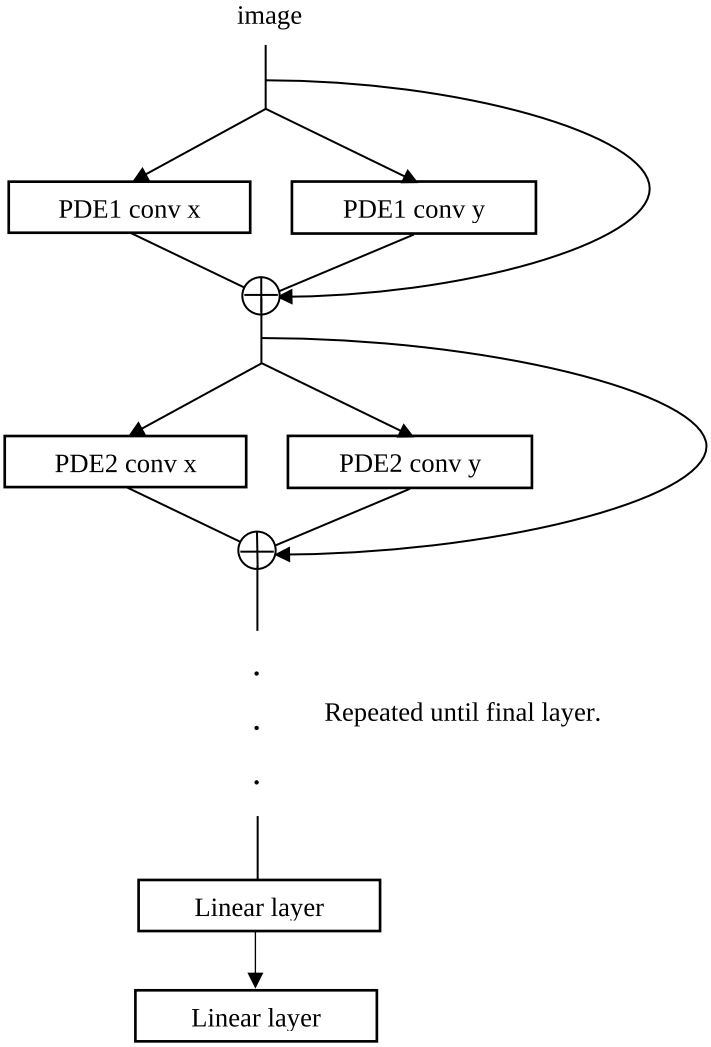
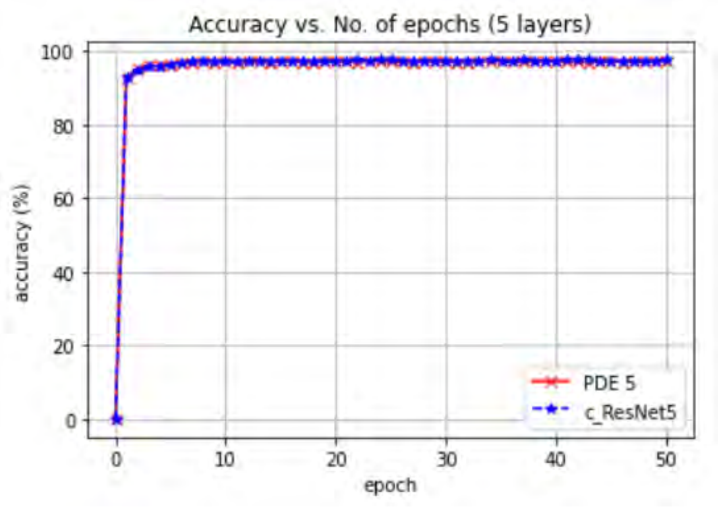


Figure 4.2: Algorithm for PDE neural networks

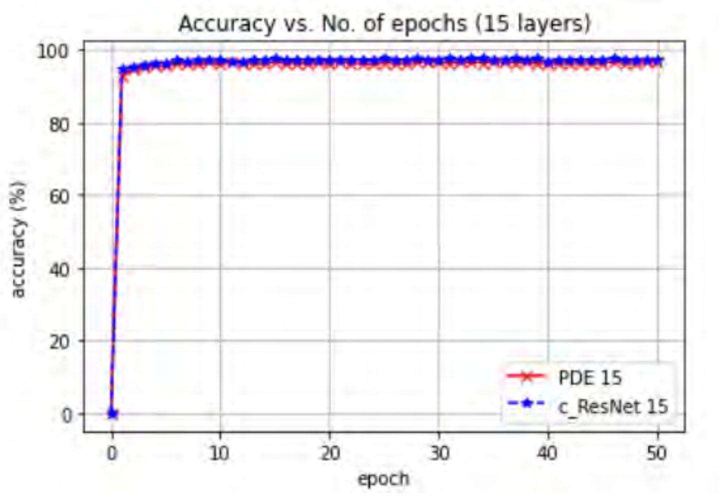
4.2 Result and discussion

4.2.1 Performance comparison between PDE network and ResNet

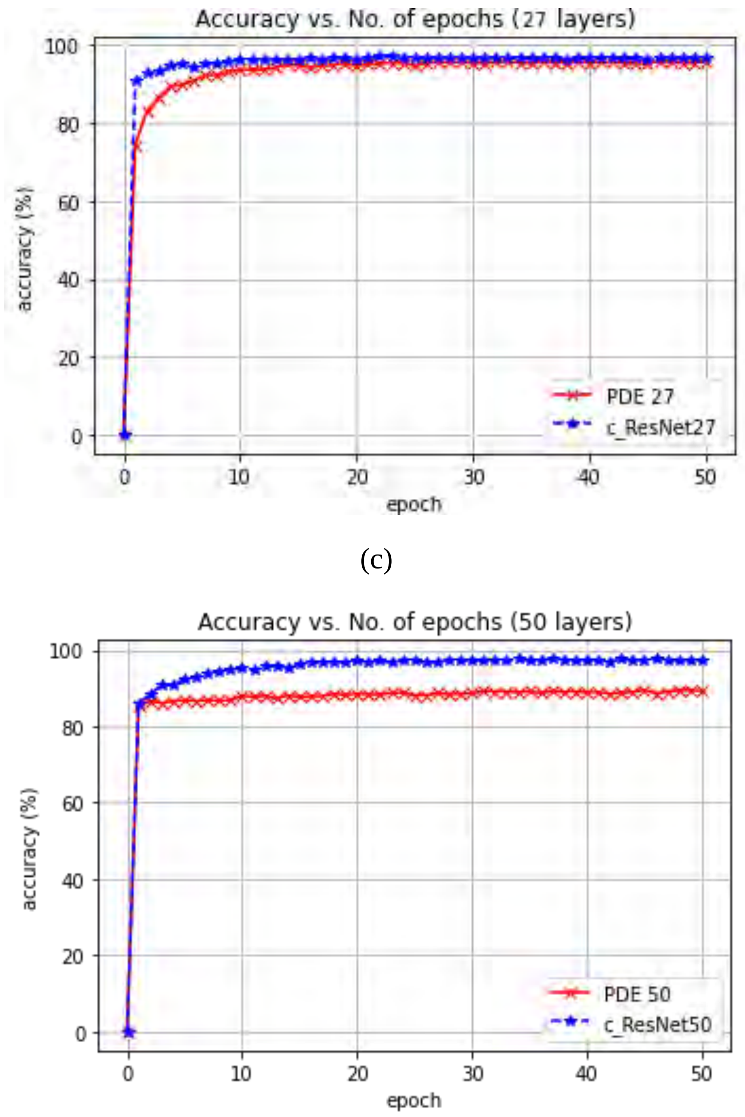
The comparison of performance between PDE network and residual network with depth of 5 layers, 15 layers, 27 layers, and 50 layers over 50 epochs are shown in Figure 4.3. we see that by replacing convolution kernel as partial differential operators can achieve the similar level of accuracy in classification task to the ResNet.



(a)



(b)



(c)

(d)

Fig.4.3 Comparison of accuracies of iteration over 50 epochs of PDE neural network and residual network with depth of (a) 5 layers, (b) 15 layers, (c) 27 layers, and (d) 50 layers.

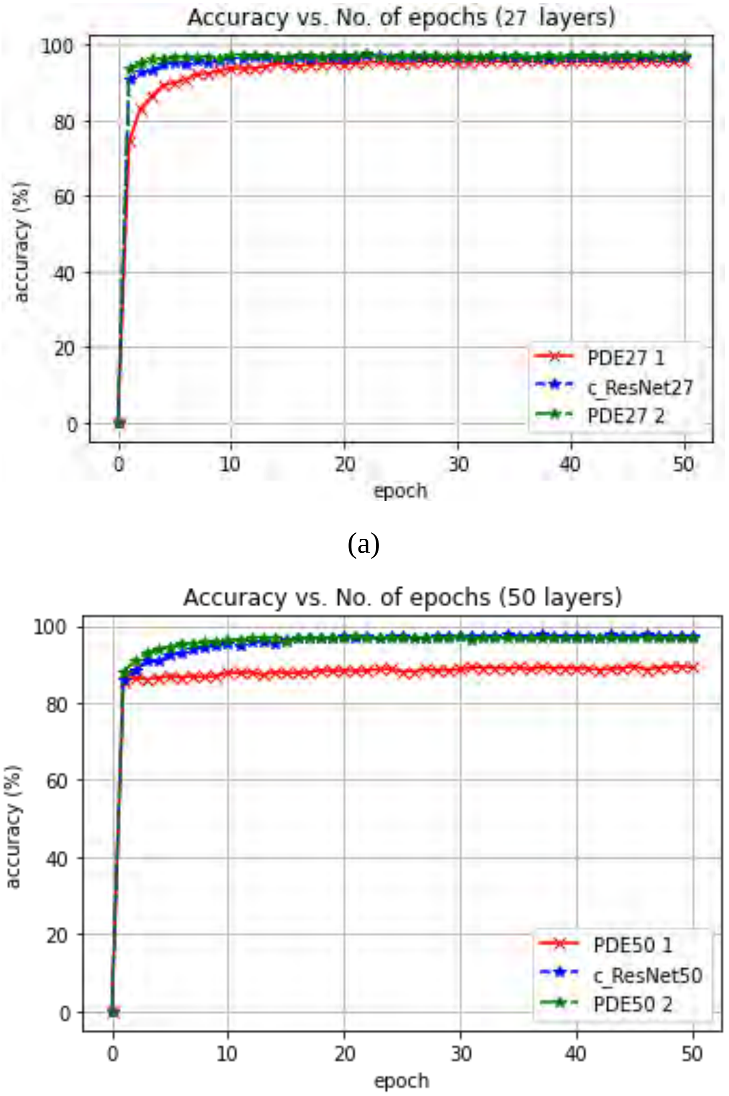
Table 4.1: Final output from logistic regression of PDE network and ResNet when learning rate is fixed at the same value of 10^{-3} and learning over 50 epochs.

layers	PDE neural network	c_ResNet
5	97.4%	97.5%
15	96.5%	97.3%
27	95.8%	97.0%
50	89.2%	96.2%

From the experiment we see that the accuracy drops as the network goes deeper, the cause of accuracy degradation is due to the nature of the PDE chosen for the network. The diffusion solution cause value of feature i.e., the non-zero value of tensor element to decrease and increase the value of nearby elements in the rate dictated by learning variables, this is not only making features to be more difficult to extract but also make the feature to "drift" to the direction assigned by the PDE which may cause features to vanish when the features are drift out the boundary of the tensor. This problem can be tackled by changing the PDE to change the rate of diffusion and control drifting of the features, so we decide to replace (39) with:

$$\frac{\partial\psi(x,t)}{\partial t}=\frac{\alpha^2}{2}\frac{\partial^2\psi(x,t)}{\partial x^2}-\frac{\beta}{8}\frac{\partial\psi(x,t)}{\partial x}+\frac{\sigma^2}{2}\frac{\partial^2\psi(x,t)}{\partial y^2}-\frac{\rho}{8}\frac{\partial\psi(x,t)}{\partial y}+\gamma\psi(x,t) \tag{40}$$

This time the model performance that shown in Figure 4.4, which clearly improved in terms of final accuracy and development of the model, which goes like the residual network we create as control.



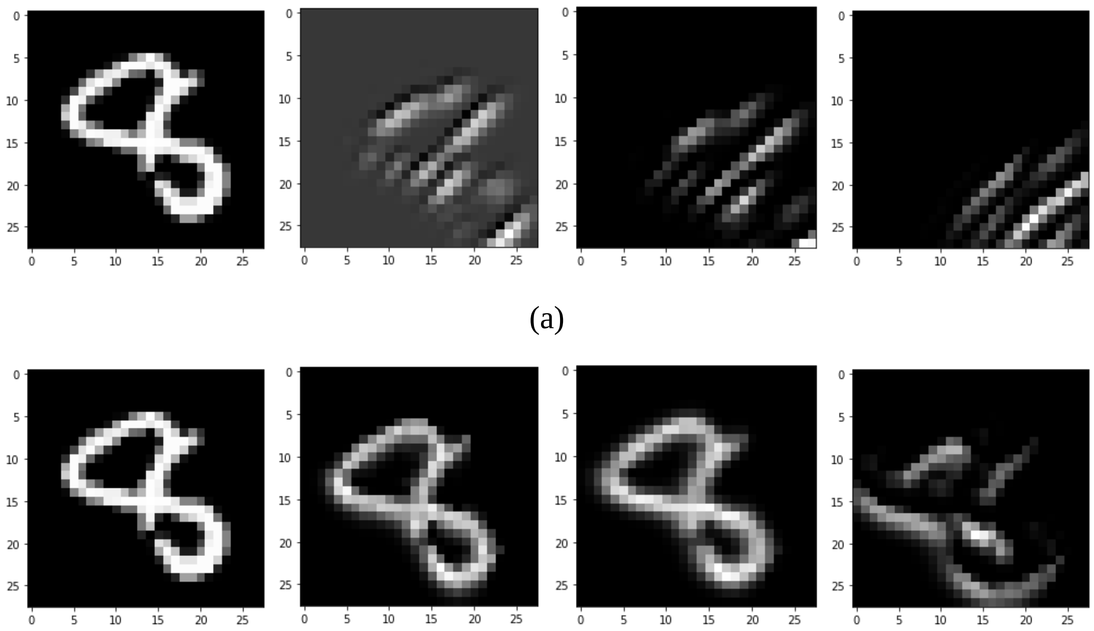
(a)

(b)

Fig 4.3 The comparison of performance of two PDE neural networks and residual network, note that PDE 1 is based on the equation (39) and PDE 2 is based on equation (40) in the depth of 27 layers (a) and 50 layers(b) which accuracy drops significantly.

Table 4.2 Comparison of final accuracy acquired from PDE 1, PDE2, and residual network.

layer	PED network 1	PDE network2	c_ResNet
27	95.8%	97.0%	96.7%
50	89.2%	96.9%	97.3%



(a)

(b)

Fig 4.5 Evolution of input data through 10 layers, 15 layers, and 27 layers of PDE network based on (a) equation (39) , and (b) based on equation (40).

CHAPTER V

FROM REDISUAL BLOCK TO NEURAL ODE

Mathematical understanding of hidden unit through a residual block showed in (25) leads to designing new architecture of ANN called neural ordinary differential equation or neural ODE in short, since (25) is considered as Euler discretization of a differential equation. By consider the continuous form and replace the spatial differential operators in the right-hand side with an arbitrary function of the hidden unit $\psi(t)$ and learning parameter θ_t at specific time t. The hidden unit becomes solution to the initial value problem of the ODE:

$$\frac{d\psi(t)}{dt} = f(\psi(t), \theta_t, t) \tag{41}$$

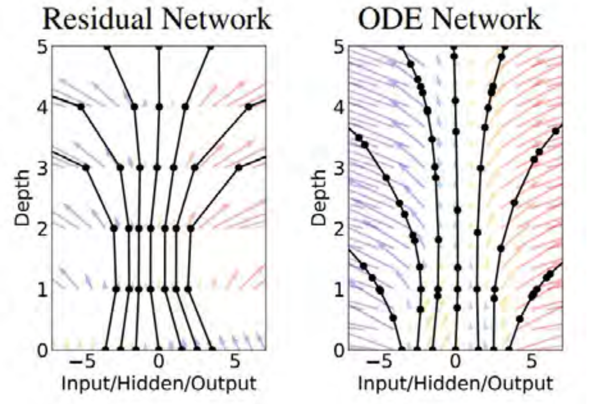


Fig 5.1 Paths of hidden unit propagated by hidden layer of ResNet and Neural ODE in comparison, We can see that ODE Network is the continuous limit of path caused by the architecture of ResNet.

In practice, the function $f(\psi(t), \theta_t, t)$ is the model created by a neural network layer at layer t [11]. The ODE is solvable via any differential equation solver which approximate in form:

$$\psi(t_1) = ODESolve(\psi(t_0), f, t_0, t_1, \theta) = \psi(t_0) + \int_{t_0}^{t_1} f(\psi(t), \theta_t, t)\, dt \tag{42}$$

Loss function used in this network architecture will tell how different our ODE solver approximation and the real function is [12].

Instead of optimize loss function $\mathcal{L}(\psi(t_1))$ with backpropagation, neural ODE optimizes loss function by use the benefits of ODE solver by calculating adjoint $a(t) = \frac{\partial \mathcal{L}}{\partial \psi(t)}$ and then use value of each adjoint to approximate gradient of loss function with respect to learning parameters at each time. However, the calculation of derivative

the set of $a(t)$ is obtained by solving ODE:

$$\frac{da(t)}{dt} = -a(t)^T \frac{\partial f(\psi(t), \theta_t, t)}{\partial \psi} \tag{43}$$

And $\frac{\partial \mathcal{L}}{\partial \psi(t_0)}$ can be calculated by solving (43) using ODE solver but backward in time starting from $z(t_1)$. The calculation of gradient with respect to parameter is also evaluate using integral:

$$\frac{d\mathcal{L}}{d\theta} = -\int_{t_1}^{t_0} a\,(t)^T \frac{\partial f(\psi(t), \theta_t, t)}{\partial \theta} \tag{44}$$

Where $a(t)^T \frac{\partial f(\psi(t),\theta_t,t)}{\partial \psi}$ and $a(t)^T \frac{\partial f(\psi(t),\theta_t,t)}{\partial \theta}$ can be calculated with automatic differentiation, this method is called “adjoint sensitivity method” [13]. One major benefits of learning parameter this way is that the method does not need to store any quantity of parameters when calculating gradient like traditional backpropagation causing the memory cost of this method constant as $f(\psi(t), \theta_t, t)$. The network also has many benefits such as the highly adaptability with several ODE solvers we currently have, the speed of calculation and error can be controlled. And have high efficiency when it comes to predict continuous dataset of time-series [14].

CHAPTER VI

CONCLUSION

What we have done so far is develop the mathematical explanation of dynamics of hidden unit of a neural network architecture, in which we choose residual network, we show that the structure of residual block is equivalent to the PDE and when compare dynamics of hidden unit to the dynamics of quantum particle we can point that convolution kernel act as Hamiltonian of the system of hidden unit and we can use this Hamiltonian to derive path integral formulation of output of N residual block similar to the Feynman path integral. The concept of PDE and residual block is shown experimentally by create PDE network which achieve value of accuracy close to the ResNet with skip connection between each convolutional kernel we create as a control. Finally we shown that the understanding of residual block as differential equation leads to the innovation of new neural network called neural ordinary differential equation which adopt adjoint sensitivity method to update parameter enhance efficiency of memory management, easy to adjust the model, and able to perform the prediction of continuous dynamics.

REFERENCE

[1] Metha, P., Bukov, M., Wang, C., Day, G.R. A., Richardson, C., Fisher K. C., Schwab, j. D. (2019, March 23). A high-bias, low-variance introduction to Machine Learning for physicists. Retrieved from https://arxiv.org/abs/1803.08823

[2] Saha, S. (2018, December 16). A Comprehensive Guide to Convolutional Neural Networks – the ELI5 way. A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way | by Sumit Saha | Towards Data Science

[3] Srizhevsky, A., Sutskever, I., Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. Retrieved from imagenet_classification_with_deep_convolutional.pdf (toronto.edu)

[4] He, K., Zhang, X., Ren, S., Sun, J., (2015, December 10). Deep Residual Learning for Image Recognition. Retrieved from https://arxiv.org/abs/1512.03385

[5] Wang, C., (2019, January 8). The Vanishing Gradient Problem: The Problem, Its causes, It Significance, and Its Solutions. https://towardsdatascience.com/the-vanishing-gradient-problem-69bf08b15484

[6] Griffiths, D. (2005) “Introduction to quantum mechanics” 2nd ed. Pearson.

[7] Riley, K., Hobson, M., Bence, S. (2006). “Mathematical Methods for Physics and Engineering” 3rd ed. Cambridge university press.

[8] Dirac, P. (1939, April). “A New Notation for Quantum Mechanics”. Mathematical Proceedings of the Cambridge Philosophical Society, Vol25, pp 416-418. Cambridge, England: Cambridge university press.

[9] Feynman, R. (1948, April). “Space-Time Approach to non-Relativistic Quantum Mechanics”. Rev. Mod. Physics, Vol 20, pp. 367-387.

[10] Yin, M., Li, X., Zhang, Y., & Wang, S. (2019, April 16). On the Mathematical Understanding of ResNets with Feynman Path Integral. Retrieved from https://arxiv.org/abs/1904.07568

[11] Sinai, J. (2019, January 18). Understanding Neural ODE’s. https://jontysinai.github.io/jekyll/update/2019/01/18/understanding-neural-odes.html

[12] Surtsukov, M. (2019, February 16). Neural Ordinary Differential Equations. neural-ode/Neural ODEs.ipynb at master · msurtsukov/neural-ode (github.com)

[13] Chen, T.R., Rubanova, Y., Bettencourt, J., & Duvenaud, D. (2018, June 19). Neural Ordinary Differential Equations. Retrieved from https://arxiv.org/abs/1806.07366

[14] Honchar, A. (2019 June 12). Neural ODEs: breakdown of another deep learning breakthrough. Neural ODEs: breakdown of another deep learning breakthrough | by Alexandr Honchar | Towards Data Science