

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YAPAY SİNİR AĞLARI YARDIMIYLA
SCHRÖDİNGER DENKLEMİNİN NÜMERİK
ANALİZİ

Emrah SAMUR

YÜKSEK LİSANS TEZİ
Fizik Anabilim Dalı
Fizik Programı

Danışman
Doç. Dr. Bora IŞILDAK

Haziran, 2023

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YAPAY SİNİR AĞLARI YARDIMIYLA SCHRÖDİNGER
DENKLEMİNİN NÜMERİK ANALİZİ

Emrah SAMUR tarafından hazırlanan tez çalışması 16.06.2023 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Fizik Anabilim Dalı Fizik Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Doç. Dr. Bora IŞILDAK
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Doç. Dr. Bora IŞILDAK, Danışman
Yıldız Teknik Üniversitesi

Prof. Dr. Kutsal BOZKURT, Üye
Yıldız Teknik Üniversitesi

Doç. Dr. Alper HAYRETER, Üye
Özyeğin Üniversitesi

Danışmanım Doç. Dr. Bora IŞILDAK sorumluluğunda tarafımca hazırlanan YAPAY SİNİR AĞLARI YARDIMIYLA SCHRÖDİNGER DENKLEMİNİN NÜMERİK ANALİZİ başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim.

Emrah SAMUR

İmza

Aileme ve hocalarima

TEŞEKKÜR

Bu çalışmanın yapılmasında desteklerini esirgemeyen danışman hocam Doç. Dr. Bora Işıldak’a sonsuz teşekkürlerimi sunarım. Ayrıca tez jürümde olmayı kabul eden Prof. Dr. Kutsal Bozkurt ve Doç. Dr. Alper Hayreter hocalarıma da teşekkür etmek isterim. Tez sürecimde bana yardımcı olan Fatih Ilgın hocama da teşekkür ederim.

Bu çalışmanın başlamasında bana her türlü maddi ve manevi desteğini esirgemeyen aileme ve arkadaşlarıma sonsuz teşekkürler ederim.

Emrah SAMUR

İÇİNDEKİLER

TEŞEKKÜR	vii
SİMGE LİSTESİ	viii
KISALTMA LİSTESİ	ix
ŞEKİL LİSTESİ	x
TABLO LİSTESİ	xii
ÖZET	xiii
ABSTRACT	xv
1 GİRİŞ	17
1.1 Giriş	17
2 TEORİK BİLGİLER	19
2.1 Makine Öğrenmesi	19
2.1.1 Yapay Zeka ve Makine Öğreniminin Günlük Hayata Yansıması	20
2.2 Derin Yapay Sinir Ağları	21
2.2.1 Keras	22
3 KULLANILAN YÖNTEM VE ARAÇLAR	23
3.1 Schrödinger Denklemi'nin Farklı Potansiyele Değerleri İçin Nümerik Olarak Çözümlemesi	24
3.1.1 Türev Operatörleri ve Python'da Kodlanması	24
3.1.2 Sonsuz Potansiyel Kuyusu	26
3.1.3 Potansiyellerin Üretilmesi	32
3.1.4 Derin Sinir Ağlarının Oluşturulması	32
4 SONUÇ	36
4.1 Eğitilen Modellerin Değerlendirilmesi	36

KAYNAKÇA	40
TEZDEN ÜRETİLMİŞ YAYINLAR	42

TEŞEKKÜR

$\hbar$	Planck Sabiti
ω	Açısal Frekans
m	Kütle
L	Uzunluk
$V(x)$	Potansiyel
$\psi(x)$	Dalga Fonksiyonu

SİMGE LİSTESİ

$\hbar$	Planck Sabiti
ω	Açısal Frekans
m	Kütle
L	Uzunluk
$V(x)$	Potansiyel
$\psi(x)$	Dalga Fonksiyonu

KISALTMA LİSTESİ

ML	Makine Öğrenmesi
API	Uygulama Programlama Arayüzü
DNN	Deep Neural Network
a.u.	Arbitrary Unit

ŞEKİL LİSTESİ

Şekil 3.1	Sonsuz kuyu potansiyeli	26
Şekil 3.2	Sonsuz potansiyel kuyusu için elde edilmiş nümerik dalga fonksiyonları.	27
Şekil 3.3	Sonsuz potansiyel kuyusu nümerik dalga fonksiyonları ile analitik dalga fonksiyonlarının karşılaştırılması	28
Şekil 3.4	Harmonik salıncı potansiyeli	28
Şekil 3.5	Harmonik salıncı için nümerik olarak elde edilmiş dalga fonksiyonları.	29
Şekil 3.6	Harmonik osilatör için nümerik olarak elde edilmiş dalga fonksiyonları ile analitik olarak elde edilmiş dalga fonksiyonlarının karşılaştırılması.	30
Şekil 3.7	Doğrusal potansiyel.	30
Şekil 3.8	Doğrusal potansiyel için nümerik olarak elde edilmiş dalga fonksiyonları.	31
Şekil 3.9	Rastgele potansiyeller. İlk sıra stil 0 potansiyel, ikinci sıra stil 1 ve üçüncü sıra stil 2 potansiyeldir.	33
Şekil 3.10	Enerji değerlerinin bulunması için kullanılan model için Keras kod parçası.	34
Şekil 3.11	Dalga fonksiyonlarının bulunması için kullanılan model için Keras kod parçası.	35
Şekil 4.1	Enerji öz değerleri arasındaki farkın dağılımı	36
Şekil 4.2	Üst Panel: Dalga fonksiyonları üzerine eğitilen modelden elde edilen fonksiyonların ve nümerik çözümden elde edilen dalga fonksiyonların mutlak kareleri ile doğrusal regresyon (Linear Regression) ve ElasticNet Regresyon yöntemleri ile elde edilen dalga fonksiyonu tahminleri. Alt Panel: DNN ve nümerik çözüm arasındaki bağıl fark.	37

Şekil 4.3	Üst Panel: Dalga fonksiyonları üzerine eğitilen modelden elde edilen fonksiyonların ve nümerik çözümde elde edilen dalga fonksiyonların mutlak kareleri ile doğrusal regresyon (Linear Regression) ve ElasticNet Regresyon yöntemleri ile elde edilen dalga fonksiyonu tahminleri. Alt Panel: DNN ve nümerik çözümde arasındaki bağıl fark.	37
Şekil 4.4	Üst Panel: Dalga fonksiyonları üzerine eğitilen modelden elde edilen fonksiyonların ve nümerik çözümde elde edilen dalga fonksiyonların mutlak kareleri ile doğrusal regresyon (Linear Regression) ve ElasticNet Regresyon yöntemleri ile elde edilen dalga fonksiyonu tahminleri. Alt Panel: DNN ve nümerik çözüm arasındaki bağıl fark.	38
Şekil 4.5	Dalga fonksiyonları arasındaki açılarının dağılımı	38

TABLO LİSTESİ

Tablo 3.1	Sonsuz potansiyel kuyu enerji çözümlerinin karşılaştırılması. Nümerik çözümler arasındaki fark %0.1'den azdır.	26
Tablo 3.2	Sonsuz kuyu potansiyeli için nümerik enerji çözümleri	27
Tablo 3.3	Harmonik osilator enerji çözümlerinin karşılaştırılması. Nümerik çözümler arasındaki fark %0.1'den azdır.	29
Tablo 3.4	Doğrusal potansiyel için elde edilmiş nümerik enerji çözümleri. .	31
Tablo 3.5	Doğrusal potansiyel için enerji çözümlerinin karşılaştırılması. Nümerik çözümler ile arasındaki fark %0.1'den azdır.	32

YAPAY SİNİR AĞLARI YARDIMIYLA SCHRÖDİNGER DENKLEMİNİN NÜMERİK ANALİZİ

Emrah SAMUR

Fizik Anabilim Dalı
Yüksek Lisans Tezi

Danışman: Doç. Dr. Bora IŞILDAK

Bu çalışmada Zamandan Bağımsız Schrödinger Denklemini (ZBSD) çözmek ve çeşitli rastgele potansiyel kuyuları için bir parçacığın ilk beş öz durumunun enerjilerinin ve dalga fonksiyonlarının bulunması için bir Derin Sinir Ağı (DSA) kullanılmıştır. Bu rastgele potansiyeller karmaşıklıklarına göre üç kategoride oluşturulmuştur. Karşılaştırma için analitik çözümlerin elde edilebildiği durumlarda analitik çözümler kullanılmıştır. Eğer Schrödinger Denklemini (ZBSD) analitik olarak çözülebilir değilse, sayısal yaklaşımları bulmak için Sonlu Farklar Yöntemini (SFY) kullanılmıştır. Derin Sinir Ağı (DSA) kullanılarak elde edilen dalga fonksiyonları ve enerjiler, analitik yöntemler veya Sonlu Farklar Yöntemini elde edilen sonuçlarla iyi bir uyum içerisinde. Schrödinger denkleminin formu fiziksel duruma bağlıdır. Bu formlar zamana bağlı Schrödinger denklemi ve zamandan bağımsız Schrödinger denklemi olarak bilinmektedir. Zamana bağlı Schrödinger denklemi, dalga fonksiyonlarının durağan durumlar olarak adlandırılan duran dalgalar oluşturabileceğini tahmin eder. Bu durumların bireysel çalışmaları daha sonra herhangi bir durum için zamana bağlı Schrödinger denklemini çözme görevini basitleştirdiği için özellikle önemlidir. Durağan durumlar ayrıca, zamandan bağımsız Schrödinger denklemi olan Schrödinger denkleminin daha basit bir formuyla da tanımlanabilir. Bu süreçte yazılım dillerinden yeni ve yaygın olan Python'dan yararlandık. İlerideki bölümlerde göreceğiniz gibi önce Schrödinger denklemini kod için türev operatörlerini tanımladık. Daha sonrasında potansiyellerimizi üreterek bunların nümerik çözümlerini bulduk. Son olarak da

modelimizi oluřturarak elde ettiđimiz sonuların dođruluđunu inceledik.

Anahtar Kelimeler: Yapay Sinir Ađları, Schrödinger Denklemi, Özdeđer Problemi

ABSTRACT

NUMERICAL ANALYSIS OF SCHRÖDINGER EQUATION WITH THE HELP OF ARTIFICIAL NEURAL NETWORKS

Emrah SAMUR

Department of Physics
Master of Science Thesis

Supervisor: Assoc. Prof. Dr. Bora IŞILDAK

In this study, a Deep Neural Network (DNN) is used to solve the Time Independent Schrödinger Equation (TISE) and to find the energies and wave functions of the first five eigenstates of a particle for various random potential wells. These random potentials are categorized into three categories according to their complexity. Analytical solutions were used for comparison when analytical solutions were available. If the Schrödinger Equation (SE) was not solvable analytically, the Finite Difference Method (FDM) was used to find numerical approximations. The wave functions and energies obtained using the Deep Neural Network (DNN) are in good agreement with the results obtained by analytical methods or the Finite Difference Method. The form of the Schrödinger equation depends on the physical state. These forms are known as time-dependent Schrödinger equation and time-independent Schrödinger equation. The time-dependent Schrödinger equation predicts that wave functions can form standing waves, so-called stationary states. Individual studies of these states are particularly important as they then simplify the task of solving the time-dependent Schrödinger equation for any given state. Stationary states can also be described by a simpler form of the Schrödinger equation, the time-independent Schrödinger equation. As you will see in the following sections, we first define the derivative operators to code the Schrödinger equation, then we generate our potentials and find their numerical solutions, and finally we build our model and check the accuracy of our results.

Keywords: Neural network; Schrödinger equation; Eigenvalue problems

YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

1.1 Giriş

Bir kuantum sisteminin dalga fonksiyonunu tanımlayan Schrödinger denklemi, kuantum mekaniğinde önemli bir keşiftir. 1926'da Erwin Schrödinger tarafından ortaya atılan doğrusal bir kısmi diferansiyel denklemdir ve klasik mekanikteki karşılığı olan Newton'un ikinci yasasına benzer [1]. Bu ünlü denklemin adı, çalışmasıyla 1933'te Nobel Fizik Ödülü'nü kazanan Erwin Schrödinger'den geliyor. Schrödinger denklemi, izole bir fiziksel sistemi karakterize eden bir dalga fonksiyonunun zamanla değişimini tanımlayan, kuantum mekaniğinin temelini oluşturan bir denklemdir. Özünde, klasik mekanikte Newton'un ikinci yasasının kuantum eşdeğeri. Newton'un ikinci yasası, bilinen başlangıç koşullarına dayalı olarak fiziksel bir sistemin zaman içindeki hareketini tahmin ederken, Schrödinger denklemi dalga fonksiyonunun zaman içindeki evrimini tahmin eder. Bu denklem, kendine eşlenik bir operatörün veya bir kuantum Hamiltoniyenin üstel değeri tarafından üretilen, zaman evrimi operatörünün üniter olması gerekliliğinden türetilir [2].

Schrödinger denklemi, bir sistemdeki parçacıkların davranışını tanımlayan kuantum mekaniğinde temel bir denklemdir. Bu denklemin hem genel bir formu hem de göreceli olmayan bir versiyonu vardır. Genel denklem oldukça çok yönlü olsa ve çeşitli kuantum mekaniği uygulamalarında kullanılsa da, rölativistik olmayan versiyon, birçok durumda oldukça doğru sonuçlar veren bir yaklaşımdır. Schrödinger denklemini kullanmak için, önce sistemi oluşturan parçacıkların kinetik ve potansiyel enerjileri cinsinden Hamiltoniyeni ifade etmek gerekir. Bu Hamiltoniyen daha sonra Schrödinger denklemine dahil edilir ve sistem hakkındaki bilgileri kodlayan dalga fonksiyonunu elde etmek için çözülebilen kısmi bir diferansiyel denklemle sonuçlanır. Dalga fonksiyonunun mutlak değerinin karesi, bir parçacığı uzayda belirli bir noktada bulma olasılığını tanımlayan bir olasılık yoğunluk fonksiyonunu tanımlamak için kullanılabilir. Schrödinger denklemi fiziksel duruma bağlı olarak farklı biçimler alır, yani zamana bağlı

Schrödinger denklemi ve zamandan bağımsız Schrödinger denklemi. Zamana bağlı denklem sayesinde, kuantum mekaniği çalışması için çok önemli olan duran dalgaların veya durağan durumların oluşumunu tahmin edebiliriz. Bu durağan durumlara odaklanarak, herhangi bir durum için zamana bağlı Schrödinger denklemini çözmek daha kolay hale gelir. Ayrıca, zamandan bağımsız Schrödinger denklemini kullanarak durağan durumların tanımını basitleştirebiliriz [3].

Makine öğrenimi son yıllarda oldukça popüler hale geldi ve çok sayıda alanda uygulandı. Karmaşık problemleri çözmek için güçlü bir araç olmasının yanı sıra, kuantum sistemlerinin incelenmesine uygulanması sayesinde, fizik araştırmalarında hızla artan bir paya sahip olmaya devam etmektedir. Örneğin, araştırmacılar kuantum fiziğindeki kuantum durum tomografisi [4], Hamiltoniyenleri öğrenme [5], kuantum faz geçişlerini anlama ve yeni kuantum deneyleri üretme [6] gibi çeşitli zorlukların üstesinden gelmek için klasik makine öğrenimi tekniklerini kullanmaktadır. Klasik makine öğrenimi, bilinmeyen kuantum sistemlerini karakterize etmek için büyük miktarlarda deneysel veya hesaplamalı verilerin işlenmesinde özellikle yararlıdır. Bu yaklaşım, kuantum bilgi teorisi, kuantum teknolojisi geliştirme ve hesaplamalı malzeme tasarımı dahil olmak üzere çeşitli araştırma konularında kendisine uygulama alanı bulmaktadır. Kullanımına bir örnek olarak, önceden hesaplanmış atomlar arası potansiyelleri interpolasyon yapmak [7] veya varyasyonel yöntemle Schrödinger denklemini doğrudan çözmek verilebilir [8].

Bu çalışmada, Python yazılım dilini kullanarak farklı potansiyeller için zamandan bağımsız Schrödinger denklemini çözerek elde ettiğimiz verileri inceledik. İlk olarak, türev işlemlerini tanımlayarak Schrödinger denklemini programlamak için hazırlıklar yaptık. Daha sonra, çeşitli potansiyelleri oluşturarak nümerik çözümlerini bulduk. Son olarak, modelimizi oluşturduk ve elde ettiğimiz sonuçların doğruluğunu değerlendirdik. İlerideki bölümlerde daha ayrıntılı olarak göreceğiniz gibi, bu yöntem, farklı potansiyellere sahip sistemlerin davranışlarını anlamak için kullanılabilir ve daha geniş kapsamlı araştırmalar için önemli bir temel oluşturur.

2 TEORİK BİLGİLER

2.1 Makine Öğrenmesi

Makine öğrenimi (ML), verilerden öğrenebilen ve bu öğrenmeye dayalı olarak tahminler veya kararlar verebilen algoritmalar geliştirmeye odaklanan yapay zekanın (AI) bir alt alanıdır. Makine öğreniminin arkasındaki ana fikir, kendilerine verilen verilerden öğrenerek performanslarını artırabilecek sistemler oluşturmaktır. Bu, makinelerin, insanların manuel olarak yapması zor veya imkansız olan görevleri gerçekleştirmesini sağlar.

Üç ana makine öğrenimi türü vardır: denetimli öğrenme, denetimsiz öğrenme ve takviyeli öğrenme.

Denetimli öğrenme, makine öğreniminin en yaygın kullanılan türüdür. Her veri noktasının bilinen bir çıktıya sahip olduğu etiketli bir veri kümesi üzerinde bir modelin eğitilmesini içerir. Model daha sonra girdileri çıktılara eşlemeyi öğrenir ve yeni, görünmeyen veriler üzerinde tahminler yapmak için kullanılabilir.

Öte yandan denetimsiz öğrenme, bilinen bir çıktının olmadığı, etiketlenmemiş bir veri kümesi üzerinde bir model eğitmeyi içerir. Model daha sonra herhangi bir yönlendirme olmadan verilerdeki kalıpları ve yapıyı kendi başına tanımlamalıdır.

Takviyeli öğrenme, bir temsilcinin bir ortamla etkileşime girerek ve ödülleri veya cezalar şeklinde geri bildirim alarak karar vermeyi öğrendiği bir makine öğrenimi türüdür. Temsilcinin amacı, zaman içinde kümülatif ödülünü maksimize etmektir.

Makine öğrenimindeki en büyük zorluklardan biri, modellere eklenebilecek önyargıdır (bias). Önyargı, önyargılı eğitim verileri veya önyargılı algoritmalar gibi birçok yolla tanıtılabilir. Makine öğrenimi modellerinde önyargıyı ele almak, adil ve doğru olmalarını sağlamak için çok önemlidir [9].

2.1.1 Yapay Zeka ve Makine Öğreniminin Günlük Hayata Yansıması

Makine öğrenimi, görüntü ve konuşma tanıma, doğal dil işleme, öneri sistemleri ve otonom araçlar dahil olmak üzere birçok pratik uygulamaya sahiptir. Sağlık, finans ve ulaşım gibi sektörlerde devrim yaratma potansiyeline sahiptir ve önümüzdeki yıllarda önemli bir araştırma ve geliştirme alanı olmaya devam edecektir.

Makine öğrenimi giderek günlük hayatımızın bir parçası haline geliyor ve biz bunun farkında bile olmayabiliriz. Makine öğreniminin günlük uygulamalarda nasıl kullanıldığına dair bazı örnekler:

Sanal asistanlar: Siri, Alexa ve Google Assistant gibi sanal asistanlar, doğal dili anlamak ve kullanıcı isteklerine yanıt vermek için makine öğrenimi algoritmalarını kullanır. Zaman içinde yanıtlarını iyileştirmek için kullanıcı etkileşimlerinden öğrenirler.

Sosyal medya: Sosyal medya platformları, kullanıcı akışlarını kişiselleştirmek ve kullanıcı tercihlerine göre içerik önermek için makine öğrenimi algoritmalarını kullanır. Kişiselleştirilmiş içerik sunmak için kullanıcı davranışını, etkileşimi ve diğer verileri analiz ederler.

E-posta spam filtreleri: E-posta sağlayıcıları, spam e-postaları filtrelemek için makine öğrenimi algoritmalarını kullanır. Algoritmalar, istenmeyen e-postaları tanımlamayı ve engellemeyi kullanıcı geri bildirimlerinden öğrenir.

Çevrimiçi alışveriş: Çevrimiçi alışveriş platformları, kullanıcı tercihlerine ve satın alma geçmişine dayalı olarak ürün önermek için makine öğrenimi algoritmalarını kullanır. Kullanıcıların hangi ürünleri satın alma olasılığının yüksek olduğunu tahmin etmek için kullanıcı davranışını analiz ederler.

Dolandırıcılık tespiti: Bankalar ve kredi kartı şirketleri, sahte işlemleri tespit etmek için makine öğrenimi algoritmaları kullanır. Algoritmalar, şüpheli etkinliği belirlemek için kullanıcı davranışını ve işlem verilerini analiz eder.

Trafik tahmini: Google Haritalar ve Waze gibi navigasyon uygulamaları, trafik koşullarını tahmin etmek için makine öğrenimi algoritmalarını kullanır. Doğru seyahat süresi tahminleri sağlamak için geçmiş trafik verilerini, gerçek zamanlı trafik güncellemelerini ve diğer verileri analiz ederler.

Ses tanıma: Ses tanıma teknolojisi, konuşulan kelimeleri tanımak ve bunları metne dönüştürmek için makine öğrenimi algoritmalarını kullanır. Bu teknoloji sanal

asistanlarda, dikte yazılımlarında ve diğer uygulamalarda kullanılır. Bunlar, makine öğreniminin günlük hayatımızda nasıl kullanıldığına dair yalnızca birkaç örnek. Teknoloji ilerlemeye devam ettikçe, çeşitli endüstrilerde ve sektörlerde daha fazla makine öğrenimi uygulaması görmeyi bekleyebiliriz[10–12].

2.2 Derin Yapay Sinir Ağları

Derin sinir ağı (DNN), giriş ve çıkış katmanları arasında birden çok katmana sahip bir tür sinir ağıdır. Bir DNN'deki her katman, girdinin doğrusal olmayan bir dönüşümünü gerçekleştirir ve bir katmanın çıktısı, bir sonraki katmana girdi görevi görür. DNN'ler son yıllarda görüntü tanıma, doğal dil işleme gibi karmaşık sorunları çözmek ve Go ve satranç gibi oyunlar oynamak için giderek daha popüler hale geldi.

Schrödinger denklemini farklı potansiyellerle çözmek, kuantum mekaniğinde yaygın bir sorundur. Bu sorunu çözmek için bir yaklaşım, sistemin dalga fonksiyonunu öğrenmek için bir DNN kullanmaktır. DNN, potansiyel işlevi girdi olarak alır ve karşılık gelen dalga işlevini verir. Bu yaklaşım "kuantum makine öğrenimi" olarak bilinir.

Schrödinger denklemini çözmek için bir DNN kullanmak için, önce girdi-çıkı çiftlerinden oluşan bir eğitim seti oluşturmamız gerekir. Bunu, Schrödinger denklemini farklı potansiyel fonksiyonlar için sayısal olarak çözerek ve ortaya çıkan dalga fonksiyonlarını çıktılar ve potansiyelleri girdiler olarak kullanarak yapabiliriz. Daha sonra, tahmin edilen ve gerçek çıktılar arasındaki farkı en aza indirmek için ağırlıklarını ayarlayan geri yayılım gibi bir teknik kullanarak bu veri kümesi üzerinde DNN'yi eğitiriz.

DNN eğitildikten sonra, eğitim setinde olmayan yeni potansiyel fonksiyonlar için dalga fonksiyonunu tahmin etmek için kullanabiliriz. Bu, farklı sınır koşulları veya dış alanlar gibi farklı koşullar altında kuantum sistemlerinin davranışını tahmin etmek için yararlı olabilir.

Genel olarak, DNN'ler kuantum mekaniğindeki karmaşık problemleri çözmek için umut verici bir yaklaşım sağlar ve kuantum makine öğrenimi, birçok heyecan verici gelişmeyle birlikte aktif bir araştırma alanıdır.

2.2.1 Keras

Keras, kullanıcıların sinir ağlarını hızlı ve kolay bir şekilde oluşturmalarına ve eğitmesine olanak tanıyan, Python'da yazılmış üst düzey bir sinir ağı uygulama programlama arayüzü (API)'dir. Kullanıcı dostu ve modüler olacak şekilde tasarlanmıştır, kullanıcıların farklı ağ mimarileri ve eğitim algoritmaları ile hızlı bir şekilde prototip oluşturmalarına ve deneme yapmasına olanak tanır.

Fizikte Keras, görüntü ve sinyal işleme, zaman serisi analizi ve Schrödinger denklemi gibi diferansiyel denklemleri çözme gibi çeşitli uygulamalar için kullanılabilir.

Schrödinger denklemini çözmek için Keras'ı kullanmak için, kuantum makine öğreniminin bir biçimi olan sinir ağı kuantum durumları (NEURAL NETWORK QUANTUM STATE NQS) adı verilen bir teknik kullanabiliriz. NQS'de, bir kuantum sisteminin dalga fonksiyonunu öğrenmek için bir sinir ağı kullanıyoruz. Sinir ağı, sistemdeki parçacıkların koordinatlarını girdi olarak alır ve karşılık gelen dalga fonksiyonunu çıktı olarak verir.

Sinir ağını eğitmek için, girdilerin parçacık koordinatları ve çıktıların karşılık gelen dalga fonksiyonları olduğu girdi-çıkı çiftlerinden oluşan bir veri kümesi kullanabiliriz. Bu veri setini farklı potansiyeller için Schrödinger denklemini sayısal olarak çözerek ve elde edilen dalga fonksiyonlarını çıktı olarak kullanarak üretebiliriz.

Sinir ağı bu veri seti üzerinde eğitildikten sonra, eğitim setinde olmayan yeni potansiyeller için dalga fonksiyonunu tahmin etmek için kullanabiliriz. Bu, farklı sınır koşulları veya dış alanlar gibi farklı koşullar altında kuantum sistemlerinin davranışını tahmin etmek için yararlı olabilir.

Keras, çok çeşitli kullanılabilir katmanlar ve aktivasyon işlevleriyle sinir ağları oluşturmak ve eğitmek için kullanıcı dostu bir arayüz sağlar. Ayrıca, TensorFlow ve PyTorch gibi popüler makine öğrenimi kitaplıklarıyla da bütünleşerek onu fizik ve diğer alanlardaki çeşitli sorunları çözmek için çok yönlü bir araç haline getirir [13–16].

Schrödinger denklemi, kuantum mekaniğinde temel bir denklemdir ve parçacıkların davranışını açıklar. Analitik ve nümerik çözümler, bu denklemin çözülmesi ve sonuçların elde edilmesi için farklı yöntemlerdir.

Analitik Çözüm: Analitik çözüm, Schrödinger denklemini doğrudan matematiksel yöntemlerle çözmeyi ifade eder. Bu yöntem, denklemdeki bilinmeyen fonksiyonların kesin ifadesini bulmayı hedefler. Örneğin, bir parçacığın serbest hareketi durumunda Schrödinger denklemi, birinci dereceden bir diferansiyel denklem olan zaman bağımlı Schrödinger denklemi olarak ifade edilebilir. Bu denklemi analitik olarak çözebilir ve parçacığın dalga fonksiyonunu elde edebiliriz. Bu çözüm, sistemin tam ve kesin matematiksel betimlemesini sağlar [1, 17].

Nümerik Çözüm: Nümerik çözüm, Schrödinger denklemini matematiksel olarak çözmek yerine sayısal yöntemler kullanarak yaklaşık sonuçlar elde etmeyi ifade eder. Bu yöntem, genellikle analitik çözümün mümkün olmadığı durumlarda veya hesaplamaların karmaşık olduğu sistemler için kullanılır. Nümerik yöntemler, denklemi bir adım adım yaklaşım kullanarak çözer ve sonuçları elde eder. Bunun için örneğin, sonlu farklar yöntemi (Finite Difference Method - FMD) veya sonlu elemanlar yöntemi gibi yaklaşık türetilmiş denklemler kullanılabilir. Nümerik çözümler, hesaplama yoğunluğu daha yüksek olsa da daha genel sistemlerin analitik çözümlerinin bulunmasını sağlar [18–20].

Örnek olarak, bir parçacığın sonsuz potansiyel kuyuda (potansiyel enerjisi sınırlı bir bölgede sonsuz olmayan bir değere sahip) bulunduğunu düşünelim. Bu durumda, Schrödinger denklemi analitik olarak çözülebilir. Elde edilen analitik çözüm, dalga fonksiyonunu ve enerji seviyelerini tam olarak belirler.

Ancak, daha karmaşık potansiyel formlarında veya etkileşimli sistemlerde analitik çözümler genellikle mümkün değildir. Bu tür durumlarda nümerik yöntemler kullanılabilir. Örneğin, bir parçacığın bir potansiyel engelde hareketini

modellenmesini düşünelim. Bu durumda, sonlu farklar yöntemi veya sonlu elemanlar yöntemi gibi yöntemlere başvurulabilir.

Yapay zeka ve derin öğrenme algoritmalarının kullanılarak, istenilen çıktıların elde edilebilmesi için öncelikle kullanılacak olan yapay öğrenme algoritmasının eğitilmesi gerekmektedir. Eğitim verilerinin oluşturulabilmesi için öncelikle bu çalışmada Python programlama dili kullanılarak Schrödinger denkleminin nümerik çözümleri yapılmıştır. sonrasında ise bulunan bu nümerik çözümler, bilinen analitik çözümlerle karşılaştırılarak doğrulukları kontrol edilmiştir.

3.1 Schrödinger Denklemi'nin Farklı Potansiyele Değerleri İçin Nümerik Olarak Çözümlemesi

3.1.1 Türev Operatörleri ve Python'da Kodlanması

Denklem 3.1'de verilen Schrödinger denkleminin Python dilinde nümerik olarak çözülebilmesi için öncelikle denklemdeki operatörlerin matrisler halinde gösterilebilmesi gerekmektedir.

$$-\frac{\hbar^2}{2m} \frac{d^2}{dx^2} \psi(x) + V(x) \psi(x) = E \psi(x) \quad (3.1)$$

Bunu yapabilmek için öncelikle türev tanımının yapılması gerekmektedir. Denklem 3.2'de türev tanımı yapılmıştır. sonrasında ise denklem 3.3'de kullanılacak olan fonksiyon ve türevi alınmış olan fonksiyon gösterilmiştir.

$$\frac{d}{dx} f(x) = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x} \approx \frac{f(x + h) - f(x)}{h} + O(h) \quad (3.2)$$

$$f(x) = [f_0, f_1, f_2, \dots, f_{N-1}] \quad f'(x) = [f'_0, f'_1, f'_2, \dots, f'_{N-1}] \quad (3.3)$$

Buna göre fonksiyon ve türevi arasındaki ilişkiyi bulabilmek için denklem 3.2 kullanılmıştır ve böylece aralarındaki ilişki denklem 3.4'de gösterilmiştir.

$$f'_i = (f_{i+1} - f_i) / h \quad (3.4)$$

Denklem 3.4'de gösterilen ilişki kullanılarak bir fonksiyonun türevi ve kendisi matris formunda denklem 3.5'deki gibi ifade edilebilir.

$$\begin{pmatrix} f'_0 \\ f'_1 \\ \vdots \\ f'_{N-1} \end{pmatrix} = \frac{1}{h} \begin{pmatrix} -1 & 1 & 0 & 0 & \dots \\ -1 & 0 & 1 & 0 & \dots \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ \dots & 0 & -1 & 1 & 0 \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{N-1} \end{pmatrix} \quad (3.5)$$

Şimdiye kadar izlenilen yöntemlerin aynısını izleyerek ikinci türev için de matris formunda bir denklem elde edilebilmektedir. Bu amaçla denklem 3.7 kullanılarak işe başlanmıştır. Amaç burada tekrardan ikinci türevi ve fonsiyon arasında ilişkiyi bulup bunu matris formunda ifade edebilmektir. Bunun için daha önce elde edilen birinci türev ait denklemler burada denklem 3.7'de kullanılarak denklem 3.7 elde edilmiştir.

$$\frac{d^2}{dx^2}f(x) = \lim_{\Delta x \rightarrow 0} \frac{f'(x) - f'(x - \Delta x)}{\Delta x} \quad (3.6)$$

$$\frac{d^2}{dx^2}f(x) = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - 2f(x) + f(x - \Delta x)}{\Delta x^2} \approx \frac{f(x + h) - 2f(x) + f(x - h)}{h^2} \quad (3.7)$$

İkinci dereceden türev fonsiyonuna ait olacak türev matrisin yazabilmek için daha önce yapıldığı gibi öncelikle ikinci dereceden türev fonsiyonu 3.8'de tanımlanmıştır.

$$f''(x) = [f''_0, f''_1, f''_2 \dots, f''_{N-1}] \quad (3.8)$$

Son olarak ta denklem 3.3 ve denklem 3.7 kullanılarak fonsiyon ve ikinci türevi arasındaki ilişki denklem 3.9'de gösterilmiştir.

$$f''_i = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} \quad (3.9)$$

Böylece bir fonsiyonun kendisi ve ikinci türevi arasındaki ilişkiyi matris formunda ifade edebilmek için denklem 3.10 elde edilmiştir.

$$\begin{pmatrix} f''_0 \\ f''_1 \\ f''_2 \\ \vdots \\ f''_{N-1} \end{pmatrix} = \frac{1}{h^2} \begin{pmatrix} -2 & 1 & 0 & 0 & \dots \\ 1 & -2 & 1 & 0 & \dots \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ \dots & 0 & 0 & 1 & -2 \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_{N-1} \end{pmatrix} \quad (3.10)$$

3.1.2 Sonsuz Potansiyel Kuyusu

3.1.2.0.1 Nümerik Çözüm Sonsuz potansiyel kuyusu için potansiyel terimi 3.11'deki gibi yazılabilir. Bunu Python ile 3.1'deki gibi kodlanabilir.

$$V(x) = \begin{cases} \infty & x < -L \text{ ise} \\ 0 & -L < x < L \text{ ise} \\ \infty & x > L \end{cases} \quad (3.11)$$

```
def well_potential(x, L):  
    if x < -L or x > L:  
        return 1e+9  
    else:  
        return 0
```

Şekil 3.1 Sonsuz kuyu potansiyeli

Dalga fonksiyonlarının analitik çözümleri tablo 3.2'deki gibi olur.

3.1.2.0.2 Nümerik Çözümler ile Analitik Çözümlerin Karşılaştırılması

Bulduğumuz nümerik enerji değerlerinin analitik değerler ile karşılaştırılması tablo 3.1 görülmektedir.

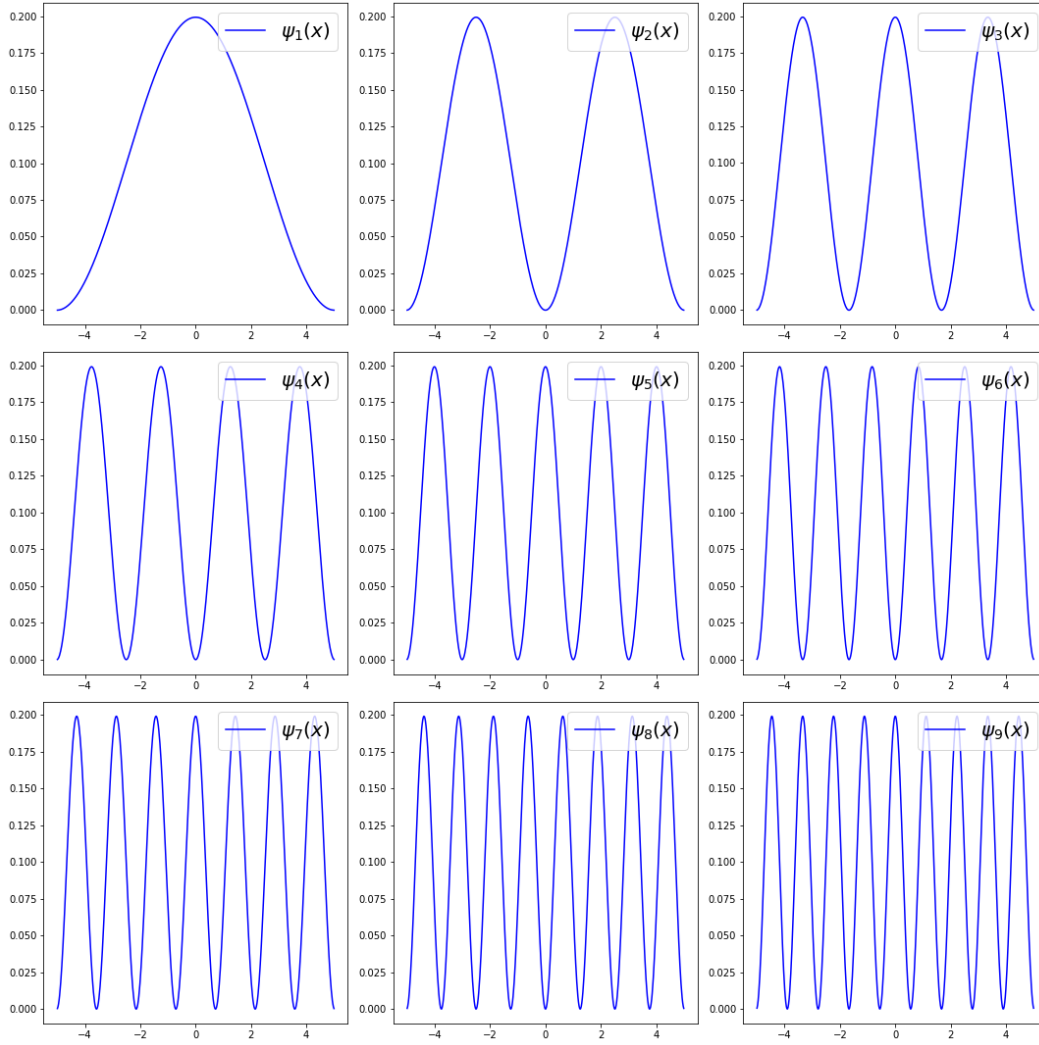
Tablo 3.1 Sonsuz potansiyel kuyu enerji çözümlerinin karşılaştırılması. Nümerik çözümler arasındaki fark %0.1'den azdır.

	Nümerik Çözüm	Analitik Çözüm
E_1	0.0490	0.0493
E_2	0.1958	0.1974
E_3	0.4406	0.4441
E_4	0.7832	0.7896
E_5	1.2238	1.2337
E_6	1.7622	1.7765
E_7	2.3984	2.4181
E_8	3.1325	3.1583
E_9	3.9643	3.9972

Nümerik dalga fonksiyonları ile analitik dalga fonksiyonlarının karşılaştırıldığı şekil 3.3'de verilmiştir.

3.1.2.1 Harmonik Salıncı

3.1.2.1.1 Nümerik Çözüm Harmonik salıncı potansiyeli 3.12'deki gibidir ve 3.4'deki gibi kodlanabilir.



Şekil 3.2 Sonsuz potansiyel kuyusu için elde edilmiş nümerik dalga fonksiyonları.

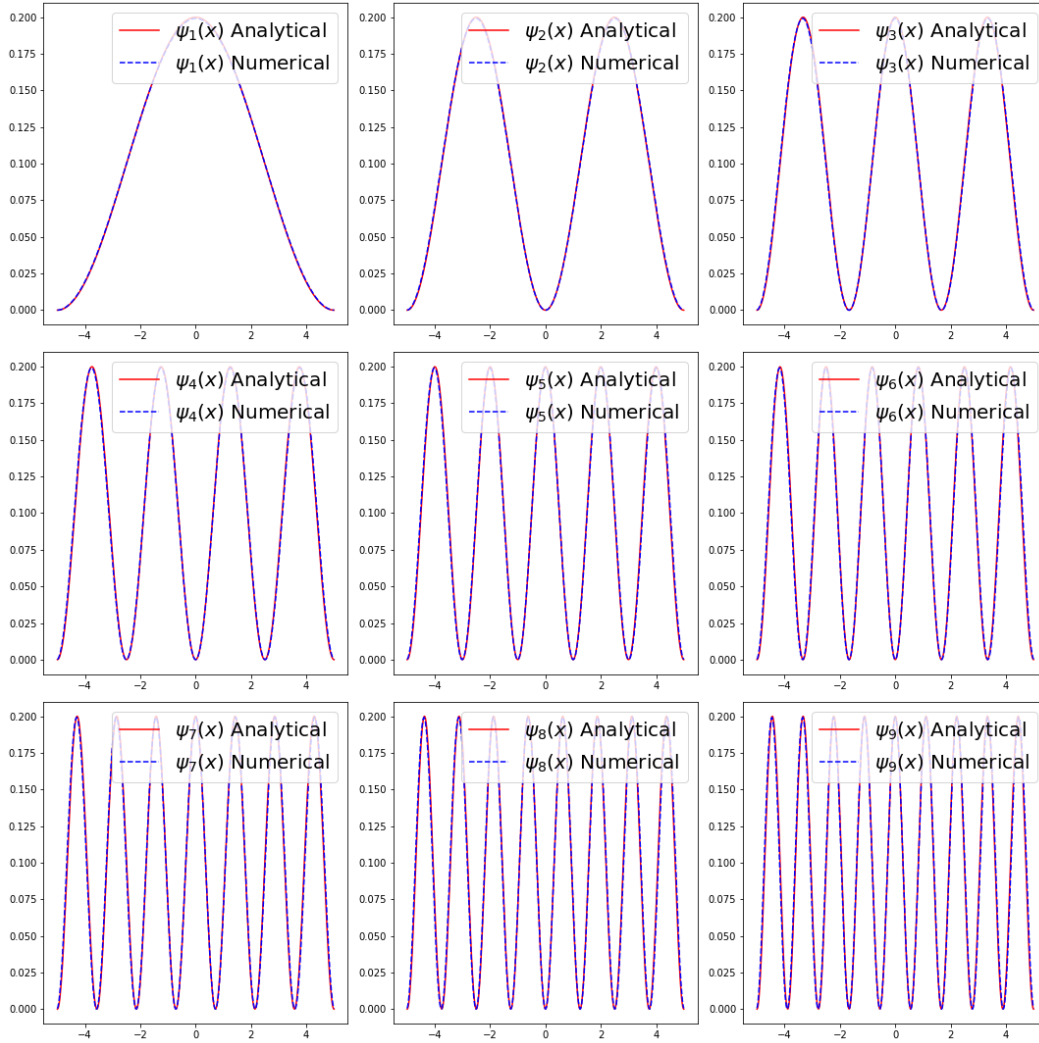
$$V(x) = \frac{1}{2}m\omega^2\hat{x}^2 \quad (3.12)$$

3.4'deki potansiyeli kullanarak Schrödinger denklemini nümerik olarak çözdürdüğümüzde 3.2'deki enerji değerlerini elde ederiz.

Tablo 3.2 Sonsuz kuyu potansiyeli için nümerik enerji çözümleri

Nümerik Enerji Çözümleri		
$E_0 = 0.4999$	$E_3 = 3.4987$	$E_6 = 6.4957$
$E_1 = 1.4997$	$E_4 = 4.4979$	$E_7 = 7.4943$
$E_2 = 2.4993$	$E_5 = 5.4969$	$E_8 = 8.4927$

Nümerik olarak elde edilen dalga fonksiyonları da 3.5'deki gibi olur.



Şekil 3.3 Sonsuz potansiyel kuyusu nümerik dalga fonksiyonları ile analitik dalga fonksiyonlarının karşılaştırılması

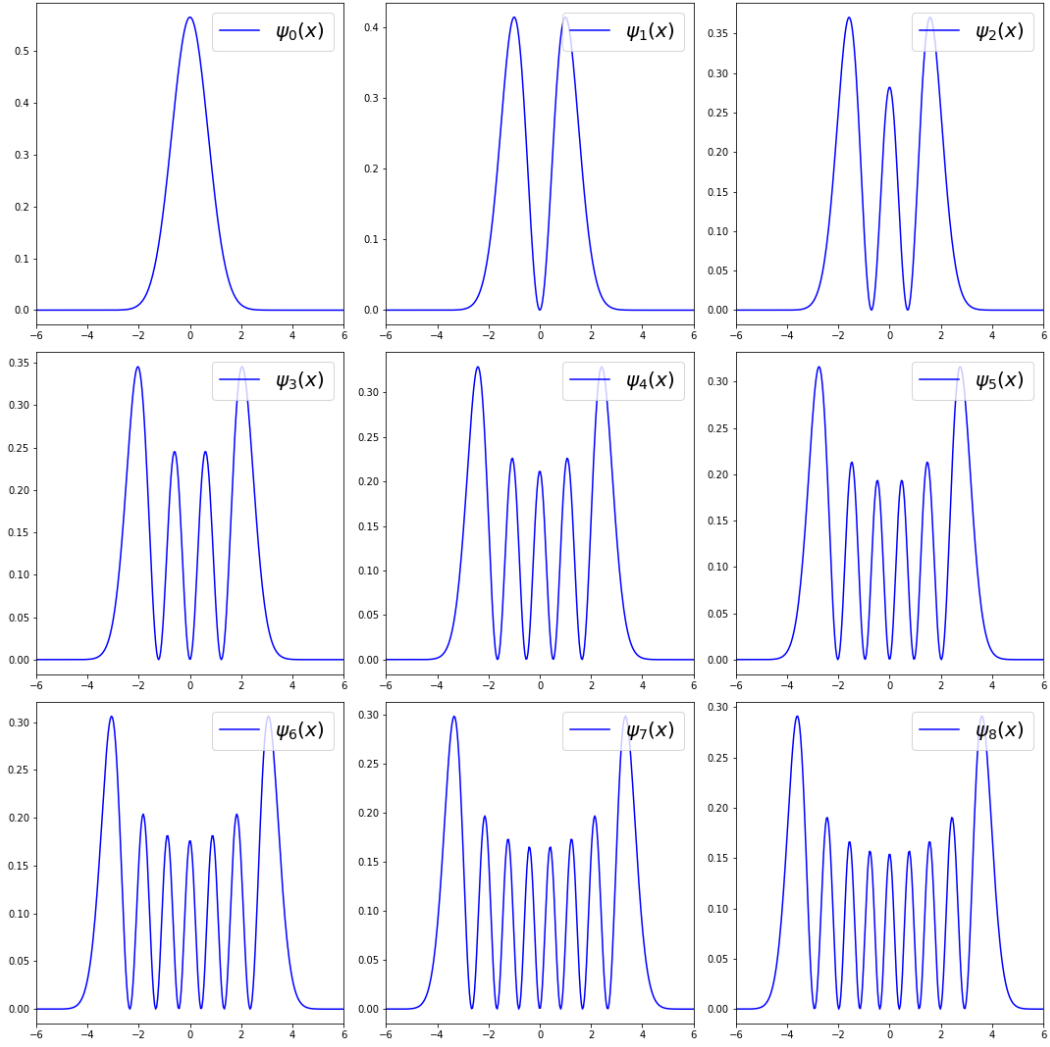
$$V = 0.5 * m * \omega^2 * x * x$$

Şekil 3.4 Harmonik salıncı potansiyeli

3.1.2.1.2 Nümerik Çözüm İle Analitik Çözümün Karşılaştırılması

Bulduğumuz nümerik enerji değerlerini analitik değerleri ile karşılaştırılması 3.3’de görülebilir.

Nümerik dalga fonksiyonları ile analitik dalga fonksiyonların karşılaştırıldığı grafik 3.6’de verilmiştir.



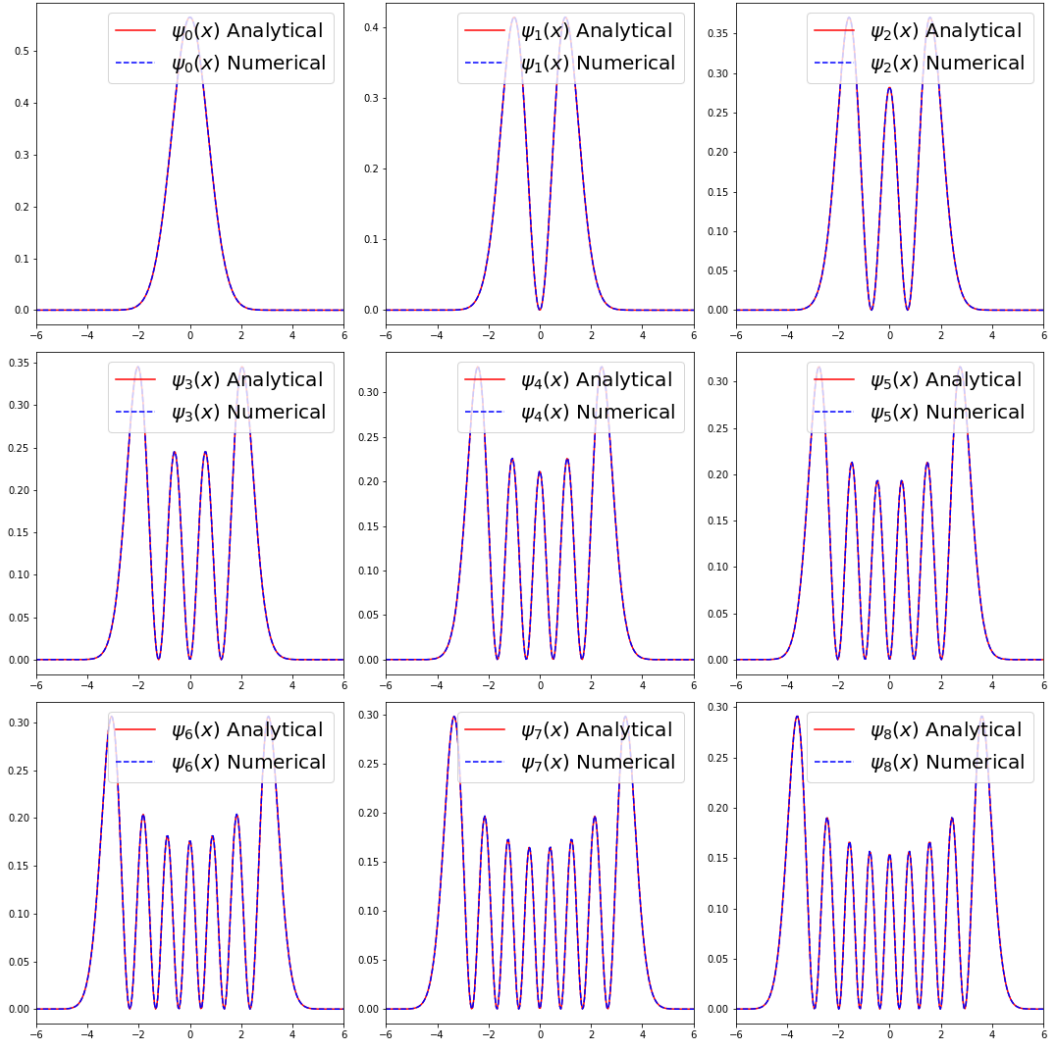
Şekil 3.5 Harmonik salıncı için nümerik olarak elde edilmiş dalga fonksiyonları.

Tablo 3.3 Harmonik osilator enerji çözümlerinin karşılaştırılması. Nümerik çözümler arasındaki fark %0.1'den azdır.

	Nümerik Çözüm	Analitik Çözüm
E_0	0.4999	0.5000
E_1	1.4997	1.5000
E_2	2.4993	2.5000
E_3	3.4987	3.5000
E_4	4.4979	4.5000
E_5	5.4969	5.5000
E_6	6.4957	6.5000
E_7	7.4943	7.5000
E_8	8.4927	8.5000

3.1.2.2 Doğrusal Potansiyel

3.1.2.2.1 Nümerik Çözüm Doğrusal potansiyelin 3.13'deki gibidir. Burada kolaylık olması açısından diye a katsayısı 1 olarak alınarak 3.7'deki gibi



Şekil 3.6 Harmonik osilatör için nümerik olarak elde edilmiş dalga fonksiyonları ile analitik olarak elde edilmiş dalga fonksiyonlarının karşılaştırılması.

kodlanmıştır.

$$V(x) = \begin{cases} \infty & x < 0 \text{ ise} \\ ax & 0 < x \end{cases} \quad (3.13)$$

```
def linear_potential(x):
    if x < 0 :
        return 1e+99
    else:
        c = 1
        return c*x
```

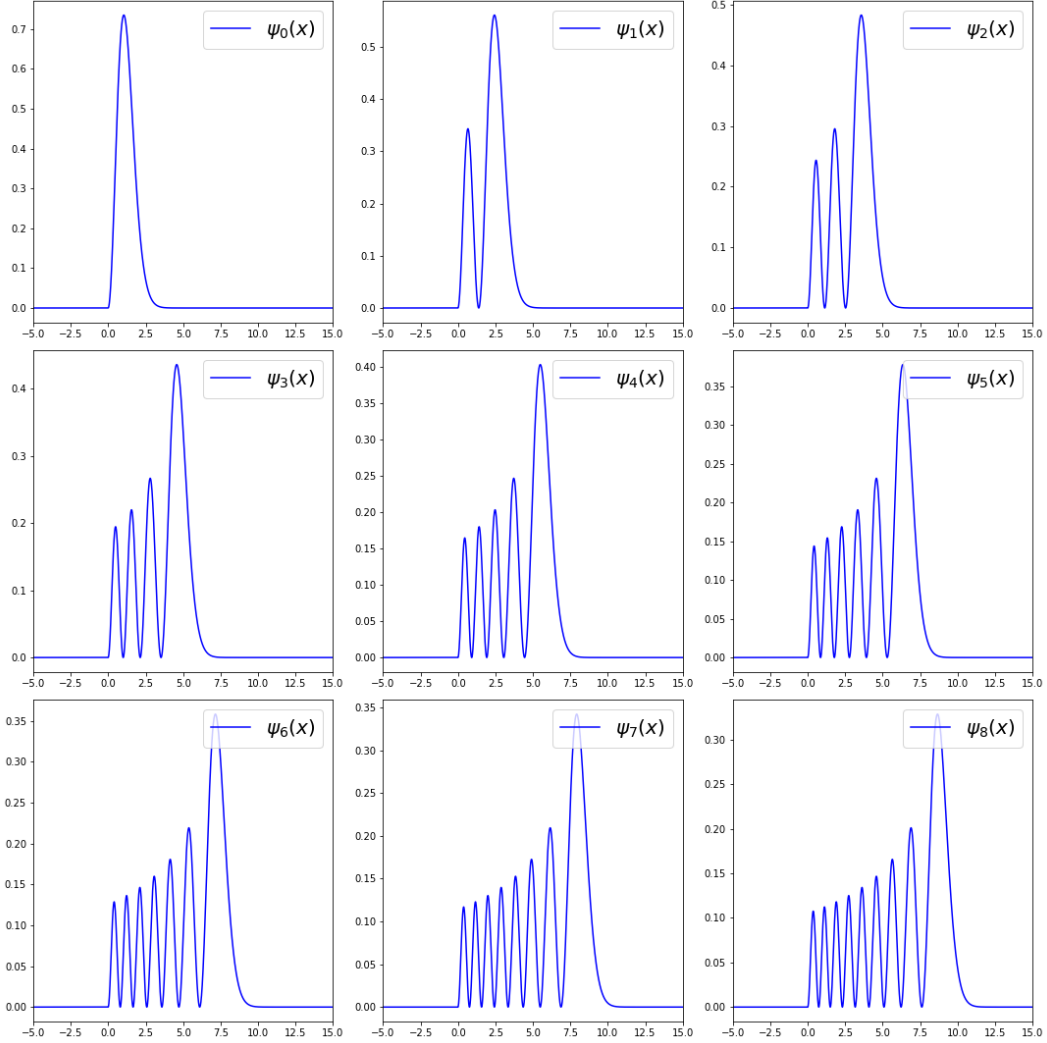
Şekil 3.7 Doğrusal potansiyel.

3.7'deki potansiyel kullanarak Schrödinger denklemi nümerik olarak çözüldüğünde 3.4'deki enerji değerleri elde edilmiştir.

Tablo 3.4 Doğrusal potansiyel için elde edilmiş nümerik enerji çözümleri.

Nümerik Enerji Çözümleri	
$E_0=$ 1.8517	$E_3=$ 5.3826
$E_1=$ 3.2406	$E_4=$ 6.3012
$E_2=$ 4.3776	$E_5=$ 7.1572

Yine nümerik olarak elde edilen dalga fonksiyonları da 3.8'deki gibi olur.



Şekil 3.8 Doğrusal potansiyel için nümerik olarak elde edilmiş dalga fonksiyonları.

3.1.2.2.2 Nümerik Çözüm İle Analitik Çözümün Karşılaştırılması Nümerik olarak elde edilen enerji değerlerinin analitik olarak elde edilen değerler ile karşılaştırılması 3.5 verilmiştir.

Tablo 3.5 Doğrusal potansiyel için enerji çözümlerinin karşılaştırılması. Nümerik çözümler ile arasındaki fark %0.1'den azdır.

	Nümerik Çözüm	Analitik Çözüm
E_0	1.8517	1.8558
E_1	3.2406	3.2446
E_2	2.4993	4.3817
E_3	5.3826	5.3866
E_4	6.3012	6.3053
E_5	7.1572	7.1613
E_6	7.9648	7.9689

3.1.3 Potansiyellerin Üretilmesi

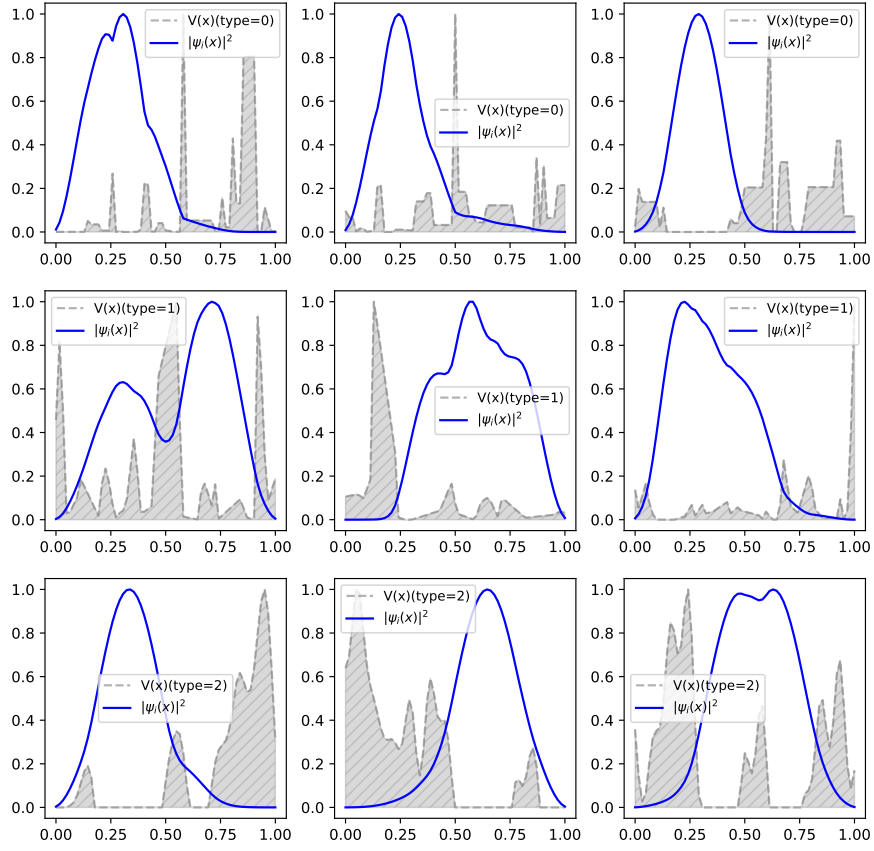
Genel çözüm üreten bir yapay zeka modeli eğitmek için hamiltonyendeki potansiyel kısmı için mümkün mertebe farklı potansiyeller kullanılmalıdır. Bunu gerçekleştirmek için önce *stil* ve *parametre* olmak üzere iki önemli değişken tanımlanmıştır. Stil'de potansiyeller step fonksiyonu, doğrusal fonksiyon ve Fourier serisi olarak 3 gruba ayrılmış ve sırasıyla 0, 1 ve 2 olarak tanımlanmıştır. Stil 0 ve 1 için atlama noktalarının sayısının ve stil 2 için Fourier üssünün belirlenmesi için *parametre* değerleri kullanılmıştır. Python'daki numpy kitaplığındaki numpy.random.uniform() ve numpy.random.randint() fonksiyonları kullanılarak potansiyellerin rastgele olması sağlanmıştır. Şekil 3.9'de üretilen rastgele bazı potansiyeller ve bu potansiyeller için nümerik olarak elde edilen dalga fonksiyonları görülebilir.

3.1.4 Derin Sinir Ağlarının Oluşturulması

Yapay sinir ağı modelleri Keras arayüzü kullanılarak yapılmıştır. Enerji ve dalga fonksiyonlarının tahmini için girdilerin rastgele potansiyeller olduğu ayrı iki model oluşturulmuştur. Önceden elde edilen potansiyeller modellere girdi olarak verilmeden önce standart ölçeklendirme ile dönüştürülmüştür.

3.1.4.1 Enerji Değerlerinin Bulunması

Energilerin bulunması için oluşturulan modelde 3 gizli katman bulunmaktadır. Her katmanın sonunda ReLU aktivasyon fonksiyonu kullanılmıştır. Ayrıca son katmanda da ReLU kullanılmıştır. Kayıp fonksiyonu (Loss Function) olarak MeanAbsolutePercentageError(), optimize edici olarak da Adam algoritması kullanılmıştır. İlgili Keras kod parçası 3.10'de görülebilir.



Şekil 3.9 Rastgele potansiyeller. İlk sıra stil 0 potansiyel, ikinci sıra stil 1 ve üçüncü sıra stil 2 potansiyeldir.

```

model = Sequential()
model.add(Dense(512, input_dim=input_size, activation='relu'))
model.add(Dense(256, activation='relu'))
model.add(Dense(512, activation='relu'))

model.add(Dense(output_size, activation='relu'))

model.compile(loss=tf.keras.losses.MeanAbsolutePercentageError(),
              optimizer='Adam',
              metrics=['accuracy'])
model.summary()

```

Şekil 3.10 Enerji değerlerinin bulunması için kullanılan model için Keras kod parçası.

3.1.4.2 Dalga Fonksiyonlarının Bulunması

Dalga fonksiyonlarının bulunması için oluşturulan modelde 9 gizli katman bulunmaktadır. Aktivasyon fonksiyonu olarak ReLu ve tanH fonksiyonları kullanılmıştır. Dalga fonksiyonları negatif değerler alabileceği için son katmanda aktivasyon fonksiyonu tanH olarak seçilmiştir. Kayıp fonksiyonu olarak kısmi dalga doğruluğu (partial wave fidelity) ismi verilen bir metrik kullanılmıştır. Kısmi dalga doğruluğu " K " şu şekilde tanımlanır:

$$K = \frac{|\langle \psi_{FDM} | \psi_{DNN} \rangle|^2}{|\langle \psi_{FDM} | \psi_{FDM} \rangle| |\langle \psi_{DNN} | \psi_{DNN} \rangle|}. \quad (3.14)$$

. Optimize edici olarak da yine Adam algoritması kullanılmıştır. İlgili Keras kod parçası 3.11’de görülebilir..

```

model = Sequential()
model.add(Dense(1024, input_dim=input_size, activation='relu'))
model.add(Dense(512, activation='tanh'))
model.add(Dense(512, activation='tanh'))
model.add(Dense(512, activation='tanh'))
model.add(Dense(512, activation='relu'))
model.add(Dense(512, activation='relu'))
model.add(Dense(512, activation='relu'))
model.add(Dense(512, activation='relu'))
model.add(Dense(1024, activation='relu'))
model.add(Dense(output_size, activation='tanh'))

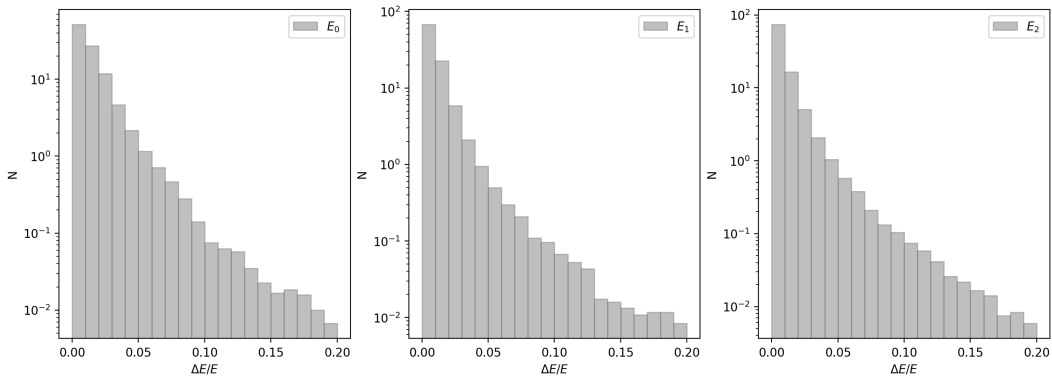
optimizer = tf.keras.optimizers.Adam(learning_rate=0.001)
model.compile(loss= 'mae', optimizer=optimizer, metrics=['accuracy'])
model.summary()

```

Şekil 3.11 Dalga fonksiyonlarının bulunması için kullanılan model için Keras kod parçası.

4.1 Eğitilen Modellerin Değerlendirilmesi

Energiler için ilk üç enerji öz değerinin ortalama bağıl enerji farkları karşılaştırılmıştır. Bütün enerjilerin ortalama bağıl enerji farkı 0.01285 çıkmıştır ve ilgili dağılım 4.1’de görülmektedir.

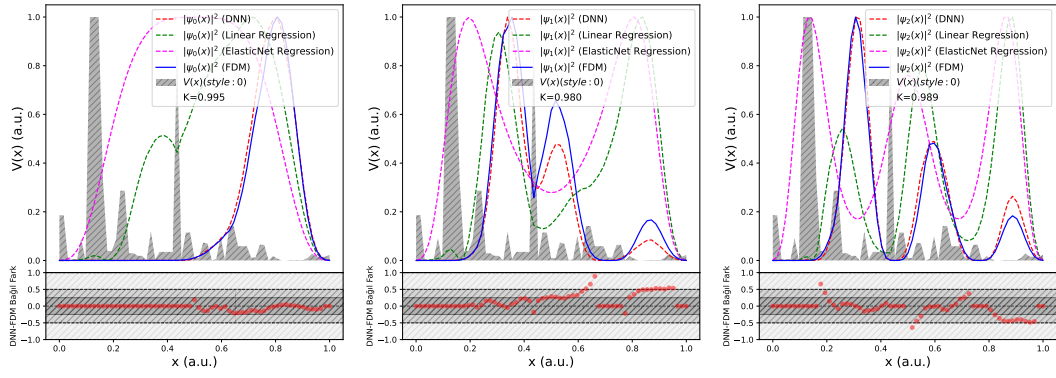


Şekil 4.1 Enerji öz değerleri arasındaki farkın dağılımı

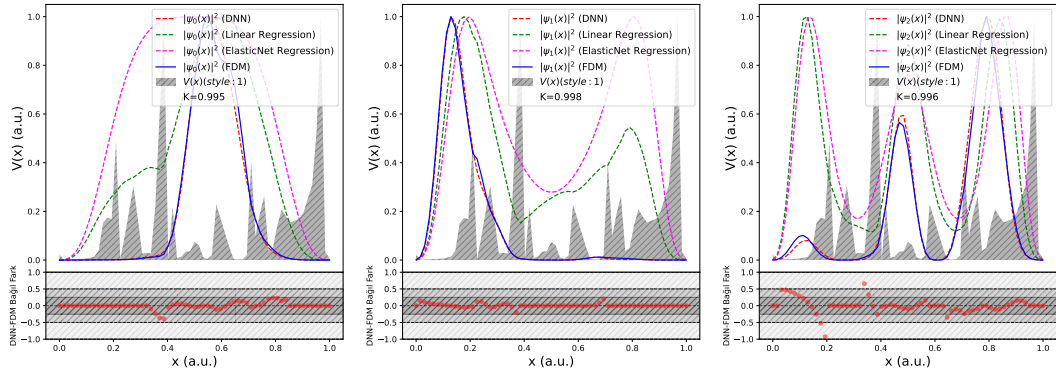
Farklı *stil* değerleri kullanılarak oluşturulan rastgele potansiyeller için, dalga fonksiyonları üzerine eğitilen modelden elde edilen fonksiyonların ve nümerik çözümünden elde edilen dalga fonksiyonların mutlak kareleri Şekil 4.2, 4.3 ve 4.4’de gösterilmiştir. Bu şekillerde ayrıca geleneksel makine öğrenmesi yöntemlerinden olan doğrusal regresyon (Linear Regression) ve ElasticNet Regresyon yöntemleri ile elde edilen dalga fonksiyonu tahminleri de gösterilmiştir. Bu yöntemlerin dalga fonksiyonlarını bulmada oldukça başarısız oldukları gözlemlenmiştir. Bu gözlem, böyle bir görev için derin sinir ağlarının kullanılması gerekliliğini işaret etmektedir.

Daha genel bir değerlendirme için DNN’den ve nümerik çözümünden elde edilen dalga fonksiyonları arasındaki açının kosinüs değerlerine bakılmıştır. Ortalama kosinüs değeri 0.9868’dir. 4.5’de dağılım grafiğini görülmektedir.

Sonuç olarak, bu çalışma, zamandan bağımsız Schrödinger denklemi ile



Şekil 4.2 Üst Panel: Dalga fonksiyonları üzerine eğitilen modelden elde edilen fonksiyonların ve nümerik çözümden elde edilen dalga fonksiyoların mutlak kareleri ile doğrusal regresyon (Linear Regression) ve ElasticNet Regresyon yöntemleri ile elde edilen dalga fonksiyonu tahminleri. Alt Panel: DNN ve nümerik çözüm arasındaki bağlı fark.

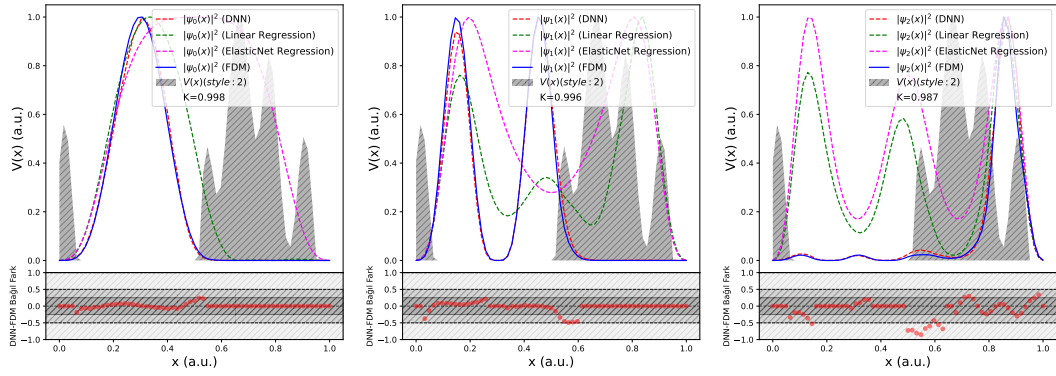


Şekil 4.3 Üst Panel: Dalga fonksiyonları üzerine eğitilen modelden elde edilen fonksiyonların ve nümerik çözümden elde edilen dalga fonksiyoların mutlak kareleri ile doğrusal regresyon (Linear Regression) ve ElasticNet Regresyon yöntemleri ile elde edilen dalga fonksiyonu tahminleri. Alt Panel: DNN ve nümerik çözümden arasındaki bağlı fark.

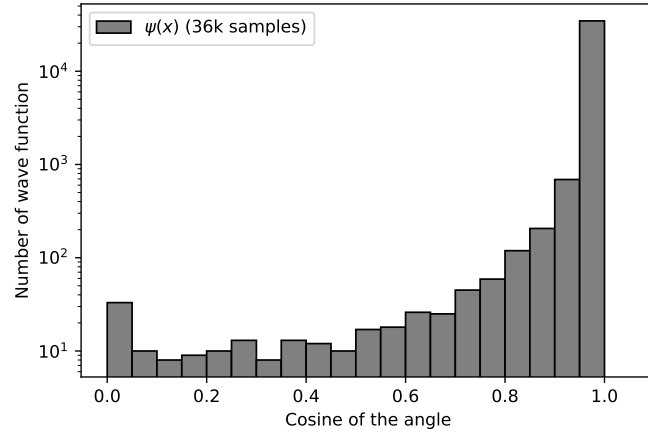
yönetilen sistemlerin farklı potansiyeller için davranışını Python programlama dili kullanarak analiz etmeyi amaçlamıştır. Schrödinger denklemini sayısal olarak çözerek, kuantum sistemlerinin davranışlarına ilişkin içgörüler elde edebildik ve sonuçlarımızın doğruluğunu değerlendirebildik.

Programlama ortamını hazırlama ve gerekli türev işlemlerini tanımlama sürecinde, Schrödinger denklemini uygulamak için temel attık. Bu, çeşitli potansiyeller oluşturmaya ve sayısal çözümler elde etmeye devam etmemizi sağladı. Sonuçları dikkatlice inceleyerek, farklı potansiyellere sahip sistemlerin sergilediği farklı davranışları gözlemleyebildik ve anlayabildik.

Çalışmamızın sonuçları, kuantum mekaniği sistemlerini keşfetmek için sayısal yöntemler ve Python gibi programlama dilleri kullanmanın etkinliğini



Şekil 4.4 Üst Panel: Dalga fonksiyonları üzerine eğitilen modelden elde edilen fonksiyonların ve nümerik çözümden elde edilen dalga fonksiyoların mutlak kareleri ile doğrusal regresyon (Linear Regression) ve ElasticNet Regresyon yöntemleri ile elde edilen dalga fonksiyonu tahminleri. Alt Panel: DNN ve nümerik çözüm arasındaki bağıl fark.



Şekil 4.5 Dalga fonksiyonları arasındaki açılarının dağılımı

vurgulamaktadır. Potansiyellerin ve bunlara karşılık gelen dalga fonksiyonlarının ayrıntılı bir analizini sağlayarak, bu yaklaşımın kuantum fenomeni anlayışımızı nasıl geliştirebileceğini gösterdik.

Ayrıca, bulgularımız bu alanda gelecekte yapılacak araştırmalar için değerli bir temel oluşturmaktadır. Burada sunulan metodoloji, daha karmaşık sistemleri araştırmak, ek fiziksel etkileri dahil etmek veya farklı potansiyel manzaraları keşfetmek için genişletilebilir. Sayısal yöntemlerin esnekliği ve ölçeklenebilirliği, daha fazla araştırma ve analiz için heyecan verici olanaklar sunar.

Özetle, bu çalışma Python programlama dilinde zamandan bağımsız Schrödinger denklemini ve sayısal teknikleri kullanarak kuantum sistemlerinin davranışına dair değerli bilgiler edinebileceğimizi göstermiştir. Sonuçlarımız, bu yaklaşımın daha fazla araştırma için bir temel oluşturma potansiyelinin altını çiziyor ve kuantum

mekaniğinin daha derin bir şekilde anlaşılmasına yönelik bir basamak taşı sunuyor. Bu çalışma aracılığıyla geliştirilen bilgi ve teknikler, alandaki daha geniş bilgi birikimine katkıda bulunur ve kuantum fiziğinde gelecekteki ilerlemeler için umut vaat eder.

- [1] D. J. Griffiths, D. F. Schroeter, *Introduction to quantum mechanics*. Cambridge university press, 2018.
- [2] E. Schrödinger, “Schrödinger 1926e,” *Annalen der Physik*, vol. 81, p. 109, 1926.
- [3] R. Shankar, *Principles of quantum mechanics*. Springer Science & Business Media, 2012.
- [4] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, G. Carleo, “Neural-network quantum state tomography,” *Nature Physics*, vol. 14, no. 5, pp. 447–450, Feb. 2018. doi: 10 . 1038 / s41567 – 018 – 0048 – 5. [Online]. Available: <https://doi.org/10.1038/s41567-018-0048-5>.
- [5] P. Huembeli, A. Dauphin, P. Wittek, “Identifying quantum phase transitions with adversarial neural networks,” *Physical Review B*, vol. 97, no. 13, p. 134 109, 2018.
- [6] D. Vedran, J. Briegel Hans, “Machine learning & artificial intelligence in the quantum domain,” *arXiv preprint arXiv:1709.02779*, 2017.
- [7] J. Behler, M. Parrinello, “Generalized neural-network representation of high-dimensional potential-energy surfaces,” *Physical review letters*, vol. 98, no. 14, p. 146 401, 2007.
- [8] G. Carleo, M. Troyer, “Solving the quantum many-body problem with artificial neural networks,” *Science*, vol. 355, no. 6325, pp. 602–606, 2017.
- [9] A. Geron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, 2nd. O’Reilly Media, Inc., 2019, isbn: 1492032646.
- [10] K. P. Murphy, *Machine learning : a probabilistic perspective*. Cambridge, Mass. [u.a.]: MIT Press, 2013, isbn: 9780262018029 0262018020. [Online]. Available: https://www.amazon.com/Machine-Learning-Probabilistic-Perspective-Computation/dp/0262018020/ref=sr_1_2?ie=UTF8&qid=1336857747&sr=8-2.
- [11] I. J. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016, <http://www.deeplearningbook.org>.
- [12] A. Burkov, *The Hundred-Page Machine Learning Book*. Andriy Burkov, 2019, isbn: 9781999579517. [Online]. Available: <https://books.google.com.tr/books?id=0jbxwQEACAAJ>.

- [13] E. Stoudenmire, D. J. Schwab, “Supervised learning with tensor networks,” *Advances in neural information processing systems*, vol. 29, 2016.
- [14] M. Raissi, P. Perdikaris, G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational physics*, vol. 378, pp. 686–707, 2019.
- [15] M. V. Altaisky, *Quantum neural network*, 2001. arXiv: quant - ph / 0107012 [quant-ph].
- [16] E. Alpaydin, *Introduction to machine learning*. MIT press, 2020.
- [17] J. J. Sakurai, E. D. Commins, *Modern quantum mechanics, revised edition*, 1995.
- [18] R. J. LeVeque, *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM, 2007.
- [19] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.
- [20] L. N. Trefethen, D. Bau III, *Numerical linear algebra*, vol. 50, 1997.

TEZDEN ÜRETİLMİŞ YAYINLAR

Konferans Bildirisi

1. Yapay Sinir Ağları Yardımıyla Schrödinger Denkleminin Nümerik Analizi, Emrah SAMUR ve Bora Işıldak, 8.ULUSLARARASI AKADEMİK ÖĞRENCİ ÇALIŞMALARI KONGRESİ