

**DİNAMİK SİSTEMLERİN KUANTUM SİNİR AĞLARIYLA
ANALİZİ**

Beşir OĞUR
091450102

DOKTORA TEZİ

Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Doktora Programı
Danışman: Prof. Dr. İhsan YILMAZ

İstanbul
T.C. Maltepe Üniversitesi
Lisansüstü Eğitim Enstitüsü
Temmuz, 2023

DİNAMİK SİSTEMLERİN KUANTUM SİNİR AĞLARIYLA ANALİZİ

Beşir OĞUR

091450102

ORCID: 0000-0003-3266-0233

DOKTORA TEZİ

Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Doktora Programı
Danışman: Prof. Dr. İhsan YILMAZ

İstanbul

T.C. Maltepe Üniversitesi
Lisansüstü Eğitim Enstitüsü
Temmuz, 2023



JÜRİ VE ENSTİTÜ ONAYI

Bu belge, Yükseköğretim Kurulu tarafından 19.01.2021 tarihli “Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge” ile bildirilen 6698 Sayılı Kişisel Verilerin Korunması Kanunu kapsamında gizlenmiştir.



ETİK İLKE VE KURALLARA UYUM BEYANI

Bu belge, Yükseköğretim Kurulu tarafından 19.01.2021 tarihli “Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge” ile bildirilen 6698 Sayılı Kişisel Verilerin Korunması Kanunu kapsamında gizlenmiştir.



TEŞEKKÜR

Doktora tez belirleme sürecinde tanışma fırsatı bulduğum, gerçekleştirdiğimiz bu tez çalışması için danışmanım olmayı kabul eden, akademik birikimi ve öngörüsü ile bilimsel düşünme perspektifimi sürekli geliştiren, bütün tıkanma süreçlerimde sabırla yolumu açan, insani nitelikleriyle örnek bir şahsiyet olan değerli hocam Prof. Dr. İhsan Yılmaz'a, yüksek lisans ve doktora öğrenciliğim boyunca bilimsel bilgi birikimi ve deneyimleriyle akademik sürecime ışık tutan saygıdeğer hocam Prof. Dr. İlhami Yavuz'a, katkılarından dolayı değerli kardeşlerim Erdi Acar ve Sedri Oğur'a, uzun süren tez çalışması boyunca benden desteğini esirgemeyen sevgili eşim Songül'e, tüm yaşamlarını çocuklarına adayan annem ve babama sonsuz teşekkürlerimi sunuyorum.

Beşir OĞUR

Temmuz, 2023

ÖZET

DİNAMİK SİSTEMLERİN KUANTUM SINIR AĞLARIYLA ANALİZİ

Beşir Oğur

Doktora Tezi

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Doktora Programı

Danışman: Prof. Dr. İhsan Yılmaz

Maltepe Üniversitesi Lisansüstü Eğitim Enstitüsü, 2023

Günümüzde kuantum hesaplama için önerilen algoritmalar ve üretilen kuantum bilgisayar teknolojileri gelişmeye devam etmektedir. Diğer taraftan makine öğrenmesi bilgisayarlı görü, doğal dil işleme, tahmin, sınıflandırma gibi, birçok problemin çözümü için başvurulan önemli yöntemlerden biri haline gelmiştir. Kuantum makine öğrenmesi, bu iki temel yöntemin avantajlı taraflarının birleştirilmesiyle geliştirilen yeni bir alandır. Kuantum ve klasik hesaplamaların hibrit bir yaklaşımı olarak varyasyonel kuantum devreleri, girdi değişkenlerine karşılık bir çıktı değerinin tahmin edilmesini sağlayan makine öğrenmesi formundadır. Bu çalışmada, süper pozisyonun ve dolaşıklığın etkileri, veri kümesi boyutunun küçük olması durumunda, varyasyonel kuantum devre modeli ile hava durumu tahmini araştırılmıştır. Varyasyonel katmanlar arasında dolaşıklık katmanının kullanılması devre performansı üzerinde önemli bir etkiye sahip olduğu görülmüştür. Başlangıçta süperpozisyon katmanının kullanılması daha az sayıda varyasyonel katman kullanmamıza olanak sağlamıştır.

Anahtar Sözcükler: Kuantum Hesaplama, Makine Öğrenmesi, Hava Durumu Tahmini, Varyasyonel Kuantum Devreleri, Hibrid Kuantum-Klasik Sinir Ağları

ABSTRACT

ANALYSIS OF DYNAMIC SYSTEMS WITH QUANTUM NEURAL NETWORKS

Beşir Oğur

PhD Thesis

Department of Computer Engineer

Computer Engineer PhD Programme

Thesis Advisor: Prof. Dr. İhsan Yılmaz

Maltepe University Graduate School, 2023

Recently, proposed algorithms for quantum computing and generated quantum computer technologies continue to evolve. On the other hand, machine learning has become an essential method for solving many problems such as computer vision, natural language processing, prediction and classification. Quantum machine learning is a new field developed by combining the advantages of these two primary methods. As a hybrid approach to quantum and classical computing, variational quantum circuits are a form of machine learning that allows predicting an output value against input variables. In this study, the effects of superposition and entanglement on weather forecasting, were investigated using a variational quantum circuit model when the dataset size is small. The use of the entanglement layer between the variational layers has made significant improvements on the circuit performance. The use of the superposition layer before the data encoding layer resulted in the use of less variational layers.

Keywords: Quantum Computing, Machine Learning, Weather Forecasting, Variational Quantum Circuit, Hybrid Quantum-Classic Neural Network

İÇİNDEKİLER

JÜRİ VE ENSTİTÜ ONAYI	ii
ETİK İLKE VE KURALLARA UYUM BEYANI	iii
TEŞEKKÜR.....	iv
ÖZET	v
ABSTRACT.....	vi
İÇİNDEKİLER	vii
TABLolar LİSTESİ.....	xii
ŞEKİLLER LİSTESİ	xiii
KISALTMALAR.....	xv
SEMBOLLER.....	xvi
1. GİRİŞ.....	1
1.1 Problem	3
1.2 Amaç	4
1.3 Önem	4
1.4 Varsayımlar	5
1.5 Sınırlıklar.....	5
1.6 Tanımlar	6
2. TEMEL LİTERATÜR.....	7
2.1 Klasik Hesaplama Teorisi	7
2.1.1 Church-Turing tezi ve evrensel makineler.....	7
2.1.2 Turing makinesinin formel tanımı	8
2.1.3 Evrensel Turing makinesi	9

2.1.4	Belirlenimci olamayan (nondeterministic) turing makinesi	9
2.1.5	Olasılıksal turing makinesi (OTM).....	9
2.2	Kuantum Hesaplamanın Temel Postulaları ve Kuantum Hesaplama Teorisi..	10
2.2.1	Kuantum mekaniğinin temel postulatları.....	11
2.2.1.1	Durum uzayı potulatu	11
2.2.1.2	Bileşik sistemler postulatı.....	11
2.2.1.3	Gözlenebilir büyüklüklerin operatörlerle gösterilme postulatı.....	12
2.2.1.4	Kuantum sistemlerin evrim postulatı.....	12
2.2.1.5	Ölçme postulatı.....	12
2.2.2	Kuantum hesaplama teorisi.....	14
2.2.2.1	Kuantum turing makinesi (KTM).....	15
2.2.2.2	Kuantum turing makinesinin (KTM) formel tanımı.....	15
2.2.2.3	Kara-kutu (Oracle/ Kahin) turing makinesi.....	16
2.2.2.4	Kara-kutu (Oracle/ Kahin) turing makinesinin formel tanımı.....	16
2.3	Hilbert Uzayı ve Dirac Notasyonu	17
2.3.1	Hilbert uzayı	18
2.3.2	Lineer bağımsızlık	20
2.3.3	Baz ve boyut	21
2.3.4	Tensörel çarpım	21
2.3.5	Dolaşıklık (Entanglement).....	23
2.3.6	Matrisler ve operatörler	24

2.3.7	Öz değerler ve öz vektörler.....	25
2.3.8	Spektral (tayfsal) ayrıştırma.....	26
2.3.9	Bir operatörün izi.....	27
2.3.10	Bir operatörün beklenen değeri.....	27
2.3.11	Operatörlerin fonksiyonları.....	28
2.3.12	Kübit ve kuantum bellek.....	29
2.3.12.1	Çoklu kübit gösterimi.....	30
2.3.12.2	Kübitlerin bellekte (register) saklanması.....	31
2.3.13	Kuantum kapılar.....	32
2.3.13.1	Tek kübit için kuantum kapıları.....	33
2.3.13.1.1	Hadamard kapısı.....	33
2.3.13.2	Çoklu kübit sistemi için kapılar.....	33
2.3.13.2.1	CNOT (XOR) kapısı.....	34
2.3.13.2.2	Kontrollü-U (Controlled-U) kapıları.....	34
2.3.13.2.3	Toffoli veya CCNOT kapısı.....	35
2.3.13.2.4	Çapraz geçiş veya takas kapısı (crossover or swap gate).....	35
2.3.13.2.5	Fredkin veya kontrollü takas kapısı (controlled-swap).....	36
2.4	Klasik Makine Öğrenmesi ve Yapay Sinir Ağları.....	36
2.4.1	Tanım ve özellikler.....	36
2.4.2	Temel işlem birimi nöron.....	37
2.4.3	YSA topolojileri.....	39
2.4.3.1	İleri beslemeli (feed-forward) YSA.....	39

2.4.3.2	Geril beslemeli (recurrent) YSA.....	40
2.4.3.3	Rekabetçi (Competitive) YSA.....	40
2.4.4	Çok katmanlı-ileri beslemeli YSA.....	41
2.4.5	YSA da öğrenme algoritmaları	41
2.4.5.1	Danışmanlı öğrenme (supervised learning)	42
2.4.5.2	Delta kuralı ile öğrenme	42
2.4.5.3	Geril yayma (backpropagation) kuralı ile öğrenme	43
2.4.6	Danışmansız öğrenme (Unsupervised Learning).....	45
2.4.7	Takviyeli öğrenme (Reinforcement Learning)	45
2.4.8	Derin öğrenme (deep learning).....	46
2.5	Kuantum Makine Öğrenmesi	46
2.5.1	Kauntum temel bileşen analiz (TBA).....	48
2.5.2	Kuantum faz tahmin algoritması.....	49
2.5.3	Takas (swap) testi	52
2.5.4	Kuantum destek vektör makinesi (DVM).....	53
2.5.5	Kuantum K-means kümeleme	55
2.5.6	Lineer cebir tabanlı kuantum makine öğrenmesi.....	57
2.5.6.1	HHL (Harrow, Hassidim ve Lloyd) algoritması.....	58
3.	ARAŞTIRMA YÖNTEMİ	60
3.1	Varyasyonel Kuantum Devresi	60
3.2	Veri Kümesi	62

3.3	Veri Önifleme	62
3.4	Deneyisel uygulamalar	62
3.5	Deney Prosedürü	67
4.	BULGULAR ve TARTIŞMA	68
5.	SONUÇ ve ÖNERİLER	73
	KAYNAKLAR	75
	ÖZGEÇMİŞ	85



TABLULAR LİSTESİ

Tablo 1. Dirac Notasyonları	18
Tablo 2. Biyolojik ve Yapay Nöron Bileşenlerinin Karşılaştırılması.	37
Tablo 3. Hava durumu veri kümesinin özellikleri	62
Tablo 4. Tüm Deneyler için test verilerinde ortalama kare hatası, R^2 Skoru ve Standart Sapma Sonuçları	74



ŞEKİLLER LİSTESİ

Şekil 1. Bloch Küresi.....	30
Şekil 2. YSA’ daki bir Nöronun Temel Elemanları.	38
Şekil 3. Aktivasyon Fonksiyonları.	39
Şekil 4. Geri Beslemeli YSA.....	40
Şekil 5. Rekabetçi YSA.....	41
Şekil 6. Çok Katmanlı YSA Modeli.....	41
Şekil 7. Derin Öğrenme YSA Modeli.	46
Şekil 8. Takas Testi Devresi.	53
Şekil 9. Destek Vektör Makinesi.....	54
Şekil 10. K-means Kümeleme.....	57
Şekil 11. Varyasyonel kuantum algoritmasının çalışma prensibi.....	60
Şekil 12. Temel kuantum varyasyonel devre modeli şeması.....	63
Şekil 13. Varyasyonel katmanda kullanılan farklı dolaşıklık ve rotasyon alt katmanlarının detaylı gösterimi.	64
Şekil 14. Deney 1’de kullanılan kuantum varyasyonel devre mimarisi.	64
Şekil 15. Deney 2’de kullanılan kuantum varyasyonel devre mimarisi.	65
Şekil 16. Deney 3’de kullanılan kuantum varyasyonel devre mimarisi.	65
Şekil 17. Deney 4’de kullanılan kuantum varyasyonel devre mimarisi.	66
Şekil 18. Deney 5’de kullanılan kuantum varyasyonel devre mimarisi.	66

Şekil 19. Deney 1 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi..... 68

Şekil 20. Deney 2 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi..... 69

Şekil 21. Deney 3 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi..... 70

Şekil 22. Deney 4 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi..... 71

Şekil 23. Deney 5 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi..... 72

.

KISALTMALAR

KTM	: Kuantum Turing Makinesi
OTM	: Olasılıksal Turing Makinesi
KTM	: Kuantum Turing Makinesi
AFD	: Ayrık Fourier Dönüşümü
KFD	: Kuantum Fourier Dönüşümü
YSA	: Yapay Sinir Ağları
İBA	: İleri Beslemeli Ağlar
GSA	: Geri Beslemeli Sinir Ağları
ÇKA	: Çok Katmanlı Ağlar
TBA	: Kuantum Temel Bileşen Analiz
DVM	: Destek Vektör Makinesi
K-DVM	: Kuantum Destek Vektör Makinesi
Ku-YSA	: Kuantum Yapay Sinir Ağı
KI-YSA	: Klasik Yapay Sinir Ağı
OKH(MSE)	: Ortalama Kare Hatası (Mean Squared Error)

SEMBOLLER

- δ : Turing makinesinde geiş fonksiyonu.
- Λ : Turing makinesinin formel tanımında boşluęu temsil eder.
- Σ : Turing makinesinin formel tanımında alfabeyi temsil eder.
- Q : Turing makinesinin formel tanımında durumların kümesini temsil eder.
- $\{M_m\}$: Kuantum mekanięinde ölçüm operatörlerinin kümesini tanımlar.
- $\langle \cdot | \cdot \rangle$: Hilbert uzayında bir durumun iç çarpımını temsil eder.
- $\sigma_0, \sigma_1, \sigma_2, \sigma_3$: Pauli kapılarını temsil ederler.
- $\|\cdot\|$: Bir vektörün normunu gösterir.
- \otimes : Tensörel çarpımı gösterir.
- \mathbb{C} : Karmaşık sayıları temsil eder.
- η : Geriye yayma algoritmasında öğrenme katsayısını ifade eder.

1. GİRİŞ

Doğa olaylarını veya günlük yaşamdaki gerçek problemleri çözerken temel amaç, en düşük maliyetle ve en kısa sürede istenen çözümün bulunmasıdır. Problemin doğasından kaynaklanan zorluk derecesine karşılık geliştirilen çözüm yöntemleri başarının ölçüsü olarak kabul edilir. Bu yaklaşım bilim ve teknolojinin gelişen bütün alanlarında olduğu gibi, hesaplama metotlarının evriminde de temel bir hedef olarak ortaya çıkmıştır.

Ortaçağda gündelik yaşamdan doğan ihtiyaçlara çözüm getirmek için kullanılan abaküsten, İngiliz Matematikçi Charles Babbage tarafından hesaplama sırasında ortaya çıkan insana ait hataları minimize etmek için tasarlanan mekanik bilgisayardan, 2.Dünya savaşında kullanılmak üzere Amerika Birleşik Devletleri tarafından geliştirilen ENIAC (*Electronic Numerical Integrator And Computer*)'dan günümüze kadar insanoğlu kendi ihtiyaçları doğrultusunda hesaplama yetenekleri olan makineleri (bilgisayarları) geliştirmeye devam etmiştir. Bu süreç modern hesaplama konseptinde yazılım, donanım ve bilgisayar iletişimleri şeklinde devam etmektedir.

Genel olarak hesaplama teorisi, cevaplanması gereken üç temel soru üzerinden şekillendirilebilir:

1. Karmaşıklık sınıfı itibariyle bir problem için 'hesaplanabilirlik' neyi ifade eder?
2. Verilen herhangi bir problemin hesaplanabilmesi için nasıl bir algoritma analizi yapılmalı?
3. Verilen bir problemin hesaplanabilmesi için ihtiyaç duyulan asgari kaynaklar (bellek, zaman ve enerji gibi) nelerdir?(Maheshwari ve Smid, 2012).

Hesaplama fiziksel bir süreç olarak tanımlanır. Klasik hesaplama, klasik fiziğin kuralları üzerinden inşa edilirken, kuantum hesaplama, kuantum mekaniğinin temel postülatları üzerinden temellendirilir(Desurvire, 2009; Nielsen ve Chuang, 2002). Klasik fizik yasaları ile anlaşılmayan, atom altı parçacıkların davranışlarını açıklamak üzere ortaya çıkan kuantum mekaniği, günümüzde kompleks problemleri çözebilme ve daha hızlı işlem yapabilme özellikleri açısından ihtiyaçlara cevap veremeyen Von Neumann (1993)

mimarisine sahip klasik bilgisayarlara alternatif bir hesaplama yöntemi ve bilgisayar mimarisini önererek inovatif bir alan yaratmıştır. Dolaşıklık ve süperpozisyon ilkeleri sayesinde elde edilen avantajlar, kuantum bilgisayarları daha cazip hale getirmiştir. Geliştirilen kuantum algoritmalar (Deutch-Jozsa algoritması (Deutch ve Jozsa, 1992) Grover'in arama algoritması (Grover, 1996, Temmuz), Shor'un asal çarpanlara ayırma algoritması (Shor P. W., 1994, Kasım)) ve 100 kübit sınırını aşan kuantum işlemciler, kuantum teknolojilerinin gelişimi için önemli adımlar olarak kabul görmektedir (Copeland B. J., 1997).

Makine öğrenmesi, veriler arasındaki ilişkileri ve örüntüleri kullanarak öğrenebilen bir modelledir. Klasik bilgisayarlar temel cebirsel işlemleri ve algoritmik olarak ifade edilebilen problemleri, insan beynine göre çok hızlı seviyelerde çözebilme yeteneğine sahiptir. Buna karşın, tahmin, sınıflandırma, ilişkilendirme, örüntü tanıma vb. insan beyni gibi düşünmeyi gerektiren konularda klasik bilgisayar algoritmaları oldukça verimsizdir. Makine öğrenmesi ile klasik bilgisayar programlama arasındaki en önemli fark kesin belirlenmiş kuralları olan algoritmalara gerek duymadan, örnekler üzerinden öğrenme sürecini gerçekleştirebilmesidir (Alpaydin, 2020; Dreyfus, 2005; Howard, Mark, ve Martin, 2004).

Hesaplama teorisi ve bu teoriye bağlı geliştirilen hesaplama aracı ne olursa olsun, temel amaç insan algısıyla gerçek dünyadaki problemleri en etkili yöntemle çözebilme çabasıdır. Gerçek dünyada karşılaştığımız problemlerin tamamına yakını dinamik bir sistem gibi davrandığından, dinamik sistem teorisi, temel bilimlerde (fizik, kimya, matematik ve biyoloji), mühendislik, ekonomi, hava tahmini ve tıp alanlarında problem analiz yöntemi olarak kullanılmaktadır.

Dinamik sistemlerin analizi ile ilgili ilk çalışmalar klasik mekaniğin kurucusu Newton tarafından yapılan üç temel hareket prensibi (eylemsizlik, dinamiğin temel prensibi, etki-tepki) ve genel kütle çekimi yasasıdır. Poincaré, periyodik olmayan ve başlangıç koşullarına hassasiyet gösteren kararlı sistemlerin kaotik bir davranış sergileyebileceğinden, sistemin davranışlarının uzun vadede neye evrileceğini kestirmenin mümkün olmadığını ortaya koymuştur.

Dinamik bir sistem soyut bir faz uzayının ya da durum uzayının başlangıç koşullarını esas alarak, bir sonraki adımda sistemdeki değişimin kurallarını ifade eder. Sistem değişimini belirleyen bu kurallar matematiksel olarak modellenir (Scheinerman, 1996; ve Wiggins, 2003).

1.1 Problem

Günümüzde kullandığımız modern konvansiyonel bilgisayarlar, hesaplama sürecinin bazı temel özellikleri Turing makinesi tarafından karakterize edilebilen, aritmetik ve mantık birimi (Arithmetic Logic Unit), kayıt ediciler (registers, sayaçlar) ve kontrol biriminden oluşan Von Neumann mimarisine sahiptirler (Godfrey, 1993). Moore yasasına göre “öngörülebilir bir gelecekte her yıl bütünleşik devrelerdeki transistör sayısı iki katına çıkacaktır.” Intel’in kurucularından olan Gordon E. Moore’un bu yaklaşımı, bilgisayar teknolojilerinin gelişmesiyle beraber üretilen daha küçük devreler sayesinde, aynı hacme fazla sayıda işlem gücü sıkıştırma mantığını özetlemektedir. Bilgisayarları küçültme teknolojisinin klasik bilgisayar mimarileri açısından yakın bir gelecekte atomik seviye sınırına dayanacağı kabul edilmektedir. Ayrıca donanımsal olarak gittikçe küçülen ve işlem yetenekleri bakımından hızlanan klasik bilgisayarlar hala karmaşıklık sınıfı NP (non-polinomial) olan problem tiplerini çözebilme yetenekleri bakımından da yetersizdirler. İleri sürülen bu nedenler klasik bilgisayarlara alternatif olacak yeni bir hesaplama teorisi ve bilgisayar mimarisi ihtiyacını doğurmaktadır.

Makine öğrenmesi için geliştirilen algoritmalar, klasik bilgisayarın işlem gücünün hızla artması ve katlanarak artan veri kümeleri ile makine öğrenmesinin sınıflandırma, regresyon, kümeleme gibi birçok alana uygulanması sağlanmıştır. Önemli yöntemlerden biri olan sinir ağları, makine öğrenmesindeki tahmin problemlerinde oldukça verimli sonuçlar vermektedir. Hava durumu tahmini, bu çerçevede önemli bir örnek problemdir.

Genel olarak hava tahmini olayları kararsız bir dinamik sistem olan atmosferin pertürbasyon etkileri nedeniyle sistemin değişen fiziksel özelliklerini belirli bir periyotta kestirebilme çalışmalarıdır. Tarım, ulaşım, turizm ve enerji talebi gibi önemli yaşamsal faaliyetler nedeniyle hava tahmini inceleme alanı itibarıyla araştırmacıların ilgisini cezbetmiştir. Yapılan bazı çalışmalarda, hava tahmini için yapay sinir ağlarıyla

geliştirilen yöntemlerin, geleneksel hava tahmini metotlarına (sayısal tahmin modelleri) göre daha iyi sonuçlar verdiği tespit edilmiştir (Holmstrom, Liu, ve Vo, 2016, Aralık).

Günümüzde belli periyotlarda (saatlik, günlük, haftalık ve aylık) yapılan hava durumu tahminleri ulaşım, tarım, lojistik vb. gündelik yaşamsal faaliyetler için önemli bir rol üstlenmektedir. Her yıl meydana gelen aşırı sıcak hava dalgaları ve kasırgalar binlerce ölüme ve maddi hasarlara sebep olmaktadır. Bu durumlar hava durumu tahminini önemli bir hale getiriyor. Hava durumu tahmini karmaşık ve dinamik doğası gereği zor bir problemdir. İnsanların karmaşık verilerle karşı karşıya kaldıklarında, tahmin için gerekli hesaplamaları yapması ve bunları hava durumunu tahmini için kullanması zordur.

Hava tahmini için kullanılan klasik algoritmaların tahmin performansı, veri setinin boyutu ile doğru orantılıdır. Veri kümesi boyutu küçük olduğu durumlar için birçok klasik yaklaşım kullanılabilir. Bu yaklaşımlara alternatif olarak kuantum bilgisayarlar ile makine öğreniminin kesişimi olan kuantum makine öğrenimi konusunda birçok araştırma yapılmaktadır (Biamonte ve Love, 2008; Schuld, Sinayskiy, ve Petruccione, 2015).

1.2 Amaç

Bu tez çalışması, yukarıda yapılan analizlere dayanarak, kuantum hesaplamasının iki önemli özelliği olan dolaşıklık ve süperpozisyonun, hibrit kuantum-klasik sinir ağı performansı üzerindeki etkisini, hava tahmini için, küçük boyutlu eğitim verileri üzerinde araştırmayı amaçlamaktadır. Ayrıca, kuantum devresi üzerinde farklı sayıda varyasyonel katman kullanımının hava tahmini performansı üzerindeki sonuçları karşılaştırılmıştır.

1.3 Önem

Bu çalışma kapsamında lokal bir alan olarak seçilen İstanbul-Kireçburnu/Sarıyer bölgesine ait 2016-2020 yılları arasındaki saatlik hava durumu verileri kullanılarak, bir regresyon problemi olan hava tahmininde, beş kübitlik kuantum varyasyonel devre modelinde süperpozisyon ve dolaşıklığın etkileri incelenmiştir. Bu çerçevede, süperpozisyon, dolaşıklık ve rotasyon katmanlarının farklı yerlerde kullanılmasıyla ilgili beş farklı deney yapılmış ve deney sonuçları bir tablo yardımıyla karşılaştırılmıştır.

1.4 Varsayımlar

Kuantum makine öğrenmesinin önemli bir modeli olarak kabul gören ve konvansiyonel makine öğrenmesi elemanları ile kuantum devrelerinin birleştirilmesiyle oluşturulan hibrit kuantum-klasik makine öğrenmesi, klasik makine öğrenmesine göre önemli avantajlar sağlamaktadır (Benedetti, Lloyd, Sack, ve Fiorentini, 2019; Cerezo, ve diğerleri, 2021; Schuld, Bergholm, Gogolin, Izaac, ve Killoran, 2019). Ayrıca, kuantum varyasyonel devre yaklaşımına göre geliştirilen hibrit kuantum-klasik algoritmaların pratik kullanımlar için hata tolerans limitleri açısından verimli sonuçlar verdiği bilinmektedir. Bu hibrit hesaplama yöntemi, pratik problemlerin çözümünde geliştirilmiş olan kuantum hesaplamanın yeteneklerini kullanmak için önemli fırsatlar sağlar (Preskill, 2018). Ek olarak, literatürdeki bazı ilgili çalışmalar, kuantum bilgisayarların hava tahmini için kullanıldığını vurgulamaktadır. Frolov (Frolov, 2017), klasik bilgisayarlar üzerinde sayısal yöntemlerle hava durumu ve iklim değişimlerini tahmin modellerinin geliştirilmesini engelleyen temel sınırlamaları inceleyerek, kuantum hesaplamalarının/kuantum bilgisayarların sayısal hava durumu ve iklim değişimine ilişkin tahmin problemlerini çözmek için uygulanabilir bir alan olarak araştırmıştır. Safari A. ve Ghavifekr A. A. (Safari ve Ghavifekr, 2021, Aralık), hava durumu tahmininde Kuantum Sinir Ağları yöntemini diğer yapay sinir ağı tabanlı teknikleriyle karşılaştırdılar. Karşılaştırma sonucunda, kuantum teknolojisinin ve kuantum sinir ağlarının diğer tekniklerle birleştirilme potansiyeline sahip olduğu, işlem hızı bakımından doğru modellerin elde edilebileceği sonucuna vardılar. Kuantum sinir ağları (Zubov, Volponi, ve Khosravy, 2015)'de hava tahmini araçları olarak kullanılmıştır. Deneysel sonuçlar, kuantum hesaplama tabanlı modellerin, klasik bilgisayarlarda uygulanan diğer makine öğrenimi tabanlı tahmin modellerinden daha iyi tahmin performansına sahip olduğunu ortaya koymaktadır.

1.5 Sınırlıklar

Genel olarak kuantum bilgisayarların ve kuantum simülasyon teknolojilerinin henüz istenen seviyeye ulaşamamış olması, kuantum hesaplama kapsamında yapılacak bütün çalışmalar için ciddi bir kısıtlama getirmektedir. Kuantum makine öğrenmesi kapsamında, klasik makine öğrenmesi modellerine benzer bir çok algoritma geliştirilmiştir. Bu anlamda önemli zorluklardan biri, yapay sinir ağlarının doğrusal

olmayan matematiksel yapısını, doğrusal işleyen kuantum hesaplama mimarisiyle entegre etmektir. Bu çalışmada bu entegrasyon sorununu çözen varyasyonel kuantum devre algoritması yöntem olarak seçilmiştir.

1.6 Tanımlar

Gerçekleştirilen bu tez çalışmasının daha iyi anlaşılması açısından önemli görülen bazı temel terimlerin tanımı verilmiştir:

Kuantum mekaniği: Işık ve mikroskobik evrendeki atom altı parçacıkların konum(\vec{x}), hız(\vec{v}), momentum(\vec{p}) ve enerji(E) gibi niceliklerini inceleyen fizik dalıdır.

Dolaşıklık: Özdeş olarak türetilen iki veya daha fazla sayıdaki atom altı parçacık arasında var olan bir kolerasyondur.

Süperpozisyon: Aynı boyutlu Hilbert uzayı üzerinde, normalisasyon şartını sağlamak üzere bir durumu, birden fazla alt durumun lineer kombinasyonu şeklinde temsil etmektir.

Hilbert uzayı: Üzerinde skaler çarpımın tanımlı olduğu, tamlık özelliğini içeren ve vektör normlarını koruyan sonsuz boyutlu bir sanal vektör uzayıdır.

Makine öğrenmesi: Kesin kurallarla belirlenmiş bir algoritma sürecine ihtiyaç duymadan, insan beynin düşünme şeklini esas alarak, örnekler üzerinden öğrenebilen bir hesaplama modelidir.

Varyasyonel kuantum devresi: Parametre alabilme özelliğine sahip birimsel kuantum operatörlerle oluşturulan, devre tasarımının geliştircinin tasarufunda bulunduğu bir devre modelidir.

2. TEMEL LİTERATÜR

2.1 Klasik Hesaplama Teorisi

Klasik Hesaplama Teorisi klasik fiziğin temel kuramları üzerinden inşa edilmiş mekanik bir süreçtir. Hesaplama teorisi zihinsel bir süreç olmayıp matematiksel olarak tanımlanan ve fiziksel olarak gerçekleşen bir süreçtir. Modern hesaplama yaklaşımına göre \mathcal{M} ile ifade edilebilen bir prosedürün istenilen sonuçları vermesi için dört temel özelliği içermesi gerekir:

1. \mathcal{M} sonlu sayıdaki kesin komutları temsil etmelidir. Her bir komut da sonlu sayıdaki semboller ile gösterilmelidir.
2. Eğer \mathcal{M} hatasız yürütülürse (koşturulursa) sonlu adımda her zaman istenen sonucu üretmelidir.
3. \mathcal{M} , pratikte veya prensipte, kalem ve kağıttan bağımsız olarak bir insan tarafından yardım alınmadan, herhangi bir makine tarafından yürütülebilir(koşturulabilir) özellikte olmalıdır.
4. \mathcal{M} , prosedürünü gerçekleştirirken insana ait bir düşünce veya yaratıcılık talep etmeden yürütmelidir (Copeland B. J., 1997)

2.1.1 Church-Turing tezi ve evrensel makineler

Hesaplama teorileri geliştirilirken en önemli amaç hesaplamayı gerçekleştirecek makineler için bir temel oluşturmak ve kuramsal olarak -hesaplamayı yapacak olan makinenin- sınırlarını belirlemektir. Böylece bir problemin çözülüp çözülemeyeceğini kararlaştıran bir algoritmanın matematiksel olarak tanımı zorunlu hale gelir. Bu zorunluluk, ilk kez Alman matematikçi David Hilbert'in 1900 yılında yayınladığı 23 problemten 10.ncusu olarak bilinen, katsayıları tamsayı olan çok değişkenli bir polinomun tamsayı köklerini sorgulayan bir algoritmanın varlığını araştırmak için ortaya atıldı. İngiliz matematikçi Alan Turing, bu problemi çözmek için *Turing makinesi* olarak

adlandırılan matematiksel bir soyutlama yaklaşımı geliştirdi (Williams ve Clearwater, 1998).

Turing Tezi : Her hangi bir matematiksel problem, algoritmik olarak ifade edilebiliyorsa mekanik olarak hesaplanabilir.

Church Tezi: Pozitif sayıların bir fonksiyonu rekürsif (özyinelemeli) ise etkili olarak hesaplanabilir.

Church-Turing Tezi, hesaplama modeli olarak Turing Makinesinden daha etkili bir hesaplama modelinin olmadığını söyler.

Church- Turing Tezi matematiksel bir soyutlama olduğundan fiziksel bir yorumu yoktur. Bu yorum daha sonra David Deutch tarafından yapılmıştır:

Fiziksel olarak gerçekleştirilebilen her sonlu sistem, sonlu araçlarla çalışan ve evrensel olarak modellenen bir hesaplama makinesiyle mükemmel olarak simüle edilebilir (Copeland B. J., 1997).

2.1.2 Turing makinesinin formel tanımı

Turing Makinesi tasarlanabilecek herhangi bir bilgisayardan daha fonksiyonlu bir modelledir. Dolayısıyla gerçek bir bilgisayarın yapabileceği her türlü hesaplamayı yapabilir. Turing Makinesi sınırsız bir band üzerindeki bellekte bir kafa yardımıyla sembolleri okuyabilir, yazabilir veya silebilir. Bir Turing Makinesinin sonlu sayıda durumu vardır ve makine herhangi bir anda bu durumların birisindedir. Hesaplamaya başlangıç durumu (start state) ile başlanır ve herhangi bir bitiş durumuna (halt state) gelindiğinde işlem sonlanmış olur.

Turing Makinesi 7-temel elemandan oluşur. $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{\text{kabul}}, q_{\text{ret}})$. Burada Q, Σ ve Γ ifadeleri sonlu kümeleri temsil etmektedir:

1. Q durumların kümesidir.
2. Σ boşluk sembolünü içermeyen giriş alfabesidir. (Boşluk sembolü Λ veya B dir).
3. $U \in \Gamma$ ve $\Sigma \subseteq \Gamma$ olmak üzere Γ bir band alfabesidir.

4. $\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ geçiş fonksiyonudur.
5. $q_0 \in Q$ başlangıç durumudur.
6. $q_{kabul} \in Q$ son durumdur (kabul edilen).
7. $q_{ret} \neq q_{kabul}$ olmak üzere $q_{ret} \in Q$ istenmeyen durumdur (Sipsler, 2006).

2.1.3 Evrensel Turing makinesi

Teorem: U_{TM} olarak tanımlanan bir Evrensel Turing Makinesinin varlığını kabul edelim. M bir Turing Makinesi ve n bir dizi olmak üzere $\langle M, n \rangle$, n üzerinden koşturularak M yi simüle eder.

Diğer bir ifadeyle Evrensel Turing Makinesi, her bir problem için farklı olarak önerilen bütün makineleri-kendisi de dahil olmak üzere- simüle edecek güçtedir. Bu kapsamda belirlenimci olmayan Turing Makinesi ve Olasılıksal Turing Makinesi (OTM) makinesi tanımlanacaktır.

2.1.4 Belirlenimci olmayan (nondeterministic) turing makinesi

Belirlenimci olmayan Turing Makinesi, standart Turing Makinesinin genelleştirilmiş formudur. Dolayısıyla bütün standart Turing Makineleri belirlenimci olmayan Turing Makinesi kümesinin doğal elemanıdır. Belirlenimci olmayan Turing Makinesinin temel elemanları, geçiş fonksiyonu dışında, standart Turing Makinesiyle aynıdır. Aşağıda formel ifadesi verilmiş olan geçiş fonksiyonu bir ağaç veri yapısı üzerinde tanımlanmıştır (Tusarova, 2004).

$$\delta: Q \times \Gamma \rightarrow \mathcal{P}(Q \times \Gamma \times \{L, R\}) \quad (2.1)$$

2.1.5 Olasılıksal turing makinesi (OTM)

Olasılıksal Turing makinesi belirlenimci olmayan (nondeterministic) bir makine olup, tek farkı sonraki adımını tahmin yerine ‘yazı-tura’ ile kararlaştırmasıdır. OTM de, bazı durum veya sembol kombinasyonları iki farklı geçişe özelliğine sahiptir ve bu geçişler arasındaki seçim olasılık yöntemiyle belirlenir. Her bir durum geçişini belirleyen, rastgele

seçilmiş bir kontrol biti mevcuttur. Rastgele seçilen farklı bitler, bazı olasılık değerleriyle farklı sonuçların ortaya çıkmasını sağlar (Williams ve Clearwater, 1999).

Tanım: ε nin $0 < \varepsilon < 1/2$ aralığında bir sabit olduğunu kabul edelim. Bir OTM olarak M , BPP (bounded-error probabilistic polynomial time) karmaşıklık sınıfına ait bir L diline ait x dizisini polinomial zamanda, en çok $p(|x|)$ adımda aşağıdaki şartlarda tanımlar:

$L(x) = 1 \rightarrow M$ makinesi ($\geq 1 - \varepsilon$) olasılık değeri ile ‘evet/kabul’ cevabını verir,

$L(x) = 0 \rightarrow M$ makinesi ($\leq \varepsilon$) olasılık değeri ile ‘hayır/ret’ cevabını verir (Kitaev, Shen, Vyalı, ve Vyalı, 2002).

2.2 Kuantum Hesaplamanın Temel Postulaları ve Kuantum Hesaplama Teorisi

1700 yıllarında Newton, klasik mekaniğin temel kuramlarını matematiksel modellemeleriyle beraber oluşturduktan sonra, makroskobik evrendeki nesnelere ait konum(\vec{x}), hız(\vec{v}), momentum(\vec{p}) ve enerji(E) gibi niceliklerin tamamı deterministik olarak tanımlanmış oldu (Guicciardini, 2005). Işık ve mikroskobik evrendeki atom altı parçacıklar için benzer incelemeler yapıldığında klasik Newton mekaniğinin yeterli olmadığı görülmüştür. Aşağıda incelenen bazı fiziksel çalışmalar, klasik fizik yerine yeni bir yaklaşım olarak ‘Kuantum Mekaniği’ nin ortaya çıkmasını zorunlu kılmıştır: -Siyah cisim ışınması, Max Planck (1900): Herhangi bir sıcaklıktaki bir cisim, moleküler titreşimlerden dolayı kesikli enerji yayar. Bu enerji paketçikleri ‘kuant’ olarak ifade edilir.

- Fotoelektrik etki, Albert Einstein (1905): Elektromanyetik dalgaların veya ışığın etkisiyle bir metal yüzeyden elektronların koparılmasıdır.
- Bohr Atom Modeli, Niels Bohr (1913): Planck’ın kesikli enerji fikrine dayandırdığı kuantum mekaniği kapsamında bir atom modeli önermiştir.
- Compton saçılması, Arthur Compton (1923): Bir karbon atomuna gönderdiği x-ışınlarının herhangi bir elektronla çarpıştıktan sonra saçılma esnasında bir tanecik gibi davrandığını ispatlamıştır.

- Madde ve dalga ilişkisi, Louis de Broglie (1924): de Broglie doktora tezi çalışmasında fotonların ikircikli davranış modelinin (hem dalga hem de tanecik) bütün maddeler için genelleştirebileceğini ortaya koydu.
- Dalga denklemi, Erwin Schrödinger (1926): Bir kuantum sistemine ait toplam enerjiyi dikkate alarak, sistemin davranışını ifade eden genel dalga denklemini kurdu.
- Belirsizlik İlkesi, Werner Heisenberg (1927): Bir atom altı parçacığının konumu ve momentumunun aynı anda kesinlikte belirlenemeyeceğini ifade etti (Gasirowicz, 2007; Griffiths ve Schroeter, 2018; Sakurai ve Commins, 1995).

2.2.1 Kuantum mekaniğinin temel postulatları

Kuantum hesaplama teorisi, kuantum mekaniğinin temel ilkeleri üzerinde inşa edildiğinden, Kuantum mekaniğinin temel postülatlarını incelemek bu tez çalışması için önemlidir.

2.2.1.1 Durum uzayı potulatu

Bir kuantum sisteminin durumu Hilbert Uzayında, zamanla değiştiği kabul edilen $|\Psi(t)\rangle$ birim vektörü ile gösterilir. Bu vektör mevcut kuantum durumu ile ilgili elde edilebilecek bütün bilgileri içerir. Kuantum hesaplamalarda $\langle\Psi|\Psi\rangle = 1$ şeklinde normalize edilmiş durum vektörleri ile işlem yapılır.

Bir kübit iki boyutlu kompleks Hilbert Uzayında $|\alpha|^2 + |\beta|^2 = 1$ normalizasyon şartını sağlayan ve

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (2.2)$$

denklemleri ile ifade edilen bir durum vektörüdür (McMahon, 2007; Nielsen ve Chuang, 2002).

2.2.1.2 Bileşik sistemler postulatu

Kauntum mekaniğinde bir bileşik fiziksel sistemin durum uzayı, kendisini oluşturan alt fiziksel durum uzayı bileşenlerinin tensörel çarpımına eşittir. 1'den n 'e kadar numaralandırılmış alt fiziksel durumların oluşturduğu bileşik sistem $|\Psi_i\rangle \in H_i$ olmak üzere,

$$|\Psi_{bileşik}\rangle = |\Psi_1\rangle \otimes |\Psi_2\rangle \otimes \dots \otimes |\Psi_n\rangle \quad (2.3)$$

olur. Burada $|\Psi_i\rangle$ fiziksel sistemlerin durumunu ve H_i durum uzaylarını gösterir (McMahon, 2007; Nielsen ve Chuang, 2002).

2.2.1.3 Gözlenebilir büyüklüklerin operatörlerle gösterilme postulatı

Kuantum mekaniğinde, dinamik bir fiziksel sistemin bütün gözlenebilir büyüklüklerine bir hermitiyen operatör karşılık gelir. Gözlenebilirler konum, momentum, açısal momentum ve enerji gibi ölçülebilir büyüklükleri temsil eder. Operatörler bir $f(x)$ fonksiyonuna etki ederek yeni bir fonksiyon olan $g(x)$ nun, ya da bir durum üzerine etki ederek yeni bir durumun elde edilmesini sağlayan işlemcilerdir (McMahon, 2007).

2.2.1.4 Kuantum sistemlerin evrim postulatı

Kapalı bir kuantum sisteminin evrimi üniteri bir değişimle tanımlanır. Yani, kapalı bir kuantum sisteminin durum vektörünün evrimi lineerdir. Vektör normlarını koruyan lineer operatörler sadece birimsel operatörlerdir.

t_1 anında $|\Psi\rangle$ ile ifade edilen bir kuantum sisteminin durumu, U üniter operatörünün uygulanmasıyla t_2 anında $|\Psi_1\rangle$ durumuna dönüşür.

$$|\Psi_1\rangle = U|\Psi\rangle \quad (2.4)$$

Kapalı bir kuantum sisteminin zamanla evrimi Schrödinger denklemiyle de gösterilebilir:

$$i\hbar \frac{d|\Psi\rangle}{dt} = H|\Psi\rangle \quad (2.5)$$

Burada $i = \sqrt{-1}$, \hbar Planck sabiti ve H sistemin Hamiltoniyeni olup genel enerji durumunu temsil eder (Nielsen ve Chuang, 2002).

2.2.1.5 Ölçme postulatı

Kuantum mekaniğinde ölçme işlemi hem tersinir, hem de deterministik değildir. Ölçme işleminde fiziksel gözlenirlerin ortaya çıkma olasılığı, karşılık geldikleri hermitik operatörlerin her zaman reel olan öz değerleridirler. Yapılan ölçümler, $\{M_m\}$ ölçüm operatörlerinin kümesiyle tanımlanır. Bu operatörler sistem üzerinde ölçüm yapmak için

durum uzaylarına uygulanır. Burada m indeksi deneysel olarak olası sonuçları temsil eder. Eğer kuantum sistemin durumu ölçümden hemen önce $|\Psi\rangle$ ise ölçümden sonra m sonucunun ortaya çıkma olasılığı;

$$p(m) = \langle \Psi | M_m^\dagger M_m | \Psi \rangle \quad (2.6)$$

dir. Sistemin ölçümden sonraki durumu;

$$\frac{M_m |\Psi\rangle}{\sqrt{\langle \Psi | M_m^\dagger M_m | \Psi \rangle}} \quad (2.7)$$

olur. Ölçme operatörleri tamlık denklemini sağlarlar.

$$\sum_m M_m^\dagger M_m = I \quad (2.8)$$

Tamlık denklemini durumların ortaya çıkma olasılıkları toplamının '1' olması gerektiğini söyler.

$$1 = \sum_m p(m) = \sum_m \langle \Psi | M_m^\dagger M_m | \Psi \rangle \quad (2.9)$$

Kuantum hesaplamada en çok kullanılan ölçme yöntemleri aşağıdaki gibi açıklanabilir:

Yöntem 1: Kabul edelim ki $|V_1\rangle, |V_2\rangle, \dots, |V_d\rangle$ ortonormal bazların bir formu ve $\{M_i = |v_i\rangle\langle v_i|\}$ ise bir kuantum ölçme olsun. $|\Psi\rangle$ durumu üzerinde yapılan bir ölçme işleminden $|\langle v_i | \Psi \rangle|^2$ olasılığı ile aşağıdaki sonuç elde edilir:

$$\frac{|v_i\rangle\langle v_i | \Psi \rangle}{|\langle v_i | \Psi \rangle|} \quad (2.10)$$

Tanım: Bir projektör 0 ve 1 özdeğerleriyle bir hermitik matristir. 0 ve 1 özdeğeriyle alt uzay bu operatörle ilişkilendirilir (McMahon, 2007).

Yöntem 2: Kabul edelim ki S_1, S_2, \dots, S_k durum uzaylarını kapsayan (geren) ortogonal alt uzaylar olsun. P_i, S_i üzerinde bir projektör olmak üzere $\{P_i\}$ bir kuantum ölçüm işlemidir. $|\Psi_i\rangle \in S_i$ olmak üzere;

$$|\Psi\rangle = \alpha_1 |\Psi_1\rangle + \alpha_2 |\Psi_2\rangle + \dots + \alpha_k |\Psi_k\rangle \quad (2.11)$$

denklemleri yazılabilir. $|\Psi\rangle$ üzerinde yapılan ölçme işlemi $|\alpha_i|^2$ olasılıkla $|\Psi_i\rangle$ sonucunu verir (Nielsen ve Chuang, 2002).

2.2.2 Kuantum hesaplama teorisi

Kuantum Hesaplama Teorisi, klasik hesaplama teorisinin klasik fiziğin temel kuramlarına dayandırılması benzer şekilde, kuantum mekaniğinin temel kuramları üzerine inşa edilmiştir. Matematiksel bir soyutlama olarak tanımlanan ‘hesaplanabilirlik’ in gerçekleştirilmesi fiziksel bir işlemdir. Dolayısıyla her hesaplama teorisi kendi fiziksel temelleri ile gerçekleştirilir.

Kuantum hesaplama teorisi, klasik hesaplama sürecinin akabinde kendi içinde bir gelişim seyrine girdi:

- İlk kuantum bilgisayar önerisi 1960 yılında yapılan "There's Plenty of Room at the Bottom" konferansında Richard Feynman tarafından ortaya atıldı. Kuantum mekaniği kuramlarının hesaplama için yeni bir yaklaşım olarak kullanılabileceğini önerdi (Feynman, 1960).
- 1980 yılında Paul Benioff, kuantum hesaplama ile ilgili; fiziksel bir sistem olarak Turing makinesi tarafından temsil edilen bilgisayarların kuantum mekaniği ölçeğindeki Hamiltoniyen modelini içeren ilk makalesini yayınladı (Benioff, 1980).
- Yuri Manin, kuantum bilgisayarların gelişimine ışık tutan “Computable and Uncomputable” kitabını yayınladı (Manin, 1980).
- MIT’de 1981’de ‘Hesaplamanın Fiziği’ üzerine düzenlenen konferansta Benioff kuantum mekaniği yasalarına göre çalışabilen bilgisayarları tanımladı. Aynı konferansta Feynman kuantum bilgisayar için bir model önerdi (Vasconcelos, 2020).
- 1992’de Deutsch-Josza algoritması David Deutsch ve Richard Josza tarafından önerildi (Deutsch ve Jozsa, 1992).
- 1994 yılında Peter Shor, tam sayıları çarpanlarına ayırma süresini polinomial seviyeye indirgeyen ve kendi adıyla anılan algoritmasını önerdi (Shor P. W., 1994, Kasım).

– 1996 yılında Lov Grover tarafından geliştirilen Grover algoritması, sıralanmamış bir veri tabanı üzerinde random bir datayı aramak üzere geliştirilmiş bir kuantum algoritmasıdır (Grover, 1996, Temmuz).

– David P. DiVincenzo, fiziksel bir kuantum bilgisayarını gerçekleştirmek için beş temel gereksinimi önerdi (DiVincenzo, 1997).

2.2.2.1 Kuantum turing makinesi (KTM)

Kuantum Turing Makinesi (KTM), farklı özellikleriyle Olasılıksal Turing Makinesinin iyileştirilmiş halidir. KTM de adımları ifade eden katsayılar olasılıksal değil, belirli bir genlik değerine karşılık gelen karmaşık sayılardan oluşur. Her adımda genlik olasılıklarının normunun karelerinin toplamı '1' e eşittir. Her giriş değerine karşılık gelen geçiş matrisi üniteri olmak zorundadır.

Ağaç veri yapısına sahip olduğu varsayılan KTM, bütün olası yolların (dalların) final/çıkış düğümüne ulaşması durumunda durduğu kabul edilir. Çıkış değeri, başlangıç durumundan başlanarak ilk boşluk sembolünü gösteren tape üzerine yazılır. Çıkış değerinin oluşturulma olasılığı söz konusu duruma karşılık gelen genliğin normunun karesini alma işlemidir.

2.2.2.2 Kuantum turing makinesinin (KTM) formel tanımı

Bir KTM altı temel elemanla tanımlanır ($\Sigma, \Lambda, Q, q_b, q_s, \delta$):

1. Σ bütün olası sembolleri içeren sonlu bir küme olup alfabe olarak ifade edilir.
2. $\Lambda \in \Sigma$ olmak üzere boşluk sembolünü gösterir.
3. Q sonlu bir küme olup, durumların kümesini tanımlar.
4. q_b başlangıç durumunu gösterir.
5. q_s son durumu gösterir.
6. $\delta: \Sigma \times Q \rightarrow H$ geçiş fonksiyonunu gösterir ve H ise $\Sigma \times Q \times \{L, R\}$ üçlüsüne karşılık gelen baz vektörlerini kapsayan Hilbert Uzayıdır. Bütün giriş değerleri için sonlu geçiş matrisleri üniteridir.

Bir KTM de bir dilin kabulü, olasılık yorumlarının değişmesine bağlı olarak yeniden tanımlanmalıdır (Tusarova, 2004).

2.2.2.3 Kara-kutu (Oracle/ Kahin) turing makinesi

Bir KTM'e kara-kutu hesaplama özelliği kazandırılırken, makinede üzerinde yapılan sorgu süreçleri durumların üniteri yapısını korumalıdır. Bu da yapılan hesaplamının tersinir olması gerektiğini açıklar. Eğer yapılan sorgunun cevabı 'evet' ise kara-kutu mevcut biti/durumu değiştirir, 'hayır' ise mevcut biti/durumu değiştirmez, olduğu gibi bırakır. Diğer bir ifadeyle, eğer üzerinde sorgu yapılan bandın içeriği sağdaki ilk boşluk sembolü için $|x, b\rangle$ ise sonraki sorgu durumu,

$$|x, b \oplus I(x)\rangle = \begin{cases} I(x) = 1 \rightarrow x \in O \\ I(x) = 0 \rightarrow \text{diğer durumlar} \end{cases} \quad (2.12)$$

şeklinde olur (Tusarova, 2004).

2.2.2.4 Kara-kutu (Oracle/ Kahin) turing makinesinin formel tanımı

Kara-kutu Turing Makinesinin sekiz temel bileşeni vardır $(\Sigma, \Lambda, Q, q_b, q_s, \delta, q_q, q_a)$. Her bir elemanın işlevi aşağıda tanımlandığı gibidir:

1. $\Sigma = (0, 1, \Lambda)$ alfabe olarak tanımlanan sonlu elemanlı bir kümedir.
2. Λ boşluk sembolünü gösterir.
3. Q sonlu bir küme olup, durumların kümesini tanımlar.
4. q_b başlangıç durumunu gösterir.
5. q_s son durumu gösterir.
6. $\delta: \Sigma \times Q \rightarrow (\Sigma \times Q \times \{L, R\})^{|\Sigma| \cdot |Q| \cdot 2}$ ifadesi geçiş fonksiyonunu ifade eder.
7. $q_q \in Q$ bir özel sorgu durumunu temsil eder.
8. $q_a \in Q$ bir özel post-sorgu durumudur (Tusarova, 2004).

2.3 Hilbert Uzayı ve Dirac Notasyonu

Klasik mekanikte bir parçacığa ait olan konum ve momentum gibi temel fiziksel büyüklükler kesin olarak tanımlanabilir. Yani klasik fizik deterministik/belirlenimci olarak işler. Kuantum mekaniğinde ise bir parçacığa ait olan konum ve momentum gibi fiziksel büyüklükler kesin olarak belirlenemez, ancak belirli bir olasılıkla kestirilebilir. Yani kuantum mekaniği olasılıksal işler.

Kuantum mekaniğinin matematiksel modellemesini gerçekleştirmek için gerekli olan kompleks vektör uzayı ve bu uzay üzerinde işlem gören lineer operatörler, lineer cebir tarafından sağlanır. Bu nedenle kuantum mekaniğini anlayabilmek için lineer cebirin temel matematiksel kuramlarını, özellikle sonlu kompleks Hilbert Vektör Uzayını iyi incelemek gerekir.

Klasik fizikteki vektörel bir büyüklüğün \vec{A} notasyonu ile temsiline benzer şekilde, kuantum mekaniğindeki fiziksel bir durumun vektörel gösterimi Dirac Notasyon'u olarak 'ket' $|\Psi\rangle$ ile temsil edilir. Dirac Notasyonları ile ilgili temel gösterimler ve bunlar arasındaki işlemlerin özeti tablo 1'de verilmiştir (McMahon, 2007; Nielsen ve Chuang, 2002).

Tablo 1.Dirac Notasyonları

Notasyon Gösterimi	Notasyon Tanımı
$z = a + ib$	Kompleks sayı
$z^* = a - ib$	Kompleks sayı eşleniği
$ \Psi\rangle$	Ket vektörü
$\langle\Psi $	Bra vektörü. Ket vektörünün dualidir.
$\langle\Phi \Psi\rangle$	$ \Phi\rangle$ ve $ \Psi\rangle$ vektörlerinin iç çarpımını gösterir.
$ \Phi\rangle \otimes \Psi\rangle$	$ \Phi\rangle$ ve $ \Psi\rangle$ vektörlerinin tensör çarpımıdır.
$ \Phi\rangle \Psi\rangle$	Tensör çarpımının kısa gösterimi.
A^*	A Matrisinin kompleks eşleniğidir.
A^T	A Matrisinin transpozesidir.
$A^\dagger = (A^T)^*$	A Matrisinin hermisyenidir.
$\langle\Phi A \Psi\rangle$	$ \Phi\rangle$ ve $A \Psi\rangle$ ifadelerinin iç çarpımıdır. $A^\dagger \Phi\rangle$ ve $ \Psi\rangle$ ifadelerinin iç çarpımına da eşittir.
$\ \Psi\ $	Vektör normunu gösterir.

2.3.1 Hilbert uzayı

Tanım 1: Bir V lineer vektör uzayı bir F alanı (field) üzerinde aşağıdaki işlem özelliklerini taşıyan bir kümeyi temsil eder (Desurvire, 2009):

1. Toplama işlemi: $\forall |\Phi\rangle, |\Psi\rangle \in V$ için $|\Phi\rangle + |\Psi\rangle \in V$ dir.
2. Bir skalerle çarpma işlemi: $\forall |\Psi\rangle \in V, \forall \lambda \in F$ olmak üzere $\lambda|\Psi\rangle = |\Psi\rangle\lambda \in V$ dir.

V uzayı üzerinde tanımlanan toplama işlemi ve bir durumun herhangi bir skalerle çarpma özelliği aşağıdaki aksiyomları sağlamalıdır:

1. Toplamada deęişme özellięi: $|\Phi\rangle + |\Psi\rangle = |\Psi\rangle + |\Phi\rangle$
2. Toplamada birleşme özellięi: $(|\Phi\rangle + |\Psi\rangle) + |\varphi\rangle = |\Phi\rangle + (|\Psi\rangle + |\varphi\rangle)$
3. Sıfır elemanını içirme özellięi: $|sıfır\rangle \in V$ ve $\forall |\Psi\rangle \in V$ olmak üzere;
 $|\Psi\rangle + |sıfır\rangle = |\Psi\rangle$ olur. ($|sıfır\rangle \neq |0\rangle$)
4. Toplama işleminde göre ters eleman içirme özellięi: $|\Phi\rangle + |\Psi\rangle = |sıfır\rangle$ ise
 $|\Phi\rangle = -|\Psi\rangle$ olur.
5. Vektörlerin toplamada dağılma özellięi: $\lambda(|\Phi\rangle + |\Psi\rangle) = \lambda|\Phi\rangle + \lambda|\Psi\rangle$
6. Skaler toplamında dağılma özellięi: $(\lambda + \mu)|\Psi\rangle = \lambda|\Psi\rangle + \mu|\Psi\rangle$
7. Skaler çarpımında birleşme özellięi: $\lambda(\mu|\Psi\rangle) = (\lambda\mu)|\Psi\rangle$
8. Çarpma işleminde birim eleman: $1 \in F, \forall |\Psi\rangle \in V$ olmak üzere $1 \cdot |\Psi\rangle = |\Psi\rangle$ olur.

Tanım 2: V bir kompleks vektör uzayı ve $\forall |\Phi\rangle, |\Psi\rangle \in V$ olmak üzere aşağıdaki özellikleri sağlayan bir iç (skaler) çarpım $\langle \cdot | \cdot \rangle$ tanımlanabilir ($\langle \Phi | \Psi \rangle \in \mathbb{C}$):

1. $\langle \varphi | (|\Phi\rangle + |\Psi\rangle) \rangle = \langle \varphi | \Phi \rangle + \langle \varphi | \Psi \rangle$
2. $\langle \Phi | (\lambda|\Psi\rangle) \rangle = \lambda \langle \Phi | \Psi \rangle$
3. $\langle \Phi | \Psi \rangle = \langle \Psi | \Phi \rangle^*$
4. $\langle \Psi | \Psi \rangle$ işleminin değeri reel bir sayıdır, $\langle \Psi | \Psi \rangle \geq 0$, $\langle \Psi | \Psi \rangle = 0 \Leftrightarrow |\Psi\rangle = 0$.

Skaler çarpım aynı zamanda vektör normunu da tanımlar: $\| |\Psi\rangle \| = \| \Psi \| = \sqrt{\langle \Psi | \Psi \rangle}$.
Vektörlerin norm tanımına baęlı olarak Schwarz ve Üçgen eşitsizlikleri aşağıdaki gibi tanımlamır:

$$\| |\Phi\rangle + |\Psi\rangle \| \leq \| |\Phi\rangle \| + \| |\Psi\rangle \| \quad (\text{Üçgen eşitsizliği}) \quad (2.14)$$

Dual vektörlerin kullanımıyla iç çarpım daha kolay yapılabilir. $\langle \rho | \mu \rangle$ iç çarpımı aşağıdaki gibi ifade edilir (Nielsen ve Chuang, 2002):

$$\langle \rho | \mu \rangle = (\rho_1^* \ \rho_2^* \ \dots \ \rho_n^*) \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \end{pmatrix} = \rho_1^* \mu_1 + \rho_2^* \mu_2 + \dots + \rho_n^* \mu_n = \sum_{i=1}^n \rho_i^* \mu_i \quad (2.15)$$

Tanım 3 (Tamlık Teoremi): Kabul edelim ki V , $\|\cdot\|$ normu ile bir vektör uzayı ve $|\Psi_n\rangle \in V$ bir vektör serisi olsun:

1. $\forall n, m > N$, $\| |\Psi_n\rangle - |\Psi_m\rangle \| < \epsilon$ olmak üzere, eğer $\forall \epsilon > 0$ ve $\exists N > 0$ ise $|\Psi_n\rangle$ bir Cauchy serisidir.

2. $\forall \epsilon > 0, \exists N > 0, \forall n > N$ ve $\| |\Psi_n\rangle - |\Psi\rangle \| < \epsilon$ olmak üzere bir $|\Psi\rangle \in V$ ifadesi mevcut ise $|\Psi_n\rangle$ serisi yakınsaktır.

Eğer bütün Cauchy serileri yakınsak ise V tamlık teoremini sağlar (McMahon, 2007).

Tanım 4: Tamlık özelliğini içeren bir vektör uzayı, skaler çarpım $\langle \cdot | \cdot \rangle$ ve buna karşılık gelen vektör normu $\| |\Psi\rangle \| = \sqrt{\langle \Psi | \Psi \rangle}$ ile ‘Hilbert Uzayı’ olarak tanımlanır (Bertlmann, 2023).

2.3.2 Lineer bağımsızlık

Tanım 1 (Lineer Dönüşüm): V ve W vektör uzaylarının F alanı(field) üzerinde tanımlı olduklarını kabul edelim. Bütün $a, b \in F$ ve $u, v \in V$ için $f: V \rightarrow W$ lineer dönüşüm (2.16) daki gibidir (Bertlmann, 2023).

$$f(au + bv) = af(u) + bf(v) \quad (2.16)$$

Tanım 2 (Lineer bağımsızlık): Kabul edelim ki $v_1, \dots, v_n \in V$ olmak üzere V sonlu vektör uzayı F alanı üzerinde tanımlı olsun. $\alpha_i \in F$ katsayılarından en az birinin sıfırdan farklı olması şartıyla aşağıdaki denklem lineer bağımlıdır.

$$\alpha_1 v_1 + \dots + \alpha_n v_n = 0 \quad (2.17)$$

Lineer bağımlı olmayan vektör uzayı, lineer bağımsızdır. Bu durum $\alpha_i = 0$ şartının sağlanmasıyla,

$$\alpha_1 v_1 + \dots + \alpha_n v_n = 0 \quad (2.18)$$

denklemini ile mümkündür (Desurvire, 2009).

Tanım 3 (Vektörlerin lineer kombinasyonları): $\alpha_i \in \mathbb{C}$ kompleks katsayılar ve $|V_i\rangle \in V$ bir vektör kümesi olmak üzere ;

$$\alpha_1 |v_1\rangle + \alpha_2 |v_2\rangle + \dots + \alpha_n |v_n\rangle = \sum_{i=1}^n \alpha_i |v_i\rangle \quad (2.19)$$

ifadesi vektörlerin lineer kombinasyonudur (Desurvire, 2009; Nielsen ve Chuang, 2002).

2.3.3 Baz ve boyut

Tanım 1: V bir vektör uzayı ve $S = \{v_1, v_2, \dots, v_n\}$ V de tanımlanmış bir vektör kümesi olmak üzere aşağıdaki şartları sağlaması durumunda S vektör kümesi baz olarak kabul edilir;

1. S kümesi lineer bağımsızdır.
2. S kümesi V vektör uzayını (türetir)gerer.

Tanım 2: Sıfırdan farklı V vektör uzayının baz kümesi $\{v_1, v_2, \dots, v_n\}$ sonlu sayıda vektör içeriyorsa sonlu boyutlu, sonsuz sayıda vektör içeriyorsa sonsuz boyutlu vektör uzayı olarak ifade edilir.

Sonlu boyutlu V vektör uzayının boyutu, baz vektör kümesinin vektör sayısıdır. $\dim(v)$ şeklinde de gösterilir (McMahon, 2007).

2.3.4 Tensörel çarpım

Kuantum mekaniğinde her zaman izole edilmiş tek parçacık üzerinde çalışılmaz. Bazı durumlarda çoklu parçacıklarla ilgilenmemiz gerekebilir. Bu durumlarda tensörel çarpım, uzayları, vektörleri ve operatörleri birleştirme yöntemi olarak kullanılır.

H_1 ve H_2 sırasıyla n ve m boyutlu iki Hilbert uzayı olmak üzere, $H_1 \otimes H_2$ tensörel çarpımının boyutu $\dim(H_1 \otimes H_2) = m \times n$ olur.

Tanım 1: $\{|b_i\rangle\}_{i \in \{1, \dots, n\}}$, H_1 Hilbert uzayının birim dik bazları ve $\{|c_j\rangle\}_{j \in \{1, \dots, m\}}$, H_2 Hilbert uzayının birim dik bazları olmak üzere (Nielsen ve Chuang, 2002);

$$\{|b_i\rangle \otimes |c_j\rangle\}_{i \in \{1, \dots, n\}, j \in \{1, \dots, m\}} \quad (2.20)$$

ifadesi $H_1 \otimes H_2$ birleşik Hilbert uzayının birim dik bazlarının kümesidir. Tensörel çarpım aşağıdaki aksiyomları sağlar:

1. $|\Psi_1\rangle \in H_1$ ve $|\Psi_2\rangle \in H_2$ olmak üzere, $|\Psi_1\rangle \otimes |\Psi_2\rangle \in H_1 \otimes H_2$ dir. Burada $|\Psi_1\rangle$ ve $|\Psi_2\rangle$ elemanı oldukları Hilbert Uzayında birer vektörü temsil etmektedir.

2. $c \in \mathbb{C}$, $|\Psi_1\rangle \in H_1$ ve $|\Psi_2\rangle \in H_2$ ise

$$c(|\Psi_1\rangle \otimes |\Psi_2\rangle) = (c|\Psi_1\rangle) \otimes |\Psi_2\rangle = |\Psi_1\rangle \otimes (c|\Psi_2\rangle) \quad (2.21)$$

dir.

3. $|\Psi_1\rangle, |\Phi_1\rangle \in H_1$ ve $|\Psi_2\rangle \in H_2$ için ,

$$(|\Psi_1\rangle + |\Phi_1\rangle) \otimes |\Psi_2\rangle = |\Psi_1\rangle \otimes |\Psi_2\rangle + |\Phi_1\rangle \otimes |\Psi_2\rangle \quad (2.22)$$

olur.

4. $|\Psi_1\rangle \in H_1$ ve $|\Psi_2\rangle, |\Phi_2\rangle \in H_2$ olmak üzere,

$$|\Psi_1\rangle \otimes (|\Psi_2\rangle + |\Phi_2\rangle) = |\Psi_1\rangle \otimes |\Psi_2\rangle + |\Psi_1\rangle \otimes |\Phi_2\rangle \quad (2.23)$$

olur.

5. A ve B sırasıyla H_1 ve H_2 uzayları üzerinde lineer operatörler olsun. $A \otimes B$ operatörü de $H_1 \otimes H_2$ üzerinde bir lineer operatör olup aşağıdaki gibi tanımlanır:

$$(A \otimes B)(|\Psi_1\rangle \otimes |\Psi_2\rangle) = A|\Psi_1\rangle \otimes B|\Psi_2\rangle \quad (2.24)$$

Tensörel çarpım, Dirac Notasyonu dışında matris ifadelerle (operatörler) de yapılır. A matrisi $m \times n$ boyutlu, B matrisi $p \times q$ boyutlu olarak verilsin. A ve B nin soldan

Kroneker (tensörel) çarpımı $np \times mq$ boyutlu bir matristir ve aşağıdaki şekilde ifade edilir (Desurvire, 2009):

$$A \otimes B = \begin{bmatrix} A_{11}B_{11} & \cdots & A_{1n}B_{1q} \\ \vdots & & \vdots \\ A_{m1}B_{q1} & \cdots & A_{mn}B_{pq} \end{bmatrix} \quad (2.25)$$

2.3.5 Dolaşıklık (Entanglement)

Dolaşıklık, klasik fizikte açıklanamayan kuantum mekaniğince açıklanabilen birleşik sistemlerin bir özelliğidir. Özdeş olarak türetilen iki veya daha fazla sayıdaki atom altı parçacık arasında var olan bir kolerasyondur.

Tanım 1: $|\Psi_1\rangle, |\Psi_2\rangle, \dots, |\Psi_n\rangle$ kuantum durumlarının sırasıyla H_1, H_2, \dots, H_n Hilbert Uzayları üzerinde tanımlandığını kabul edelim. $|\Psi_1\rangle, |\Psi_2\rangle, \dots, |\Psi_n\rangle$ durumlarını global olarak temsil eden $|\Psi\rangle$ ifadesi, $|\Psi\rangle \in H = \otimes_{j=1}^n |H_j\rangle$ olmak üzere aşağıdaki eşitliğin sağlanmaması halinde dolaşıktır (Lomonaco ve Brandt, 2002; McMahon, 2007):

$$|\Psi\rangle = \otimes_{j=1}^n |\Psi_j\rangle \quad (2.26)$$

Dolaşık n alt durumdan oluşan kuantum bir sistem bileşenlerine ayrılamaz. Dolaşıklık, yalnızca fiziksel kaynakların lineer sınırlarında, üstel seviyedeki büyük süperpozisyon ifadelerini elde etmemizi sağlar.

Tanım 2: $\rho(j, k)$, H_j Hilbert Uzayı üzerinde bir yoğunluk operatörü olsun. $H = \otimes_{j=1}^n H_j$ olmak üzere, ρ aşağıdaki eşitliğin sağlanmaması durumunda dolaşıktır:

$$\rho = \sum_{k=1}^l \lambda_k \left(\otimes_{j=1}^n \rho(j, k) \right) \quad (2.27)$$

Burada $\sum_{k=1}^l \lambda_k = 1$ ifadesinde l pozitif sayma sayısı ve λ_k pozitif gerçel sayıdır.

Aşağıda gösterilen Bell durumları dolaşık olan durumlardır (Desurvire, 2009).

$$\begin{aligned} |\beta_{00}\rangle &= \frac{|00\rangle + |11\rangle}{\sqrt{2}} & |\beta_{01}\rangle &= \frac{|01\rangle + |10\rangle}{\sqrt{2}} \\ |\beta_{10}\rangle &= \frac{|00\rangle - |11\rangle}{\sqrt{2}} & |\beta_{11}\rangle &= \frac{|01\rangle - |10\rangle}{\sqrt{2}} \end{aligned} \quad (2.28)$$

2.3.6 Matrisler ve operatörler

Operatörler, uygulanan fonksiyonu başka bir fonksiyona çeviren matematiksel işlemlerdir. Türev ve integral işlemleri uygulandıkları fonksiyonu yeni bir fonksiyona dönüştürdükleri için birer operatör olarak kabul edilir. Operatörler vektörel büyüklükler için de genelleştirilebilir. Bir \hat{A} operatörü, $|\Psi\rangle$ durumuna uygulandığında yeni bir durum olan $|\Phi\rangle$ 'nin elde edilmesini sağlar.

$$|\Phi\rangle = \hat{A}|\Psi\rangle \quad (2.29)$$

Kuantum mekaniğindeki dönüşümler lineer olduğundan, Dirac notasyonlarıyla ifade ettiğimiz durumlara etki eden operatörlerin lineer dönüşüm yaptığını kabul ediyoruz.

Bazı operatör tanımları:

Tanım 1: Pauli operatörleri: İki boyutlu uzayda ‘ket’ ile gösterilen vektörlere dolayısıyla kubitlere etki edebilen, kuantum hesaplama teorisinin temel operatörleridir. Pauli Operatörleri aşağıdaki sembollerden biriyle gösterilebilir (Nielsen ve Chuang, 2002):

$$\sigma_0, \sigma_1, \sigma_2, \sigma_3 \equiv \sigma_0, \sigma_x, \sigma_y, \sigma_z \equiv I, X, Y, Z \quad (2.30)$$

Pauli Operatörlerinin işlevleri ve matris gösterimleri aşağıdaki gibidir:

$$\sigma_0|0\rangle = |0\rangle, \quad \sigma_0|1\rangle = |1\rangle \quad I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2.31)$$

$$\sigma_1|0\rangle = |1\rangle, \quad \sigma_1|1\rangle = |0\rangle, \quad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (2.32)$$

$$\sigma_2|0\rangle = -i|1\rangle, \quad \sigma_2|1\rangle = i|0\rangle \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \quad (2.33)$$

$$\sigma_3|0\rangle = |0\rangle, \quad \sigma_3|1\rangle = -|0\rangle \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (2.34)$$

Tanım 2: Transpozisinin kompleks eşleniğine eşit olan operatör hermitik operatördür (Nielsen ve Chuang, 2002).

$$\hat{A} = \hat{A}^\dagger \quad (2.35)$$

Hermitik operatörler aşağıdaki özellikleri sağlarlar:

$$(\alpha \hat{A})^\dagger = \alpha^* \hat{A}^\dagger \quad (2.36)$$

$$(|\Psi\rangle)^\dagger = \langle\Psi| \quad (2.37)$$

$$(\langle\Psi|)^\dagger = |\Psi\rangle \quad (2.38)$$

$$(\hat{A}\hat{B})^\dagger = \hat{B}^\dagger\hat{A}^\dagger \quad (2.39)$$

$$(\hat{A}|\Psi\rangle)^\dagger = \langle\Psi|\hat{A}^\dagger \quad (2.40)$$

$$(\hat{A}\hat{B}|\Psi\rangle)^\dagger = \langle\Psi|\hat{B}^\dagger\hat{A}^\dagger \quad (2.41)$$

$$(\hat{A} + \hat{B} + \hat{C})^\dagger = \hat{A}^\dagger + \hat{B}^\dagger + \hat{C}^\dagger \quad (2.42)$$

Tanım 3: \hat{A}^{-1} ifadesi, \hat{A} operatörünün tersi olmak üzere (2.43) denklemindeki I ifadesi birim operatördür.

$$\hat{A}^{-1}\hat{A} = I \quad (2.43)$$

Birim operatörler genellikle U ile gösterilirler ve aşağıdaki şartı da sağlarlar;

$$UU^\dagger = U^\dagger U = I \quad (2.44)$$

Tanım 4: Aşağıdaki eşitliği sağlayan operatör normal operatördür;

$$\hat{A}\hat{A}^\dagger = \hat{A}^\dagger\hat{A} \quad (2.45)$$

Pauli operatörleri hem hermitik hem de birim operatörlerdir (McMahon, 2007).

2.3.7 Öz değerler ve öz vektörler

Doğrusal dönüşümlerle modellenebilen birçok problemin öz vektörlerle basit çözümleri mümkündür. Lineer cebirde, bir kare matrisin öz vektörü (karakteristik vektörü) yapılan

lineer dönüşümle yönünü değiştirmez. Kuantum mekaniğinde, gözlenebilirler öz değerleri reel olan hermitik operatörlere karşılık gelirler. Bir hermitik operatörün öz vektörleri ortogonal olan farklı öz değerlere karşılık gelirler. Kuantum mekaniğindeki ölçme işleminde de elde edilmesi olası olan durumlar hermitik operatörlerin öz değerleriyle ifade edilir.

Sonlu boyutlu bir A gözlenebilirin öz değer bileşenleriyle ifadesi aşağıdaki gibidir;

$$A = \sum_{j=0}^{d-1} \lambda_j |\Phi_j\rangle\langle\Phi_j| \quad (2.46)$$

Burada öz değerler $\lambda_0, \dots, \lambda_{d-1} \in \mathbb{R}$ kümesinden ve öz vektörler $\{|\Phi_j\rangle\}$ Hilbert Uzayının H ortonormal bazlarından seçilebilir.

Tanım: $A: V \rightarrow V$ bir lineer operatör olmak üzere, sıfırdan farklı bir $|v\rangle \in V$ vektörü varsa $\lambda \in \mathbb{C}$, A nın bir özdeğeridir.

$$A|v\rangle = \lambda|v\rangle \quad (2.47)$$

Burada $|v\rangle$, λ özdeğeriyle A operatörünün öz vektörü olarak tanımlanır. A operatörünün öz değerleri

$$\det|A - \lambda I| = 0 \quad (2.48)$$

Karakteristik denkleminin çözümü ile bulunur. Bu denklemde λ bilinmeyen değer ve I birim matristir (McMahon, 2007; Nielsen ve Chuang, 2002).

2.3.8 Spektral (tayfsal) ayrıştırma

Spektral teoremi, hermitik matrislerin öz değerlerinin reel olduğunu ve aynı zamanda kompleks ortonormal tabanlar üzerinde köşegenleştirilebileceğini açıklar.

Tanım: n boyutlu bir V hermitik uzayı verilmiş olsun, $f: V \rightarrow V$ de verilen bütün özeşlenik lineer dönüşümler için, f ' nin özvektörlerinin (v_1, \dots, v_n) gibi bir ortonormal tabanı vardır. Bir matris için bu temel bazlar, matrisin köşegen elemanlarıdır (McMahon, 2007).

$$\begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & \lambda_n \end{bmatrix} \quad (2.49)$$

Burada $\lambda_i \in \mathbb{R}$ ve öz değerdir.

Spektral(tayfsal) ayrıştırımayı sağlayan bir A operatörü bazı $|v_i\rangle$ bazları için

$$A = \sum_{i=1}^n \lambda_i |v_i\rangle\langle v_i| \quad (2.50)$$

şeklinde ifade edilebilir.

2.3.9 Bir operatörün izi

Bir A operatörünün N boyutlu bir Hilbert uzayı üzerindeki izi, $\{|\Psi_i\rangle\}$ baz vektörlerinin herhangi bir kümesi için,

$$tr(A) = \sum_{i=1}^N \langle \Psi_i | A | \Psi_i \rangle \quad (2.51)$$

olarak ifade edilir. Eğer $|\Psi_i\rangle$, A operatörünün özvektörleri olarak seçilirse, operatörün izi, bütün özdeğerlerinin toplamına eşit olur. Bir operatörün izi, kuantum mekaniğindeki ölçme işlemi için oldukça önemlidir.

Özel olarak eğer bir A operatörü matris formunda gösterilmiş ise, matrisin köşegen elemanlarının toplamı operatörün izine eşittir.

2.3.10 Bir operatörün beklenen değeri

Projeksiyon operatörü: Projeksiyon operatörleri kuantum mekaniğinin ölçme işleminden sonra sistemin durumunu inceleyen önemli bir yöntemi temsil ederler. Normalize edilmiş bir kuantum durumun dış çarpım özelliği kullanılarak elde edilir.

$$\hat{P} = |\Psi\rangle\langle\Psi| \quad (2.52)$$

Tanım 4: Bir \hat{P} operatörü

$$1. \hat{P} = \hat{P}^\dagger \quad (2.53)$$

$$2. \hat{P} = \hat{P}^2 \quad (2.54)$$

koşullarını sağlaması durumunda projeksiyon operatörü olarak tanımlanır (Nielsen ve Chuang, 2002).

Teorem: Bir projeksiyon operatörü sadece '0' ve '1' değerlerini alabilir.

İspat: λ 'nın herhangi bir $|\lambda\rangle$ öz vektörü için;

$$\lambda|\lambda\rangle = \hat{P}|\lambda\rangle = \hat{P}^2|\lambda\rangle = \lambda^2|\lambda\rangle \quad (2.55)$$

denklemini verilebilir. Bu eşitlikten λ için, $\lambda = 0$ veya $\lambda = 1$ değerlerinden birinin elde edileceği hemen görülür.

Teorem: Bir kuantum mekanik sistemin durumu, \hat{A} genel gözlenirin ölçümünden sonra dejenere olan olası a_i özdeğerleri ile ifade edilir:

$$\hat{P}_i|\Psi\rangle. \quad (2.56)$$

Burada \hat{P}_i , a_i öz değerleri ile \hat{A} nın bütün öz vektörlerini kapsayan H Hilbert uzayının alt uzayı üzerinde bir projeksiyon operatörüdür (McMahon, 2007).

$$\hat{P}_i = \sum_i |\Psi_i\rangle\langle\Psi_i| \quad (2.57)$$

2.3.11 Operatörlerin fonksiyonları

Bir kuantum mekanik sistemin hamiltoniyen operatörü yazıldığında, operatör fonksiyonları sıkça kullanılır. Fonksiyonların bir operatör üzerinde tanımlanmasının iki yolu vardır. Birincisi, tamlık özelliğine sahip bir özvektör kümesiyle tanımlanan operatörlerle çok iyi çalışırlar.

Tanım: a_i özdeğerleriyle bir \hat{A} operatörü ve tamlık özelliği taşıyan bir $|a_i\rangle$ özvektörlerin kümesi verilmiş olsun. $f: \mathbb{C} \rightarrow \mathbb{C}$ kompleks sayılardan kompleks sayılara tanımlanan bir fonksiyon yazılabilir (McMahon, 2007):

$$f(\hat{A}) = \sum_{i=1}^N f(a_i)|a_i\rangle\langle a_i| \quad (2.58)$$

İkinci olarak, bir operatörün fonksiyonu Taylor serisi gibi bir kuvvet serisi olarak açılabilir.

Tanım: $f: \mathbb{C} \rightarrow \mathbb{C}$ şeklinde verilen bir fonksiyon,

$$f(x) = \sum_{i=0}^{\infty} f_i x^i \quad (2.59)$$

kuvvet serisi olarak açılımını göstermektedir. Bu denklem bir operatör için,

$$f(\hat{A}) = \sum_{i=0}^{\infty} f_i \hat{A}^i \quad (2.60)$$

olarak yazılır.

Teorem: Herhangi bir \hat{U} üniteri operatör için,

$$\hat{U} = e^{i\hat{H}} \quad (2.61)$$

Denklemini sağlayan bir \hat{H} hermitik operatörü vardır (McMahon, 2007; Nielsen ve Chuang, 2002).

2.3.12 Kübit ve kuantum bellek

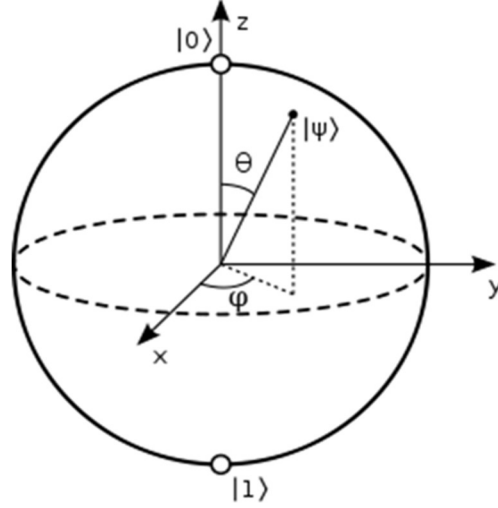
Klasik hesaplama için temel işlem biriminin ‘bit’ olmasına benzer şekilde, kuantum hesaplama teorisi için de ‘kübit’ (kuantum bit) temel işlem birimidir. Bir bit sadece ‘0’ veya ‘1’ değerini alır fakat ara bir değer almaz. Bir kübit ise klasik hesaplamadaki ‘0’ ve ‘1’ değerlerine karşılık olarak $|0\rangle$, $|1\rangle$ veya bu iki temel bazın bir lineer kombinasyonunu (superpozisyon) değer olarak alabilir.

Kuantum mekaniğinde, herhangi bir kuantum sistem her zaman birim ket ($|\cdot\rangle$) vektör normu ile temsil edilir. Dolayısıyla bir kuantum sistem olarak kübit kavramı, bu esası sağlayacak şekilde tanımlanmalıdır.

Tanım 1: $\alpha, \beta \in \mathbb{C}$ ve $|0\rangle, |1\rangle$ iki boyutlu Hilbert uzayında temel durum bazları olmak üzere, $|\alpha|^2 + |\beta|^2 = 1$ normalizasyon şartını sağlayan

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (2.62)$$

ifadesi bir kübiti temsil eder. Bir kübit, Bloch Küresi üzerinde geometrik bir gösterime sahiptir. Bloch küresi bir z karmaşık sayınının $|z| = 1$ şartını sağlaması durumunda, birim küreye dönüşür. Böylece bir kübit küre yüzeyi üzerindeki bir nokta gibi temsil edilebilir.



Şekil 1. Bloch Küresi.

Bloch küresi üzerideki bir noktanın kartezyen koordinatları denklem (2.63)'deki gibi ifade edilebilir:

$$(x, y, z) = (\sin \theta \cos \varphi, \sin \theta \sin \varphi, \cos \theta) \quad (2.63)$$

Bir kübitin formal ifadesi olan (2.62) denklemi;

$$|\Psi\rangle = e^{i\gamma} \left(\cos \frac{\theta}{2} |0\rangle + e^{i\varphi} \sin \frac{\theta}{2} |1\rangle \right) \quad (2.64)$$

olarak da gösterilebilir. θ , φ ve γ ifadeleri birer gerçel sayıdırlar. $e^{i\gamma}$ çarpanı gözlenebilir bir etkiye sahip olmadığı için ihmal edilebilir. Böylece (2.64) denklemi;

$$|\Psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\varphi} \sin \frac{\theta}{2} |1\rangle \quad (2.65)$$

olur (Nielsen ve Chuang, 2002).

2.3.12.1 Çoklu kübit gösterimi

N adet kübitten oluşan bir birleşik sistem, $n = 2^N$ olmak üzere

$|0 \dots 00\rangle, |0 \dots 01\rangle, |0 \dots 10\rangle, \dots, |1 \dots 11\rangle$ temel bazlarını gereken N tane 2-boyutlu

$H_n = H_2 \otimes \dots \otimes H_2$ Hilbert uzayının tensörel çarpımı ile ifade edilir.

$|x_1 \dots x_n\rangle, i = \{0,1\}$ Temel hesaplama bazları ile çoklu kubitlerden oluşan sistem (2.66) denkleminle gösterilebilir:

$$|\Psi\rangle = \sum_{i=0}^{n-1} c_i |i\rangle \quad (2.66)$$

Ölçme işlemi sonucunda c_i karmaşık sayılarının mutlak değerlerinin karesi $|x_1 \dots x_n\rangle$ temel baz değerlerinin ortaya çıkma olasılığını verir.

$$P(|i\rangle) = |c_i|^2 \quad (2.67)$$

İki kubitte oluşan bir sistemin genel durumu aşağıdaki gibi gösterilir (ya da iki kubitte bellekte tutulma ifadesi):

$$|\Psi\rangle = c_{00}|00\rangle + c_{01}|01\rangle + c_{10}|10\rangle + c_{11}|11\rangle \quad (2.68)$$

Burada tek bir kubitte olduğu gibi, kompleks sayılarla ifade edilen genlikler ilgili durumun ortaya çıkma olasılığını $|c_i|^2$ verir (Nielsen ve Chuang, 2002).

$$c_i, x \in (00, 01, 10, 11) \equiv \{0,1\}^2 \quad (2.69)$$

2.3.12.2 Kubitlerin bellekte (register) saklanması

Klasik hesaplamada (Vonn Neumann bilgisayar mimarisinde) bilgi ‘bit’ dizileri şeklinde saklanır. N elemana sahip bir bit dizisi 2^N adet konfigürasyona sahip olmasına rağmen, sadece bir karakteri ifade eder.

Kuantum hesaplama teorisinin öngördüğü bellekte de benzer şekilde bilgi ‘kubit’ olarak saklanır. Klasik hesaplamadan farklı olarak, N tane kubitte oluşan bir kubit dizisi, sadece bir karakteri değil 2^N kadar durumu (karakter) süperpozisyon ilkesinden dolayı aynı anda saklayabilir.

Kuantum bellek N tane kubitte dizilişi olarak düşünülebilir. İkili olarak oluşan bir sayı dizisi,

$$n = \sum_{k=0}^{N-1} n_k 2^k, \quad n_k = 0 \text{ veya } 1 \quad (2.70)$$

olarak kuantum bellekte saklanabilir ve bu diziliş (2.71) denkleminde olduğu gibi, kubitlerin tensörel çarpımı olarak gösterilebilir.

$$|n\rangle = |n_{N-1}\rangle |n_{N-2}\rangle \dots |n_1\rangle |n_0\rangle \quad (2.71)$$

Kuantum bellekten bağımsız olarak N tane kubitin dolaşık (entangled) durumu genel olarak,

$$|\Psi\rangle = \sum_{n=0}^{2^n-1} \alpha_n |n\rangle \quad (2.72)$$

denkleminle ifade edilir, burada α_n kompleks sayıları temsil eder (Lomonaco ve Brandt, 2002).

2.3.13 Kuantum kapılar

Klasik hesaplamada mantık devrelerini oluşturan ve tersinir olmayan kapılar (NOT kapısı hariç) ('0' ve '1') bitlerini yönlendirirler. Kuantum kapılar da kubitlerin $|0\rangle$ ve $|1\rangle$ durumlarını yönlendiren operatördürler. Kuantum hesaplamanın üniteri olma özelliğinden dolayı, kuantum kapıların da tersinir olması gerekir. Klasik hesaplama pratikte tersinir bir işleve sahip olmamasına rağmen, klasik hesaplama için de aşağıdaki gibi bir tersinir fonksiyon tanımlanabilir.

Tanım: Eğer bir $f: \{0,1\}^n \rightarrow \{0,1\}^n$ Boolean fonksiyonu $2^n \times 2^n$ ile verilen bir matris kombinasyonu olarak tanımlanırsa, bu fonksiyona tersinir fonksiyon denir (McMahon, 2007).

Herhangi bir $\delta: \{0,1\}^n \rightarrow \{0,1\}^m$ fonksiyonu için,

$$\delta'(z, 0) = (z, \delta(z)) \quad (2.73)$$

denkleminle verilen bir $g': \{0,1\}^{m+n} \rightarrow \{0,1\}^{m+n}$ tersinir fonksiyon mevcuttur.

Bu çalışmada klasik hesaplamada kullanılan kapıların (mantık devrelerinin) işlevsel özelliklerine değinilmeyecektir (Aaronson, S.; Grier, D. ve Schaeffer, L., 2015).

2.3.13.1 Tek kubit için kuantum kapıları

Daha önce belirtildiği gibi, bir kubit (α, β) kompleks genliklerine sahip, $\{|0\rangle, |1\rangle\}$ ortonormal temel hesaplama bazları üzerinden, 2 boyutlu Hilbert vektör uzayında tanımlanan kuantum hesaplama kuramının işlem birimidir. Kuantum kapılar aynı zamanda üniteri operatörler olduklarından, bir kübite uygulanan kuantum kapısı 2×2 boyutlu bir matris olmalıdır. Bu nedenle (2.30) de tanımlanan Pauli matrisleri ve (2.75) de verilen Hadamard matrisi bir kübite uygulanabilen kapılardır. *NOT* kapısı, *X* Pauli matrisine karşılık gelmektedir. ‘*A*’ kuantum kapısı (üniteri operatörü) $|\Psi\rangle$ kübitine uygulandığından;

$$|\Psi'\rangle = A|\Psi\rangle = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \alpha'|0\rangle + \beta'|1\rangle \quad (2.74)$$

sonucu elde edilir.

2.3.13.1.1 Hadamard kapısı

Hadamard kapısı

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{\sqrt{2}} (X + Z) \quad (2.75)$$

olarak tanımlanır ve (2.62) te tanımlanan her hangi bir kübite uygulandığında, $|+\rangle, |-\rangle$ saf kuantum durumlarını temsil etmek üzere kübiti,

$$|\Psi'\rangle = H|\Psi\rangle = \frac{1}{\sqrt{2}} (X|\Psi\rangle + Z|\Psi\rangle) = \alpha|+\rangle + \beta|-\rangle \quad (2.76)$$

süperpozisyonu durumuna dönüştürür (Desurvire, 2009).

2.3.13.2 Çoklu kubit sistemi için kapılar

Çoklu kübitlere etki eden kapılar, koşullu mantıksal işleyişi sağlayan ve kontrol bitlerine sahip olan kapılardır. Tıpkı tek kübite uygulanan kapılar gibi çoklu kubit kapıları da tersinirdir.

2.3.13.2.1 CNOT (XOR) kapısı

CNOT kapısı çoklu kubitlere etki eden bir kapı prototipi olup, kontrol ve veri kübiti olmak üzere iki giriş değerine sahiptir. Eğer control kübiti $|0\rangle$ ise hedef kubit üzerinde herhangi bir değişim olmaz, fakat control kübiti $|1\rangle$ ise hedef kübite *NOT* kapısı (*X* Pauli) kapısı uygulanır.

$|a\rangle$ ve $|b\rangle$ birer kubit olmak üzere (kontrol ve hedef kubitleri), *CNOT* kapısı dolaşık olan bu kubitlere aşağıdaki kurala göre uygulanır:

$$|a, b\rangle \rightarrow |a, b \oplus a\rangle \quad (2.77)$$

CNOT kapısının matris formu *I* ve *X* Pauli matrisleri cinsinden

$$CNOT = \begin{bmatrix} I & 0 \\ 0 & X \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (2.78)$$

olarak ifade edilir (Desurvire, 2009).

2.3.13.2.2 Kontrollü-U (Controlled-U) kapıları

Kontrollü-*U* kapıları kategorisinde *U*, herhangi bir kuantum kapısı için hermitik olması mecburi olmayan birimsel bir matrisle temsil edilmektedir. (*I*, *X*, *Y*, *Z*) Pauli kapıları, (*H*) Hadamard kapısı ve R_k faz kapısını temsil etmek üzere ($U = I, X, Y, Z, H, R_k$), genelleştirilmiş Kontrollü-*U* kapısı,

$$Controlled - U = \begin{bmatrix} I & 0 \\ 0 & U \end{bmatrix} \quad (2.79)$$

olarak ifade edilir. Örneğin, $U = 2^{2\pi i/2^k}$ için faz kapısı elde edilir.

Herhangi bir Kontrollü-*U* kapısı, $U = R_{\vec{n}}(\theta)$ ile karakterize edilen hedef kubit rotasyonuna karşılık gelir. Burada $R_{\vec{n}}(\theta)$ Bloch küresi üzerinde \vec{n} birimsel vektörün θ dönme açısıyla eksenler civarında dönüştüren rotasyon operatörüdür (Desurvire, 2009).

2.3.13.2.3 Toffoli veya CCNOT kapısı

Toffoli kapısı aynı zamanda controlled-controlled-NOT kapısına karşılık gelir. Klasik hesaplamadaki Toffoli kapısı ab ve c giriş değerlerini alarak, hesaplamada tersinirliği garantilemek için a, b control bitlerini muhafaza eden bir XOR kapısıdır. Burada a, b control bitleri ve c ise hedef bitidir. Toffoli kapısı için $c' = c \oplus ab$ olmak üzere çıkış değerleri a, b, c' dir.

Kuantum hesaplamadaki Toffoli kapısı, $|c_1\rangle, |c_2\rangle$ kontrol kütiti ve $|a\rangle$ hedef kütiti olmak üzere, $|c_1\rangle|c_2\rangle|a\rangle$ bazlarında tanımlanan 8×8 lik bir matrise karşılık gelmektedir.

$$A_{Toffoli} = \begin{bmatrix} I & 0 \\ 0 & X \end{bmatrix} \quad (2.80)$$

Burada I , 6×6 lık bir birim matris ve X , 2×2 lik kuantum NOT matrisidir. Toffoli (CCNOT) kapısı

$$A_{Toffoli} = \begin{cases} |a\rangle \rightarrow X|a\rangle & \text{eğer } |c_1\rangle = |c_2\rangle = |1\rangle \text{ ise} \\ |a\rangle \rightarrow |a\rangle & \text{eğer } |c_1\rangle = |c_2\rangle = |0\rangle \text{ ise} \end{cases} \quad (2.81)$$

kuralına göre işler (Desurvire, 2009).

2.3.13.2.4 Çapraz geçiş veya takas kapısı (crossover or swap gate)

Birleştirilmiş $CNOT$ kapılarında, control ve hedef kütitleri arasında değişim yaparak bir devre oluşturulabilir. Yani, ilk kapıya hedef olarak verilen kütit, sonraki kapı için control kütiti olarak kabul edilir. Takas (Swap) kapısının matris formu,

$$A_{SWAP} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.82)$$

dir. Giriş değeri olarak $\{|a\rangle, |b\rangle\}$ alındığında Takas kapısı,

$$(|b\rangle, |a\rangle) = A_{SWAP}(|a\rangle, |b\rangle) \quad (2.83)$$

2.3.13.2.5 Fredkin veya kontrollü takas kapısı (controlled-swap)

Üç kubitlik giriş değerine sahip olan Fredki kapısı, yalnız ve yalnız ilk kubitin $|1\rangle$ olması durumunda ikinci ve üçüncü kubitleri değiştirir. İlk kubitin $|0\rangle$ olması durumunda herhangi bir değişim olmaz. Fredkin kapısının matris formu,

$$A_{FREDKIN} = \begin{bmatrix} I & O \\ O & A_{SWAP} \end{bmatrix} \quad (2.84)$$

olarak gösterilir. Burada I , 4×4 birim matrisi temsil eder.

2.4 Klasik Makine Öğrenmesi ve Yapay Sinir Ağları

2.4.1 Tanım ve özellikler

Yapay sinir ağları (YSA), hesaplama teorileri açısından, insan beyninin düşünme yeteneklerini kullanarak matematiksel olarak modellenebilen problem çözme yöntemidir.

İlk YSA ağı modeli 1943 yılında nörolog olan Warren McCulloch ve matematikçi olan Walter Pitts tarafından geliştirilmiştir. Algoritmik davranmayan ve öğrenmeye dayalı olarak çalışabilen YSA modelleme, sınıflandırma ve tahminde bulunma uygulamalarının tamamında kullanılmaktadır. (Görüntü işleme, örüntü tanıma, doğal dil işleme, bilgisayarlı görü, optimizasyon, zaman serileri analizi vb.)

YSA 'nın genel özellikleri aşağıdaki gibi sıralanabilir:

- Nümerik girdilerden problemin genel özelliklerini belirlerler.
- Paralel işlem yapabilme özelliğine sahiptirler.
- Bilgileri bellek yerine ağına dağıtık olan yapılarında saklarlar.
- Ağ fonksiyonları lineer değildir.
- Eksik bilgi ile çalışabilmektedirler.
- Hatayı tolere etme yetenekleri mevcuttur.
- Genelleme yapabilirler.

- Örnekler üzerinden öğrenme yeteneğine sahiptirler (Demuth, Beale, ve Hagan, 2000; Fernandes de Mello ve Antonelli Ponti, 2018; Sağırođlu, 2003)

2.4.2 Temel işlem birimi nöron

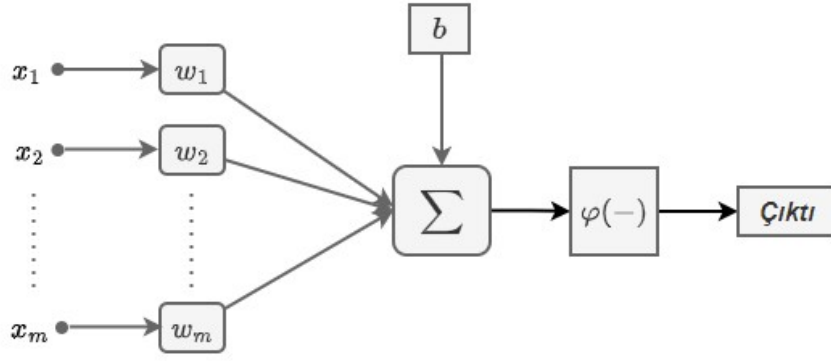
YSA da temel işlem birimi nörondur. Biyolojik olarak bir nöron, hücre gövdesi, sinyal alıcı olan dendritler ve sinyal iletici olan aksonlar olmak üzere 3 temel bileşenden meydana gelir. Biyolojik bir nöron ile YSA nın temel işlem birimi olan nöron arasındaki karşılaştırma tablo 2’de gösterilmiştir.

Tablo 2.Biyojik ve Yapay Nöron Bileşenlerinin Karşılaştırılması.

Biyojik Sinir Sistemi	Yapay Sinir Sistemi
Nöron	Temel İşlem Birimi
Dentrit	Toplama Fonksiyonu
Hücre Gövdesi (Cell Body)	Aktivasyon Fonksiyonu
Aksonlar	Nöron Çıkışı
Sinapslar	Ağırlıklar

Bir nöron için beş temel bileşen şekil 2’de gösterilmiştir.

- Giriş değerleri
- Ağırlıklar
- Toplama fonksiyonu
- Aktivasyon fonksiyonu
- Çıkış değeri



Şekil 2.YSA' daki bir Nöronun Temel Elemanları.

Bir nöron için n boyutlu giriş değerleri $\mathbf{x} = (x_1, \dots, x_n)$ ve n boyutlu ağırlık değerleri

$\mathbf{w} = (w_1, \dots, w_n)$ olmak üzere çıkış değeri (2.85) deki ifadedir. Giriş değerleri modellenen ağırlık çözmeye çalıştığı problemin türüne bağlı olarak skaler veya vektörel değerler olabilir (Fernandes de Mello ve Antonelli Ponti, 2018).

$$y = f\left(\sum_{i=1}^n w_i x_i + b\right) \quad (2.85)$$

Burada b nöron için uygun bir skaler olup, eşik veya 'bias' değeri olarak tanımlanır. f ise nöronun çıkış değerlerini istenilen değerler arasında sınırlayan aktivasyon (veya transfer) fonksiyonudur. YSA ağırları modellenirken yaygın olarak aşağıdaki aktivasyon fonksiyonları kullanılır.

- Lineer Fonksiyon,

$$f(z) = z \quad (2.86)$$

- Basamak (Threshold) Fonksiyonu,

$$f(z) = \begin{cases} 1, & \text{eğer } z \geq 0 \text{ ise} \\ 0 & \text{eğer } z < 0 \text{ ise} \end{cases} \quad (2.87)$$

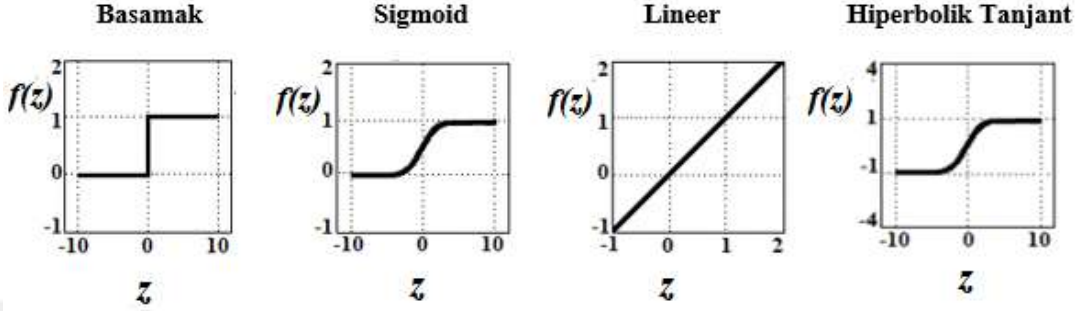
- Sigmoid Fonksiyonu,

$$f(z) = \frac{1}{1+e^{-z}} \quad (2.88)$$

- Hiperbolik Tanjant Fonksiyonu,

$$f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (2.89)$$

Tanımladığımız aktivasyon fonksiyonlarına ait grafik gösterimi şekil 3'te gösterilmiştir.



Şekil 3. Aktivasyon Fonksiyonları.

$$y = f\left(\sum_{i=1}^n w_i x_i + b\right) \quad (2.85)$$

2.4.3 YSA topolojileri

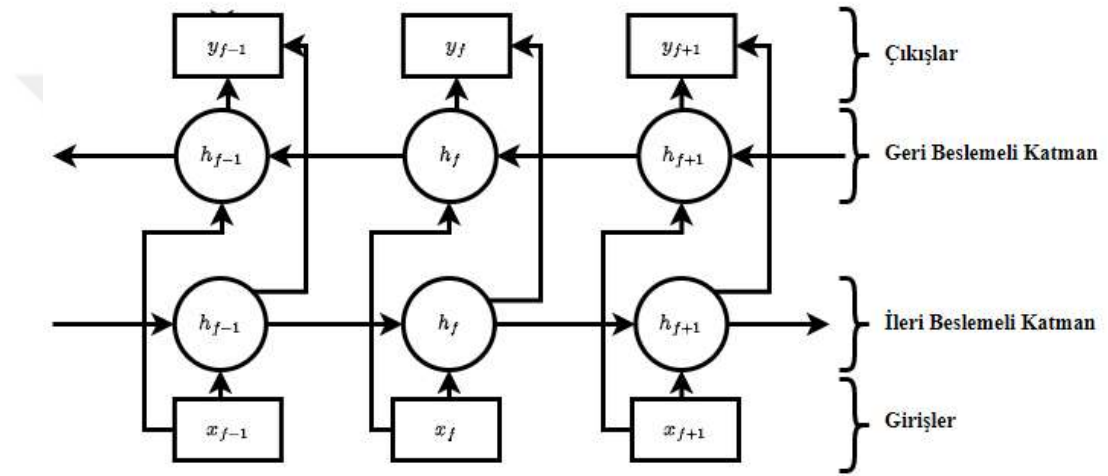
Yapay sinir ağları yapısal olarak benzer olmalarına karşın, standart bir ağ modelinden söz etmek mümkün değildir. Ağ tasarımcısının deneyim ve yetenekleri bu konuda avantajlı bir durum yaratabilir. Genel olarak ağ mimarileri açısından üç temel YSA türü mevcuttur. Bu ağ modelleri, ileri beslemeli (feed-forward) ağlar, geri beslemeli (recurrent) ağlar ve rekabetçi (competitive) ağlardır.

2.4.3.1 İleri beslemeli (feed-forward) YSA

İleri Beslemeli ağlarda (İBA) veri akışı, giriş katmanından çıkış katmanına kadar süren tek yönlü bir süreçtir. Yapısal olarak İBA lar tek ve çoklu katmanlardan oluşabilir. Giriş değerleri bir sonraki katmanda bulunan ilgili nörona iletilir. Çıkış değeri elde edilmeye kadar bu işlem devam eder. Bu YSA türünde ağ işleyişinde herhangi bir döngü oluşmaz. Adaline ve çok katmanlı perseptron ağları İBA ya örnek olarak verilebilir (Demuth, Beale, ve Hagan, 2000; Fernandes de Mello ve Antonelli Ponti, 2018; Sağiroğlu, 2003).

2.4.3.2 Geri beslemeli (recurrent) YSA

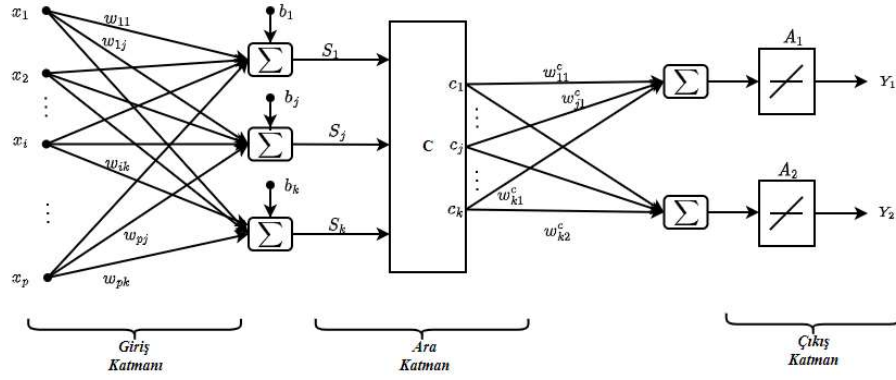
Geri beslemeli sinir ağı (GSA), ağı mimarisi açısından, yinelemeli (rekürsif) ayrık bir dinamik sistem yapısına sahiptir. Herhangi bir ara katmanda bulunan herhangi bir nöron, kendisinden önceki ve sonraki katmanlarda bulunan nöronlardan giriş değeri alabilir. Böylece ağı veriden beslenme şekli bir döngüsel yapı kazanır. Bu ağı yapısında dinamik hafıza mevcuttur. Özellikle tahmin problemlerinin çözümünde verimli sonuçların elde edilmesini sağlar. Hopfield, Elman ve Jordan ağı GSA için örnek ağılardır (Alpaydin, 2020; Demuth, Beale, ve Hagan, 2000; Sağiroğlu, 2003).



Şekil 4. Geri Beslemeli YSA.

2.4.3.3 Rekabetçi (Competitive) YSA

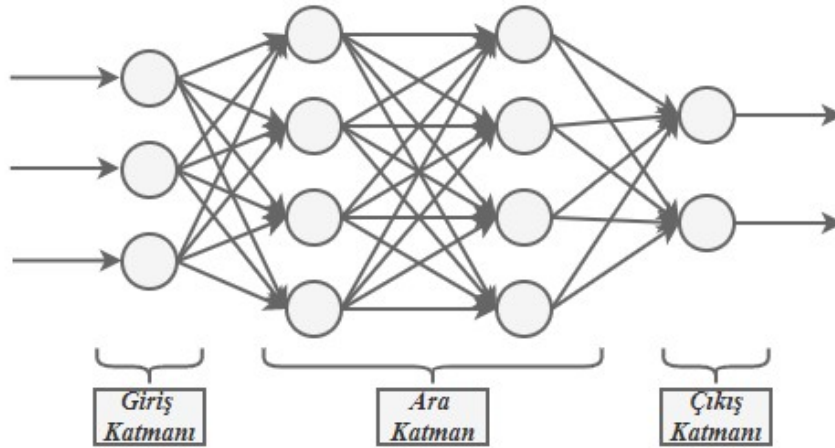
Bu YSA türü, ağı gönderilen giriş değerlerine ağıdaki tüm nöronların aynı anda ulaşmasını sağlayan rekabetçi bir topolojiye sahiptir. Belirlenmiş kriterler altında en iyi sonucu veren nöronun çıkış değeri istenen sonuç olarak kabul edilir. Sert rekabet koşullarında, ağı işleyişi açısından sadece kazanan nöronun ağı ağırlıkları güncelleştirilir. Daha esnek rekabet koşullarında ise komşu nöronlarında ağı ağırlıkları güncelleştirilir. LQV (Learning Vector Quantisation) ve SOM (Self Organizing Map) ağı modelleri rekabetçi YSA'na örneklerdir.



Şekil 5. Rekabetçi YSA.

2.4.4 Çok katmanlı-ileri beslemeli YSA

Çok katmanlı algılayıcılar (multi layer perceptrons) olarak ta bilinen bu ağ modeli, ilk katmanda bir giriş, bir veya daha fazlada sayıda ara ve çıkış katmanından oluşur. Her katmanda bir veya daha çok sayıda nöron bulunur. Aynı katmandaki nöronlar arasında bağlantı yoktur. Çıkış katmanındaki nöron sayısı tamamen problemin türüne bağlıdır. İleri doğru bir veri akışı yapısına sahip olduğu için ileri beslemeli ağlar kategorisindedir.



Şekil 6. Çok Katmanlı YSA Modeli.

2.4.5 YSA da öğrenme algoritmaları

YSA da hesaplama fonksiyonunu temelini oluşturan öğrenme işlemi, nöronlar arasındaki ağırlık değerlerinin bir matematiksel modellemeye dayanan herhangi bir kurala göre değiştirilmesi işlemidir. İlk olarak öğrenme algoritması Donald Hebb (1949) tarafından ileri sürülmüş olup akabinde delta, Kohonen ve Hopfield modelleri esas alınarak YSA

da öğrenme kuralları geliştirilmiştir. Genel olarak YSA da öğrenme işlemi üç temel yaklaşımla sınıflandırılmıştır. Bunlar, Danışmanlı Öğrenme (Supervised Learning), Danışmansız Öğrenme (Unsupervised Learning) ve Takviyeli Öğrenme (Reinforcement Learning) öğrenme algoritmalarıdır.

2.4.5.1 Danışmanlı öğrenme (supervised learning)

Bu öğrenme yaklaşımında YSA'ya ne öğrenmesi gerektiğini belirtecek bir danışmana (öğretmene) ihtiyaç vardır. Verilen giriş değerine karşılık elde edilmesi istenen çıkış değeri de ağa verilir. İstenen değer ile çıkış değeri arasındaki farka hata denir. Öğrenme süreci boyunca nöronlar arasındaki ağırlıklar güncelleştirilir, istenen hata tolerasyon değerine ulaşıldığında öğrenme işlemi bitmiş olur. Delta kuralı ve geriye yayma (backpropagation) algoritmaları danışmanlı öğrenmeye örnektir (Hu ve Hwang, 2002; Sağiroğlu, 2003).

2.4.5.2 Delta kuralı ile öğrenme

En küçük kareler yöntemi (Widrow-Hoff olarak ta bilinen) çok katmanlı ağlar için geliştirilen bir öğrenme kuralıdır. Ağın öğrenme kuralındaki temel amaç, ağa verilen örnek setinden elde edilen çıkış değeri ile istenen değer arasındaki farkı kabul edilebilir hata sınırına çekmektir.

Her bir k . girdi-çıkış örnek çifti için, j . girdi ile i . nöron arasındaki ağırlık değişimi (2.90) denklemi ile gösterilir:

$$\Delta w_{ij}^k = \alpha (d_i^k - y_i^k) x_j^k \quad (2.90)$$

Burada, d_i : istenen değer, y_i : çıkan değer, α : öğrenme katsayısı ve x_j : ağın j . girdisidir. Herhangi bir çok katmanlı sinir ağı için ortalama karesel hata (2.91) ve toplam karesel hata (2.92) ifadelerinde olduğu gibidir.

$$E_{ort} = \frac{1}{N} \sum_{i=1}^N (d_i - y_i)^2 \quad (2.91)$$

$$E_{top} = \sum_{i=1}^N (d_i - y_i)^2 \quad (2.92)$$

2.4.5.3 Geriye yayma (backprobagation) kuralı ile öğrenme

Geriye yayma algoritması çok katmanlı ağlarda delta kuralı için geliştirilmiş bir öğrenme modelidir. Hata fonksiyonu ağ ağırlıklarının türevlenebilir bir fonksiyonu ise, ağırlıkları değiştirmek için hatanın ağırlıklara göre türevi alınarak hata fonksiyonunu minimize edilmiş olur. Geri yayılım algoritması, hata fonksiyonunun türevlerinin değişimine dayanır.

Tek nöron için geriye yayma algoritması:

Bir nöron çıkış değeri, u net fonksiyonları hesaplayan bir toplama birimi ve f bir aktivasyon fonksiyonu olmak üzere (2.85) deki gibi ifade edilir.

$$y = f(u) \quad (2.93)$$

$\{(x(k), d(k)); 1 \leq k \leq K\}$ örnek kümesi ve bu örnek kümesine karşılık gelen çıkış kümesi $\{y(k); 1 \leq k \leq K\}$ ve ağırlık değerlerinin W temsil etmiş olsun. Hataların karesel toplamı (2.94) ifadesi ile elde edilir:

$$E = \sum_{k=1}^K [e(k)]^2 = \sum_{k=1}^K [d(k) - y(k)]^2 = \sum_{k=1}^K [d(k) - f(Wx(k))]^2 \quad (2.94)$$

Burada temel amaç, E hatasını en aza indirmek için ağırlık matrisi olan W 'yi ayarlamaktır. Bu yaklaşım doğrusal olmayan bir en küçük karesel optimizasyon problemine dönüşür. Bu problemin çözümü için birden çok doğrusal olmayan optimizasyon algoritması bulunmaktadır (Ruder, 2016). Temel olarak, bu algoritmalar (2.95) de gösterilen yinelemeli bir formülasyona benzemektedir:

$$w(t + 1) = w(t) + \Delta w(t) \quad (2.95)$$

Burada $w(t)$ mevcut ağırlık, $\Delta w(t + 1)$ güncellenen ağırlık değeridir.

Geriye yayma algoritmasını kullanarak, bir ağırlıkların güncellerken kullanılan eğim azaltma (gradient descend) metodu:

$$\frac{\partial E}{\partial w_i} = \sum_{k=1}^K \frac{\partial [e(k)]^2}{\partial w_i} = \sum_{k=1}^K 2[d(k) - y(k)] \cdot \left(-\frac{\partial y(k)}{\partial w_i}\right) \quad (2.96)$$

olarak verilir. Burada $i = 0,1,2, \dots$ değerlerini alır.

$$\frac{\partial y(k)}{\partial w_i} = \frac{\partial f(u)}{\partial u} \cdot \frac{\partial u}{\partial w_i} = f'(u) \frac{\partial}{\partial w_i} (\sum_{j=0} w_j x_j) = f'(u) x_i \quad (2.97)$$

olduğundan (2.98) denklemini aşağıdaki gibi elde edilir:

$$\frac{\partial E}{\partial w_i} = -2 \sum_{k=1}^K [d(k) - y(k)] f'(u(k)) x_i(k) \quad (2.98)$$

Aktivasyon fonksiyonunun türevi ile modüle edilen hata değeri (2.98) deki gibi ifade edilirse, (2.99) denkleminde (2.100) deki sonuç elde edilir (Hu ve Hwang, 2002).

$$\delta(k) = [d(k) - y(k)] f'(u(k)) \quad (2.99)$$

$$\frac{\partial E}{\partial w_i} = -2 \sum_{k=1}^K \delta(k) x_i(k) \quad (2.100)$$

Böylece ağ ağırlıklarının güncelleşmesi (2.101) deki gibi olur:

$$w_i(t+1) = w_i(t) + \eta \sum_{k=1}^K \delta(k) x_i(k) \quad (2.101)$$

Çok katmanlı ağlarda geriye yayma algoritması:

Tek nöron için yapılan geriye yayma algoritmasını, çok katmanlı nöronlara uygulamak için bazı farklı parametrelerin tanımlanması gerekir. Bu algoritmada net fonksiyon $u_j^{L-1}(k)$ ve buna karşılık gelen $(L-1)$. katmanını, j . nöronunu k . eğitim örneğinin çıkış ifadesi $y_j^{L-1}(k)$ ile gösterilmiştir. Ayrıca L . Katmanda bulunan i . nöronun sinaptik ağırlık ifadesi $w_{ij}^L(t)$ olarak tanımlanmıştır (Demuth, Beale, ve Hagan, 2000; Hu ve Hwang, 2002).

ÇKA' nın ağ ağırlık adaptasyon/güncelleme denklemi (2.102) ile elde edilir:

$$\begin{aligned} \frac{\partial E}{\partial w_{ij}^L} &= -2 \sum_{k=1}^K \frac{\partial E}{\partial u_i^L(k)} \cdot \frac{\partial u_i^L(k)}{\partial w_{ij}^L} \\ &= -2 \sum_{k=1}^K \left[\delta_i^L(k) \cdot \frac{\partial}{\partial w_{ij}^L} \sum_m w_{im}^L y_m^{L-1}(k) \right] \\ &= -2 \sum_{k=1}^K [\delta_i^L(k) \cdot y_m^{L-1}(k)] \end{aligned} \quad (2.102)$$

Buradaki $y_i^L(k)$ çıkış değeri, $(L + 1)$. katmanındaki M adet nöronun tamamından beslenir. $\delta_i^L(k)$ hata değeri aşağıdaki gibi hesaplanır:

$$\begin{aligned}\delta_i^L(k) &= \frac{\partial E}{\partial u_i^L(k)} = \sum_{m=1}^M \frac{\partial E}{\partial u_m^{L+1}(k)} \cdot \frac{\partial u_m^{L+1}(k)}{\partial u_i^L(k)} \\ &= \sum_{m=1}^M \left[\delta_m^{L+1}(k) \cdot \frac{\partial}{\partial u_i^L(k)} \sum_{j=1}^J w_{mj}^L(k) f(u_i^L(k)) \right] \\ &= f'(u_i^L(k)) \sum_{m=1}^M \delta_m^{L+1}(k) w_{mi}^L\end{aligned}\quad (2.103)$$

Çok katmanlı bir YSA da (2.103) denklemi ile çıkış katmanından giriş katmanına doğru bütün katmanlardaki hataları düzeltmeyi amaçlayan geriye yayma algoritmasının matematiksel modellemesidir.

2.4.6 Danışmansız öğrenme (Unsupervised Learning)

Danışmansız öğrenme genel olarak bir sınıflandırma veya doğal gruplandırma ağı olarak bilinir. Ağ sistemi, istenen doğru değerle ilgili herhangi bir bilgiye sahip olmadan, öğrenme metodunu giriş değerleri üzerinden gerçekleştirir. Benzer giriş değerlerine karşılık gelen çıkış değerleri aynı kümede toplanır. Kohonen tarafından geliştirilen öz örgütlemeli planlama (Self Organizing Map) ve Grossberg tarafından geliştirilen ART (Adaptive Resonance Theory) bu ağ modeli için bilinen önemli örneklerdir (Hu ve Hwang, 2002; Sağiroğlu, 2003).

2.4.7 Takviyeli öğrenme (Reinforcement Learning)

Takviyeli öğrenme, bulunduğu ortamı algılayarak otonom kararlar alabilen bir YSA sisteminin, hedefine varabilmek için doğru kararlara nasıl ulaşabileceğini öğrenebilen ağ modelidir.

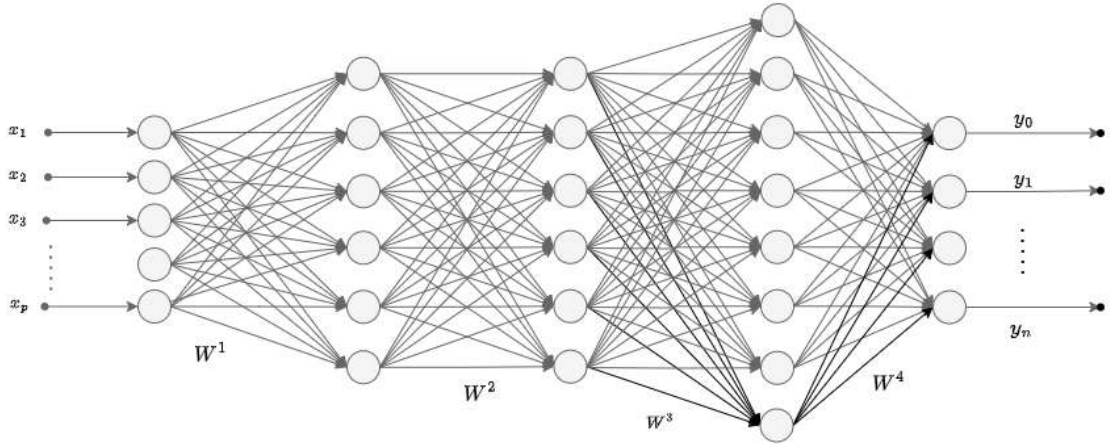
Takviyeli öğrenmede eğitmen, denetimli öğrenmedeki gibi etkin bir rol üstlenmediği için sisteme çok detay vermez veya veremez. Bu öğrenme yaklaşımında ağ sistemi bir sonuca ulaştığında, sonuç doğru ise ağ sistemi ödüllendirilir, yanlış ise cezalandırılır. Ağ sistemi öğrenme sürecini tamamladığında, olası giriş değerlerine karşılık olarak elde edilen

sonuçların doğruluğu sorgulanır ve böylece denenen tüm doğru ve yanlış durumlar ağ tarafından hatırlanmış olur.

2.4.8 Derin öğrenme (deep learning)

Derin öğrenme, çok katmanlı YSA modeli için geliştirilen bir makine öğrenmesi konseptidir. Aslında makine öğrenmesi alanında çalışan araştırmacılar, aynı anda fazla sayıda giriş değeri alabilen çok katmanlı (üç veya daha fazla sayıda) ağların daha iyi öğrenme yeteneklerine sahip olabileceğinin farkındaydılar. Fakat, data elde etme kısıtları ve düşük hızdaki işlem gücüne sahip makineler bu duruma engel teşkil etmekteydi. Günümüzde modern klasik bilgisayarların donanımsal olarak küçülüp işlem gücünün artması, derin öğrenme ağlarının birçok alanda verimli şekilde kullanılmasını sağlamıştır. Diğer makine öğrenme modelleriyle karşılaştırıldığında daha iyi sonuçların alındığı görülmüştür (Hinton, Osindero, ve Teh, 2006; Janiesch, Zschech, ve Heinrich, 2021; Schmidhuber, 2015).

Standart makine öğrenmesine göre daha fazla dataya ve işlem gücü daha yüksek olan makinelere ihtiyaç duymaktadır. Kuantum bilgisayarların donanımsal olarak geliştirilmesi, derin öğrenme algoritmasına sahip ağların yeteneklerini güçlendirecektir.



Şekil 7. Derin Öğrenme YSA Modeli.

2.5 Kuantum Makine Öğrenmesi

Kuantum hesaplamının süperpozisyon ve dolaşıklık gibi üstün özelliklerinin makine öğrenmesi modellerinde kullanabilme avantajlarını fark eden araştırmacılar tarafından çok fazla sayıda çalışma gerçekleştirilmiştir. Gelişen ve oldukça geniş bir araştırma

alanına sahip olan klasik makine öğrenmesi modellerinin kuantum makine öğrenmesi versiyonları, kuantum algoritmaların ve kuantum bilgisayar teknolojilerinin gelişmesiyle yeni bir makine öğrenmesi yaklaşımı olarak ortaya çıkmaktadır. Kuantum makine öğrenmesi modellerinin, klasik makine öğrenmesi modellerine göre öngörülen avantajları birçok araştırmacı tarafından ortaya konulmuştur (Biamonte, ve diğerleri, 2017; Gonçalves, 2016; Seow, Behrman, ve Steck, 2015; Schuld, Sinayskiy, ve Petruccione, 2015):

- Üstel seviyede bellek kapasitesi
- Az sayıdaki nöronla yüksek performansın elde edilmesi
- Hızlı öğrenme
- Hızlı işlem yeteneği
- Doğrusal problemlerin tek ağ katmanı ile çözülebilmesi
- Daha yüksek seviyelerde ağırlıklı kararlı davranışını sağlamak gibi üstünlüklerinden söz edilmektedir.

Kuantum makine öğrenmesi alanındaki çalışmalar kabaca bir kronolojik sıralamaya konulabilir:

- (Kak, 1995).
- (Perus, 1996).
- (Menneer ve Narayanan, 1995).
- (Zak ve Williams, 1998).
- (Ezhov ve Ventura, 2000).
- (Andrecut ve Ali, 2002).
- (Ricks ve Ventura, 2003).

- (de Oliveira, ve diğeri, 2008, Ekim).
- (Zhou, 2010).
- (Lloyd, Mohseni, ve Rebentrost, 2013).
- (Liu, Chen, Chang, ve Shih, 2013).
- (Schuld, Sinayskiy, & Petruccione, 2015).
- (Gonçaves, 2016).
- (Farhi ve Neven, 2018).
- (Tacchino, Macchiavello, Gerace, & Bajoni, 2019).
- (Biamonte, ve diğeri, 2017).
- (Arute, ve diğeri, 2019).
- (Broughton, ve diğeri, 2020)

Bu bölümde kuantum makine öğrenmesi ile ilgili kullanılan temel kuantum algoritmaları ve bazı kuantum makine öğrenmesi modelleri incelenecektir.

2.5.1 Kauntum temel bileşen analiz (TBA)

TBA, bir veri kümesindeki değişkenlerin boyutunu azaltmayı hedefleyen denetimsiz bir lineer dönüşüm işlemidir. Bu işlem, verilen bir veri kümesinde, aralarında bir korelasyona sahip N adet değişkenden daha az sayıda değişkeni, aralarında bir korelasyon bulunmayan, başlangıç değişkenlerinin lineer fonksiyonları olacak şekilde, yeni değişkenler olarak üreten istatistiksel bir yöntemdir. Bu dönüşüm sonrasında elde edilen ortogonal değişkenler temel bileşen olarak adlandırılır.

Matematiksel olarak TBA, veri kümesinde bulunan değişkenlerin kovaryans ve korelasyon matrislerinin özdeğer ve özvektörlerini bulma işlemidir. Bu matrisler varyansları ile özdeş, simetrik özelliğe sahip ve pozitif değerler alır (Abdi ve Williams, 2010; Jolliffe ve Cadima, 2016).

Kabul edelim ki bir data d boyutlu vektör uzayında ϑ_i olarak verilsin. T transpoz işlemini temsil etmek üzere, bu vektörlere ait kovaryans matrisi;

$$\Omega = \sum_i \vec{\vartheta}_i \vec{\vartheta}_i^T \quad (2.104)$$

şeklinde verilir. TBA, temel olarak,

$$\Omega = \sum_l e_l \vec{\Omega}_l \vec{\Omega}_l^T \quad (2.105)$$

kovaryans matrisini köşen matrise dönüştürmeyi hedefler. Bu denklemde Ω_l : Ω 'nın özvektörlerini, e_l : özvektörlere karşılık gelen özdeğerleri temsil eder. Ω_l değerlerinin en büyük birkaç tanesi alınır ve kalan diğer özvektörler ihmal edilir. Seçilen bu değerler kovaryans matrisinin temel bileşenleridir. Her bir bileşen dataların ortak özelliklerini ve korelasyon biçimini temsil eder.

TBA'nin kuantum karşılığı için aşağıdaki işlem sırası takip edilebilir (Abhijith, ve diğerleri, 2018; Biamonte ve Love, 2008; Lloyd, Mohseni, ve Rebentrost, 2013):

- $\vartheta_i \rightarrow |\vartheta_i\rangle$ vektörü kuantum yoğunluk matrisine kodlanır:

$$\rho = \frac{1}{N} \sum_i |\vartheta_i\rangle \langle \vartheta_i| \quad (2.106)$$

- ρ nun birden fazla kopyası hazırlanır.
- Elde edilen her kopyada üstel swap işlemi gerçekleştirilir.
- Özdeğerleri elde etmek için kuantum faz tahmin algoritması uygulanır.

2.5.2 Kuantum faz tahmin algoritması

Genel olarak uygulamalı matematikte ve özelde kuantum mekaniğinde birçok problem özvektörü verilen bir matrisin özdeğerini bulmayı gerektiren bir yönteme dayanır. ((2.3.7) 'de detaylandırıldığı gibi) kuantum hesaplamada U üniteri bir operatör olmak üzere, (2.107) deki eşitliği sağlayana $|\Omega\rangle$ özvektörünün, λ özdeğerini bulmayı amaçlar.

$$U|\Omega\rangle = \lambda|\Omega\rangle \quad (2.107)$$

U üniteri bir operatör olduğundan, $0 \leq \varphi < 1$ aralığında alınarak aşağıdaki denklem yazılabilir:

$$\lambda = e^{2\pi i \varphi} = \cos(2\pi \varphi) + i \sin(2\pi \varphi) \quad (2.108)$$

Böylece kuantum faz tahmin algoritması $\lambda = e^{2\pi i \varphi}$ özdeğeri üzerinden istenilen kesinlikte bir φ fazını bulmaya indirgenmiş olur. Algoritma için işlem adımları aşağıdaki verilmiştir:

- Başlangıç için n tane $|0\rangle$ kontrol kütibi ve $|\Omega\rangle$ özvektörünü kodlayan m tane kütib seçilir. Sistemin başlangıç durumu;

$$|\xi_0\rangle = |0\rangle^{\otimes n} |\Omega\rangle \quad (2.109)$$

olarak alınır.

- Başlangıç durumu üzerinde süperpozisyon durumunu elde etmek için, n tane Hadamard kapısı kontrol kütiblerine uygulanır.

$$|\xi_1\rangle = H^{\otimes n} |\xi_0\rangle = \frac{1}{\sqrt{2^n}} (|0\rangle + |1\rangle)^{\otimes n} |\Omega\rangle \quad (2.110)$$

- Bu aşamada C_U üniteri kontrol kapısı, yalnızca kontrol kütibinin $|1\rangle$ olması koşulunda $|\Omega\rangle$ durumu üzerinde U operatörüne uygulanır. Burada

$$U^{2^j} |\Omega\rangle = U^{2^{j-1}} U |\Omega\rangle = U^{2^{j-1}} e^{2\pi i \varphi} |\Omega\rangle = e^{2\pi i 2^j \varphi} |\Omega\rangle \quad (2.111)$$

ve $0 \leq j < n - 1$ olmak üzere n tane $(C_U)^{2^j}$ üniter operatörü $|\Omega\rangle$ öz durumlarına uygulanır.

$$\begin{aligned} |\xi_2\rangle &= \frac{1}{\sqrt{2^n}} \sum_{l=1}^{2^n-1} e^{2\pi i l \varphi} |l\rangle |\Omega\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{l=1}^{2^n-1} e^{2\pi i l 2^n \varphi / 2^n} |l\rangle |\Omega\rangle \end{aligned} \quad (2.112)$$

- Dördüncü adımda ters kuantum Fourier dönüşümü uygulanır (KFD^{-1}).

$|\xi_2\rangle$ durumunda bulunan kontrol kubitleri Fourier dönüşüm formundadırlar. (KFD^{-1}) 'nin uygulanması faz içeren durumun tüm genliklerini topladığı yeni bir duruma dönüştürür.

$$\begin{aligned} |\xi_3\rangle &= \frac{1}{2^n} \sum_{j=0}^{2^n-1} \sum_{l=0}^{2^n-1} e^{2\pi il\varphi} e^{-2\pi ilj/2^n} |j\rangle |\Omega\rangle \\ &= \frac{1}{2^n} \sum_{j=0}^{2^n-1} \sum_{l=0}^{2^n-1} e^{2\pi i l \frac{(2^n\varphi-j)}{2^n}} |j\rangle |\Omega\rangle \end{aligned} \quad (2.113)$$

- Faz Yaklaşımı:

φ parametresi $0 \leq \varphi < 1$ aralığında sürekli değer alabilen bir değişken ve $|j\rangle$ sadece ayrıık değişkenleri kodlayabildiğinden, α , $2^n\varphi$ 'ye yakın bir tamsayı ve δ , $0 \leq |2^n\delta| \leq \frac{1}{2}$ aralığında tanımlanan bir hata yuvarlama değişkeni olmak üzere $2^n\varphi = \alpha + 2^n\delta$ yaklaşımı ile aşağıdaki denklem yazılabilir:

$$|\xi_3\rangle = \frac{1}{2^n} \sum_{j=0}^{2^n-1} \sum_{l=0}^{2^n-1} e^{2\pi i l \frac{(\alpha-j)}{2^n} + 2\pi il\delta} |j\rangle |\Omega\rangle \quad (2.114)$$

- Ölçme:

Kontrol kubitleri ölçüldüğünde, her zaman $|\alpha\rangle = |2^n\varphi\rangle$ durumunu verir. Buradan faz değerini okunduktan sonra bir özdeğer hesaplanabilir. Ölçümden sonra tüm sistemin durumu $|2^n\varphi\rangle |\Omega\rangle$ olur. $\delta = 0$ olduğunda $P(|\alpha\rangle) = 1$, $\delta \neq 0$ durumunda $P(|\alpha\rangle)$ olasılığı ise aşağıdaki gibi olur (Hales ve Hallgren, 2000, Kasım; Kitaev, Shen, Vyalıy, ve Vyalıy, 2002; Koczyk, 2018).

$$\begin{aligned} P(|\alpha\rangle) &= \left| \frac{1}{2^n} \langle \alpha | \sum_{j=0}^{2^n-1} \sum_{l=0}^{2^n-1} e^{2\pi i l \frac{(\alpha-j)}{2^n} + 2\pi il\delta} |j\rangle \right|^2 \\ &= \left| \frac{1}{2^n} \sum_{l=0}^{2^n-1} e^{2\pi il\delta} \right|^2 \\ &= \frac{1}{2^{2n}} \left| \frac{1-e^{2\pi i 2^n\delta}}{1-e^{2\pi i\delta}} \right|^2 \end{aligned} \quad (2.115)$$

2.5.3 Takas (swap) testi

Takas testi, $|0\rangle$ durumunda verilen kontrol kubitinin ölçüm olasılığına bağlı olarak $\langle\pi|\mu\rangle$ gibi iki durumun benzerliğini ölçer. Algoritma aşağıdaki verilen adımlarla gerçekleşir:

- $|\pi\rangle$ ve $|\mu\rangle$ kuantum durumları ve kontrol kübiti olarak $|0\rangle$ seçilerek $|\phi_0\rangle = |0, \pi, \mu\rangle$ başlangıç durumu elde edilir.
- Kontrol kubitine Hadamard kapısı uygulanarak süperpozisyon durumu elde edilir:

$$|\phi_1\rangle = (H \otimes I^{\otimes n} \otimes I^{\otimes n})|\phi_0\rangle = \frac{1}{\sqrt{2}}(|0, \pi, \mu\rangle + |1, \pi, \mu\rangle) \quad (2.116)$$

- Kontrol kubitinin $|1\rangle$ olması durumunda $|\pi\rangle$ ve $|\mu\rangle$ durumlarına *Takas kapısı* uygulanır:

$$|\phi_2\rangle = \frac{1}{\sqrt{2}}(|0, \pi, \mu\rangle + |1, \pi, \mu\rangle) \quad (2.117)$$

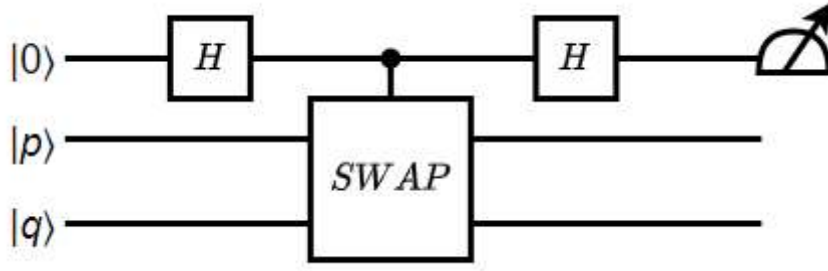
- Kontrol kubitine Hadamard kapısı tekrar uygulanır:

$$|\phi_3\rangle = \frac{1}{2}|0\rangle(|\pi, \mu\rangle + |\mu, \pi\rangle) + \frac{1}{2}|1\rangle(|\pi, \mu\rangle - |\mu, \pi\rangle) \quad (2.118)$$

- Son adımda kontrol kübiti ölçülür: Kontrol kubitinin $|0\rangle$ olma olasılığı aşağıdaki denklemlerle ifade edilir.

$$\begin{aligned} P(|0\rangle) &= \left| \frac{1}{2} \langle 0|0\rangle (|\pi, \mu\rangle + |\mu, \pi\rangle) + \frac{1}{2} \langle 0|1\rangle |1\rangle (|\pi, \mu\rangle - |\mu, \pi\rangle) \right|^2 \\ &= \frac{1}{4} |(|\pi, \mu\rangle + |\mu, \pi\rangle)|^2 \\ &= \frac{1}{4} (\langle\mu|\mu\rangle\langle\pi|\pi\rangle + \langle\mu|\pi\rangle\langle\pi|\mu\rangle + \langle\pi|\mu\rangle\langle\mu|\pi\rangle + \langle\pi|\pi\rangle\langle\mu|\mu\rangle) \\ &= \frac{1}{2} + \frac{1}{2} |\langle\pi|\mu\rangle|^2 \end{aligned} \quad (2.119)$$

Ölçüm sonucunda $|\pi\rangle$ ve $|\mu\rangle$ durumları $P(|0\rangle) = 0.5$ koşulunda ortogonal, $P(|0\rangle) = 1$ koşulunda ise aynı durumlar olarak yorumlanır (Aïmeur, Brassard, ve Gambs, 2006; Koczyk, 2018).



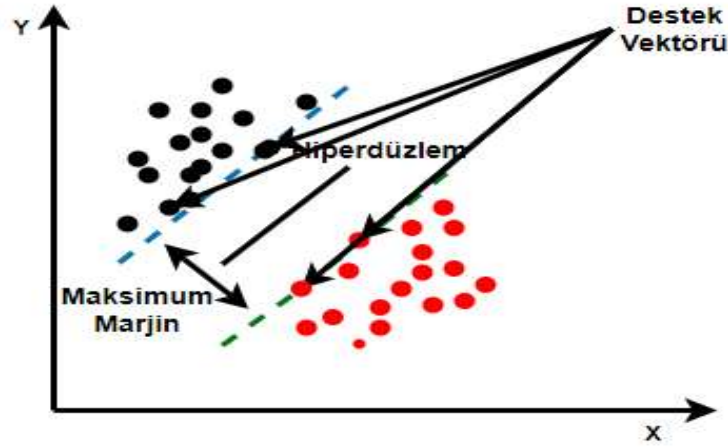
Şekil 8. Takas Testi Devresi.

2.5.4 Kuantum destek vektör makinesi (DVM)

DVM, klasik makine öğrenmesi kapsamında, etiketlenmiş veriler üzerinde dışbükey optimizasyon kuralına göre sınıflandırma yapabilen bir makine öğrenmesi algoritmasıdır. Bu algoritmada amaç, özellik vektörlerini iki sınıfa ayıran n-boyutlu uzayda optimal bir hiper düzlem tanımlamaktır. Bu hiper düzlem iki sınıf arasında bir ayırma düzlemi olup bu sınıfların özellik vektörleri arasında maksimum marja (mesafeye) sahip bir düzlemin bulunmasını sağlar. Maksimum marjı seçmenin amacı, gelecekteki özellik vektörlerini daha yüksek doğrulukla sınıflandırmak için bir boşluk oluşturmaktır. Girdi özellik vektörlerinin sayısı, hiper düzlemin boyutlarını belirler. Hiper düzlem, iki boyutlu özellik uzayı için bir doğru ve üç boyutlu bir özellik uzayı için iki boyutlu bir düzlemdir. Çoklu boyutlu düzlemlerde sınıflandırma yapmak için çekirdek fonksiyonları kullanılır. Düşük hesaplama kaynağı ile yüksek doğruluk derecelerini bulması nedeniyle denetimli makine öğrenmesi metodu olarak kullanılan en yaygın algoritmadır.

Kabul edelim ki, x_i data özellik uzayında d boyutlu bir girdi vektörü ve $y_i \in \{+1, -1\}$ çıktı değerlerini temsil etmek üzere, $\{(x_i, y_i) : i = 1, 2, \dots, M\}$ kümesinden M tane data seçilmiş olsun. Bu durumda DVM, dataları iki gruba bölen $\beta x + \beta_0 = 0$ hiper düzlemini bulur ve gelen dataları aşağıdaki gibi eşitsizliklere göre belirlenen sınırları kullanarak sınıflandırır:

$$\begin{aligned} \beta x_i + \beta_0 &\geq 1 \rightarrow y_i = +1 \\ \beta x_i + \beta_0 &\leq -1 \rightarrow y_i = -1 \end{aligned} \quad (2.120)$$



Şekil 9. Destek Vektör Makinesi.

Bir optimizasyonu yaklaşımı ile

$$y_i(\beta x_i + \beta_0) \geq 1 \quad (2.121)$$

denklemindeki $\frac{2}{|\beta|}$ değeri maksimum seçilerek iki sınıf arasındaki marj maksimuma çıkarılır. Böylece β ve β_0 parametrelerini tahmin ederek yeni bir x_i datası,

$$y(x_i) = \text{sgn}(\beta x_i + \beta_0) \quad (2.122)$$

denkleminde göre sınıflandırılır. Burada sgn – *signum* işaret fonksiyonudur (Cortes ve Vapnik, 1995; Smola ve Schölkopf, 2004).

Kuantum destek vektör makinesi (K-DVM) en küçük kareler yönteminin klasik DVM'e uygulamasının (Suykens ve Vandewalle, 1999) kuantum hesaplama metodu ile yeniden yorumlanmasıdır (Lloyd, Mohseni, ve Rebentrost, 2013). K-DVM nin amacı lineer (2.123) denklemini bir kuantum bilgisayar üzerinden çözerek sınıflandırmayı gerçekleştiren bir hiper düzlem bulmaktır.

$$\begin{bmatrix} 0 & \vec{1}^T \\ \vec{1} & K + \delta^{-1} \mathbf{1} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ \vec{y} \end{bmatrix} \quad (2.123)$$

Burada K , $M \times M$ boyutunda $K_{i,j} = x_i \cdot x_j$ olarak verilen bir çekirdek matrisini, δ ayarlama parametresini, $y = (y_1, \dots, y_m)^T$ ve $\alpha, \beta = \sum_{i=1}^M \alpha_i x_i$ olmak üzere normal vektörünü temsil eder.

K-DVM algoritmasının adımları aşağıdaki gibi sıralanabilir (Abhijith, ve diğerleri, 2018):

Girdi:

- $\{(x_i, y_i): i = 1, 2, \dots, M\}$ eğitim için veri kümesi
- x sorgulanan veri

Çıktı:

- $x \in \{+1 \text{ veya } -1\}$

Algoritma Prosedürü:

Birinci adım: (Lloyd, Mohseni, ve Rebentrost, 2013)'da verilen kuantum iç çarpım algoritmasını kullanarak $K_{i,j} = x_i \cdot x_j$ hesapla.

İkinci adım: (2.117) lineer denklemini çöz ve (Harrow, Hassidim, ve Lloyd, 2009)'de lineer denklemlerin çözümü için verilen kuantum algoritmasını kullanarak $|\beta_0, \alpha\rangle$ 'yı bul.

Üçüncü adım: HHL (Harrow, Hassidim, ve Lloyd, 2009) kuantum algoritmasını kullanarak $|\beta_0, \alpha\rangle$ eğitim sonuçlarına göre x verisinin sınıflandırmasını gerçekleştir (Giovannetti, Lloyd, ve Maccone, 2008; Rebentrost, Mohseni, ve Lloyd, 2014).

2.5.5 Kuantum K-means kümeleme

K-means kümeleme, verilen datayı istenilen sayıdaki (k) kümeye bölerek, aynı küme içinde maksimum benzerliği ve farklı kümeler arasında en az benzerliği amaçlayan denetimsiz bir öğrenme algoritmasıdır. Sınıflandırılan her bir data sadece bir kümeye ait olabilir. Bu algorithmada,

- Her biri bir kümenin merkezini veya ortalamasını temsil eden k adet data seçilir.

- Geriye kalan diğer veriler, önceden belirlenen kümelerin ortalama değerlerine olan uzaklıkları dikkate alınarak, en çok benzer oldukları kümelere dahil edilir.
- Yeni durumda kümelerin ortalama değeri hesaplanarak güncel küme merkezleri belirlenir ve tekrar data-merkez arası uzaklıklar incelenir. Küme merkezleri sabit oluncaya kadar bu işlem sürdürülür (Bradley ve Fayyad, 1998, Temmuz; Hartigan ve Wong, 1979; Sinaga ve Yang, 2020).

K-means algoritmasının kuantum uygulaması, yüksek boyutlu giriş vektörleri için üstel bir hızlanma sağlar. Bu yetenek N boyutlu klasik vektörün genlik kodlaması yardımıyla kuantum datasına dönüştürülürken sadece $\log_2 N$ adet kübit kullanarak gerçekleştirebilmesinden kaynaklıdır. K-means kümeleme için Takas testi, Mesafe ölçümü ve Grover Optimizasyonu gibi kuantum alt yordamları kullanır (Kopczyk, 2018; MacQuen, 1967; Zhang ve Ni, 2020).

Takas testi, $|0\rangle$ durumunda verilen kontrol kübitinin ölçüm olasılığına bağlı olarak $\langle\pi|\mu\rangle$ gibi iki durumun yakınlık durumunu ölçer. Bu yakınlık iki durum arasındaki benzerliğin göstergesidir. Eğer kontrol kübitinin $|0\rangle$ durumunda olma olasılığı $\frac{1}{2}$ ise $|\pi\rangle$ ve $|\mu\rangle$ durumlarının ortogonal, olasılığın 1 olması koşulunda ise $|\pi\rangle$ ve $|\mu\rangle$ nun aynı olduğu kabul edilir. Kontrol kübitinin $|0\rangle$ durumunda olma olasılığı denklem (2.124)'eki gibi ifade edilir.

$$P_{olasılık}(|0\rangle) = \frac{1}{2} + \frac{1}{2} |\langle\pi|\mu\rangle|^2 \quad (2.124)$$

Burada $|\pi\rangle$ ve $|\mu\rangle$ genlik kodlaması ile hazırlanmış n adet kübitten oluşur. Bu durumlar arasındaki Öklid uzaklığını $(|\pi - \mu|^2)$ hesaplamak için takas testi kullanılır ve bu işlem aşağıdaki adımlarla gerçekleştirilir:

1. Adımda; $Z = |\pi|^2 + |\mu|^2$ olmak üzere aşağıdaki gibi iki kuantum durumu hazırlanır.

$$|\Psi_1\rangle = \frac{1}{\sqrt{2}} (|0, \pi\rangle + |1, \mu\rangle) \quad (2.125)$$

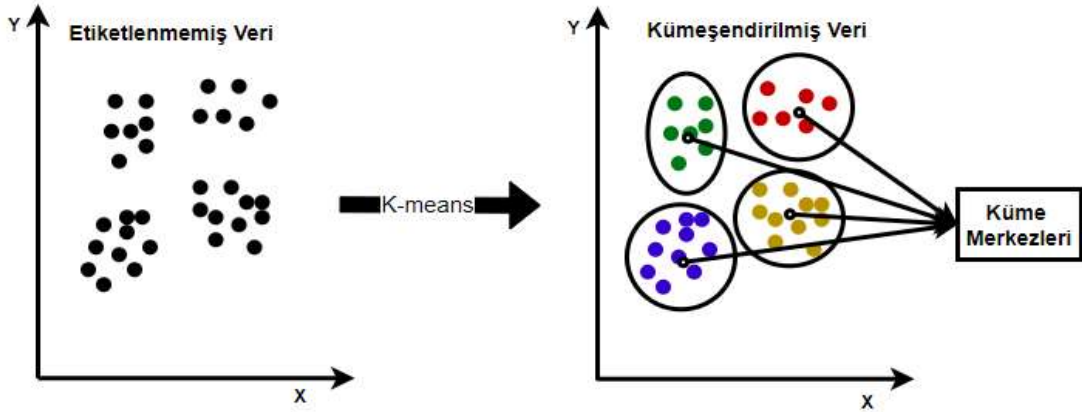
$$|\Psi_2\rangle = \frac{1}{\sqrt{2}}(|\pi\rangle|0\rangle + |\mu\rangle|1\rangle) \quad (2.126)$$

2. Adımda; Takas testi kullanılarak $\langle\Psi_1|\Psi_2\rangle$ iç çarpımı hesaplanır.

3. Adımda; Öklid mesafesi,

$$|\pi - \mu|^2 = 2Z|\langle\Psi_1|\Psi_2\rangle|^2 \quad (2.127)$$

denklemlerle hesaplanır. Bu yöntemle, her bir küme merkezine olan K mesafeleri hesaplanır. En yakın küme merkezini bulmak için Grover Algoritması üzerinden geliştirilen Grover Optimizasyonu alt programı kullanılır (Durr ve Hoyer, 1996; Grover, 1996, Temmuz; Ullah Khan, 2019).



Şekil 10. K-means Kümeleme.

2.5.6 Lineer cebir tabanlı kuantum makine öğrenmesi

Lineer cebir bir matematiksel yaklaşım olarak veri analizi ve mühendislik uygulamalarında kullanılır. Fakat kuantum mekaniği ve ona bağlı olarak geliştirilen kuantum hesaplama algoritmalarının tamamı yüksek boyutlu vektörleri temsil eden matrisler üzerinde çalışır. Kuantum makine öğrenmesi söz konusu olduğunda, herhangi bir kuantum vektör değişkeni ve bu vektörlere etki eden lineer operatörlerin matris (matematiksel) karşılığı lineer cebir kapsamındadır.

Örneğin, Bir kübitin 2 boyutlu bir sanal uzayda gösterilmesine benzer şekilde, n adet

kübit 2^n boyutlu bir kompleks Hilbert Uzayında temsil edilir. Doğal olarak bu kübitlere etki eden matris operatörlerinin de $2^n \times 2^n$ boyutunda olması gerekir (Biamonte, ve diğerleri, 2017).

2.5.6.1 HHL (Harrow, Hassidim ve Lloyd) algoritması

Lineer denklem sistemlerinin tersini bulmayı amaçlayan HHL (Harrow, Hassidim ve Lloyd) Algoritması çok sayıda kuantum makine öğrenmesi modelinin temelini oluşturur. Algoritmanın temel hedefi $Mx = \gamma$ denklemini bir kuantum bilgisayarda çözmektir (Biamonte, ve diğerleri, 2017).

$M|x\rangle = |\gamma\rangle$ formundaki ifade aşağıdaki dört adımda adımda çözülür:

1. γ vektörü $|\gamma\rangle = \sum_{i=1}^N \gamma_i |i\rangle$ gibi bir kuantum durum olarak kodlanır.
2. Faz tahmini için, $M = \sum_j \lambda_j u_j u_j^\dagger$ hermitik bir matris olmak üzere, $|\gamma\rangle$ 'ye e^{iMt} operatörü uygulanır:

$$e^{iMt} |\gamma\rangle = \sum_{j=1}^N \beta_j |u_j\rangle |\lambda_j\rangle \quad (2.128)$$

Böylece $|\gamma\rangle$ durumu denklem (2.129)'e dönüşür:

$$|\gamma\rangle = \sum_{j=1}^N \beta_j |u_j\rangle \quad (2.129)$$

Burada u_j M 'nin özvektörü, λ_j ise özdeğeridir. M 'nin hermitiyen olmaması durumunda (Harrow, Hassidim, ve Lloyd, 2009) de önerilen çözüm gerçekleştirilir. Önce hermitiyen bir C matrisi tanımlanır.

$$C = \begin{bmatrix} 0 & M \\ M^\dagger & 0 \end{bmatrix} \quad (2.130)$$

Daha sonra $C\vec{y} = \begin{bmatrix} \vec{\gamma} \\ 0 \end{bmatrix}$ denklemi çözülerek $\vec{y} = \begin{bmatrix} 0 \\ \vec{x} \end{bmatrix}$ sonucu elde edilir.

3. Yardımcı kubit ile $|\lambda_j\rangle$ kontrol kubitine rotasyon operatörü uygulanır ve aşağıdaki durum elde edilir:

$$\sum_{j=1}^N \left(\sqrt{1 - \frac{c^2}{\lambda_j^2}} |0\rangle + \frac{c}{\lambda_j} |1\rangle \right) \beta_j |\lambda_j\rangle |u_j\rangle \quad (2.131)$$

Bu denklemde C normalizasyon sabiti olup $|\lambda_j\rangle \rightarrow \frac{c}{\lambda_j} |\lambda_j\rangle$ dönüşümünü sağlar.

4. Son olarak ters faz tahmin adımı uygulanır. Bu $|\lambda_j\rangle$ yardımcı kubiti lineer bir dönüşümle $|0\rangle$ olarak alınır. Yardımcı kubit ölçümünün sonucu 1 olursa;

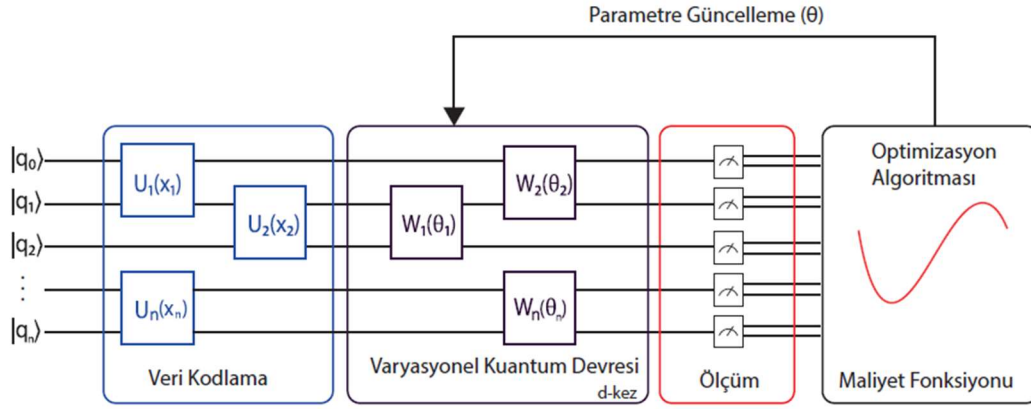
$$|x\rangle = \frac{1}{\sqrt{\sum_{j=1}^N c^2 |\beta_j|^2 / |\lambda_j|^2}} \sum_{j=1}^N \beta_j \frac{c}{\lambda_j} |u_j\rangle \quad (2.132)$$

olur. Elde edilen $|x\rangle$, $Mx = \gamma$ denklemindeki x 'in kuantum karşılığıdır (Zhang ve Ni, 2020).

3. ARAŞTIRMA YÖNTEMİ

3.1 Varyasyonel Kuantum Devresi

Yapay sinir ağlarının doğrusal olmayan matematiksel yapısını, doğrusal işleyen kuantum hesaplama mimarisiyle entegre etmek güçtür. Fakat, varyasyonel kuantum devresi ile geliştirilen algoritmalar, kuantum durumları hazırlayan ve sınırlı sayıdaki kubitlerden oluşan parametrize edilmiş kapılarla yönlendirilen bir kuantum devresini, klasik bir optimizasyon algoritması ile birleştirerek bu entegrasyon sorununu çözmektedir (Biamonte, ve diğerleri, 2017; Schuld, Sinayskiy, ve Petruccione, 2015). Şekil 11’de kuantum varyasyonel algoritmasının çalışma prensibi verilmiştir.



Şekil 11. Varyasyonel kuantum algoritmasının çalışma prensibi.

Varyasyonel kuantum devresi ile önerilen metot, ‘kuantum-klasik’ hibrit yaklaşımının uygulaması olarak, kuantum işlem biriminden (QPU-Quantum Processing Unit) elde edilen beklenen değer, klasik bir işlem (bilgisayar) birimi ünitesinde (CPU-classical processing Unit) optimize edilmesi, yapay sinir ağı (YSA) yapısına benzerdir (Benedetti, Lloyd, Sack, ve Fiorentini, 2019; Bergholm, ve diğerleri, 2018; Schuld, Bergholm, Gogolin, Izaac, ve Killoran, 2019).

Varyasyonel kuantum devresi ile geliştirilen kuantum algoritmaları parametre değiştirebilme özelliği sayesinde hata tolerasyon açısından verimlidirler. Bu avantaj varyasyonel kuantum devre yaklaşımını (near-term prosessor) yakın zamanda üretilen

kuantum işlemciler üzerinde uygulama gerçekleştirmek için ana adaylar haline getirmektedir (Biamonte, ve diğerleri, 2017; Schuld ve Petruccione, 2018). Varyasyonel devre algoritmasının çalışma prensibi aşağıda verilen adımlardan oluşmaktadır (Benedetti, Lloyd, Sack, ve Fiorentini, 2019; Mitarai, Negoro, Kitagawa, ve Fujii, 2018; Schuld ve Killoran, 2019):

1. $\{x_i\}$ girdi verileri kuantum durumuna kodlanır.

$$\{x_i\} \rightarrow |\Psi(x)\rangle \quad (3.1)$$

2. Hazırlanan kuantum durumuna parametre alabilen rotasyon operatörler (üniteri kuantum operatörler) uygulanarak yeni bir durum elde edilir.

$$|\Psi_{\text{çıkış}}(x_i; \theta)\rangle = U(\theta)|\Psi(x_i)\rangle \quad (3.2)$$

3. Varyasyonel kuantum devrelerinde, devrede üzerinde seçilen bazı gözlenirlerin beklenen değerini ölçmek için genellikle Pauli operatörleri gözlenir olarak kullanılır. Bir \hat{B} gözlenirinin devre üzerindeki beklenen değeri;

$$y(x_i; \theta) = \langle \Psi | U^\dagger(\theta) \hat{B} U(\theta) | \Psi \rangle \quad (3.3)$$

şeklindedir.

4. Klasik hesaplama yöntemleriyle, kuantum işlem biriminden elde edilen ‘beklenen değer’ bir maliyet fonksiyonuna girdi olarak verilerek klasik bir optimizasyon algoritmasıyla maliyet fonksiyonu minimize edilir. Bu işlem için belirlenen maliyet(cost) değeri, θ parametresi üzerinden, $y(x_i; \theta)$ çıktısı ve $f(x_i)$ istenen değerlere göre ifade edilir. Bu iterasyon istenen hata düzeyi yakalanıncaya kadar sürdürülür.

$$\text{Cost}(f(x_i), y(x_i; \theta)) \quad (3.4)$$

Bu çalışmada, bir regresyon problemi olan hava tahmininde, beş kübitlik kuantum varyasyonel devre modelinde süperpozisyon ve dolaşıklık etkileri incelenmiştir. Bu çerçevede, süperpozisyon, dolaşıklık ve rotasyon katmanlarının farklı yerlerde kullanılmasıyla ilgili gerçekleştirilen beş farklı deneyde ([1-8]) aralığında varyasyonel katmanlar kullanılmıştır. Kullanılan veri setinin özellikleri ve deneylerin detayları bu bölümde verilmiştir.

3.2 Veri Kümesi

Veri kümesi, İstanbul-Kireçburnu/Sarıyer bölgesine ait 2016-2020 yılları arasındaki saatlik hava durumu verilerini içermektedir. Veri kümesinde toplam yağış miktarı, buhar basıncı, nispi nem oranı, bulutluluk miktarı, aktüel basınç ve sıcaklık özellikleri ile birlikte toplam 43829 veri mevcuttur. Veri kümesinin özellikleri Tablo 3'te verilmiştir.

Tablo 3.Hava durumu veri kümesinin özellikleri

	Toplam yağış miktarı (kg/m^2)	Buhar basıncı (hPa)	Nispi nem oranı (%)	Bulutluluk miktarı (8 okta)	Aktüel basınç (hPa)	Sıcaklık (°C)
Ortalama	0,09	14,43	79,33	3,89	1008,8	15,41
Standart sapma	0,71	6,43	14,35	2,74	7,99	7,61
Varyans	0,51	41,36	205,86	7,58	63,83	57,84
Mod	0	9,3	99	7	1007,8	23,2
Ortanca	0	13	81	4	1008,3	15,4

3.3 Veri Önışleme

Öznitelik verileri (Toplam Yağış, Buhar Basıncı, Nispi Nem, Bulutluluk Miktarı, Aktüel Basınç) aşağıdaki formül kullanılarak standardize edildi:

$$\hat{x} = \frac{x - \mu}{\sigma} \quad (3.5)$$

Burada, x : öznitelik verisi, μ : özniteliğe ait ortalama, σ : özniteliğe ait standart sapma, \hat{x} : Standardize edilmiş öznitelik verisini göstermektedir.

3.4 Deneysel uygulamalar

Hava durumu tahmini için kullanılan beş kübitlik hibrit varyasyonel kuantum devresi beş aşamadan oluşmaktadır. İlk olarak tüm kübitler $|0\rangle$ başlangıç durumunda başlatılır. Ardından opsiyonel olarak tüm kübitler eşit genlikli süperpozisyon durumuna getirilir (Bu durum deney 3'te araştırılmıştır). Daha sonra hava durumu veri kümesinin beş özniteliğini $R_y(\alpha)$ döndürme kapılarından oluşan veri kodlama devresi ile R_y döndürme kapılarının α açılara gömerek özellik eşleme gerçekleştirilir. Ardından N tane

varyasyonel katman eklenir ve son olarak $\langle Z \rangle$ bazında beklenen değer için ölçüm işlemi gerçekleştirilir.

$|\psi\rangle$ ölçümden önceki kuantum durumu olarak kabul edilirse, beklenen değer aşağıdaki gibi hesaplanır. $\Psi(x) = \langle x|\Psi\rangle$ olmak üzere;

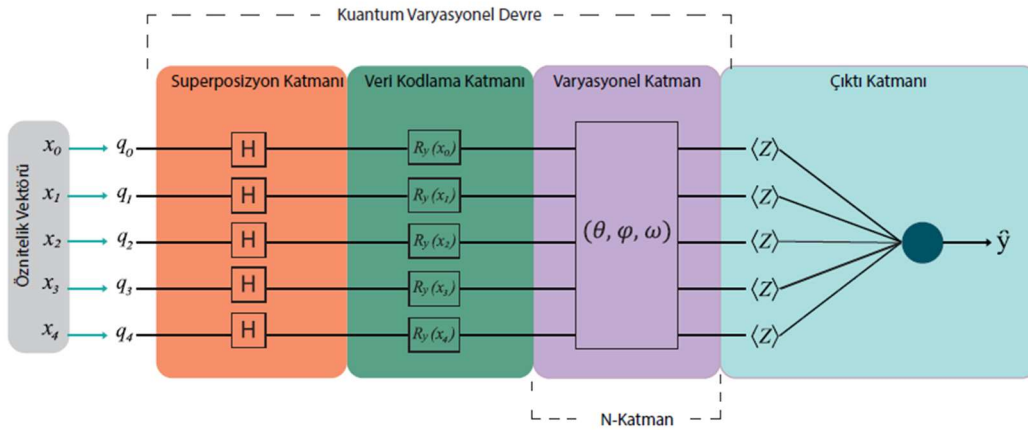
$$\langle Z \rangle = \langle \Psi|Z|\Psi \rangle = \int dx \Psi^*(x) Z\Psi(x) \quad (3.6)$$

olur. Z nin spektral ayrışması $Z|\Psi_i\rangle = z_i|\Psi_i\rangle$ olmak üzere, denklem (3.7) kullanılarak elde edilir.

$$Z = \sum_i z_i |\Psi_i\rangle \langle \Psi_i| \quad (3.7)$$

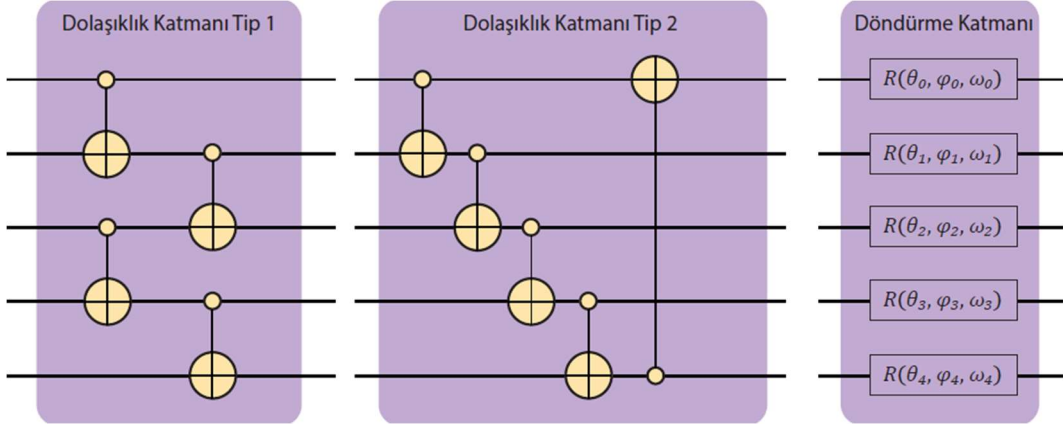
$$\langle Z \rangle = \langle \Psi|Z|\Psi \rangle = \sum_i z_i \langle \Psi|\Psi_i\rangle \langle \Psi_i|\Psi \rangle = \sum_i z_i |\langle \Psi_i|\Psi \rangle|^2 \quad (3.8)$$

Elde edilen ölçüm sonuçları çıktı katmanına beslenir. Çıkış katmanı bir nörondan oluşmasına rağmen başlangıç ağırlıkları normal dağılıma sahiptir. Ayrıca çalışmada kullanılan hibrit kuantum-klasik modellerde tüm parametreler geriye yayılım yöntemi ile eğitilmiş ve hata optimizasyonu için ortalama kare hata fonksiyonu kullanılmıştır. Daha sonra eğitim sonucunda elde edilen modellerle tahmin sonucu elde edilmiştir. Temel kuantum varyasyonel devre modeli Şekil 12'de verilmiştir.



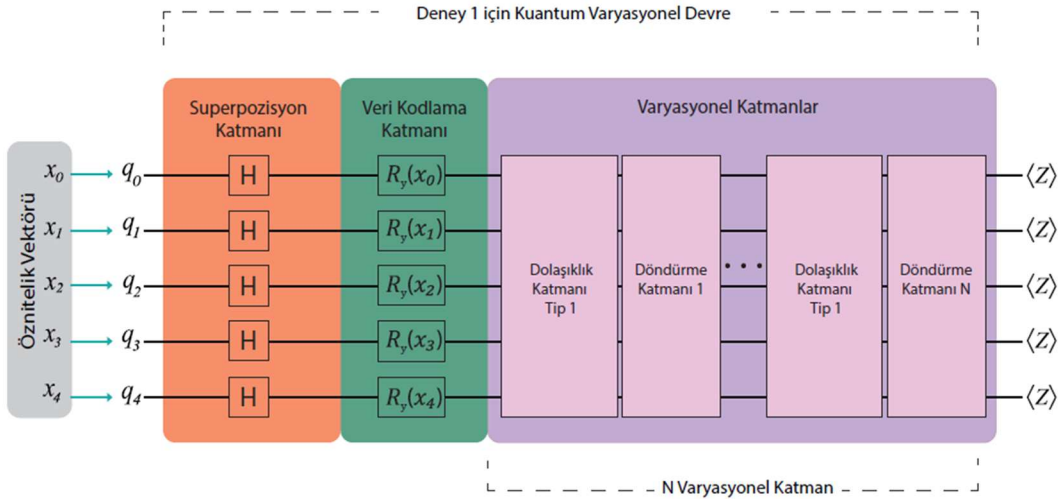
Şekil 12. Temel kuantum varyasyonel devre modeli şeması.

Değişken katmanda kullanılan devrelerin detaylı gösterimi Şekil 13'te verilmiştir.



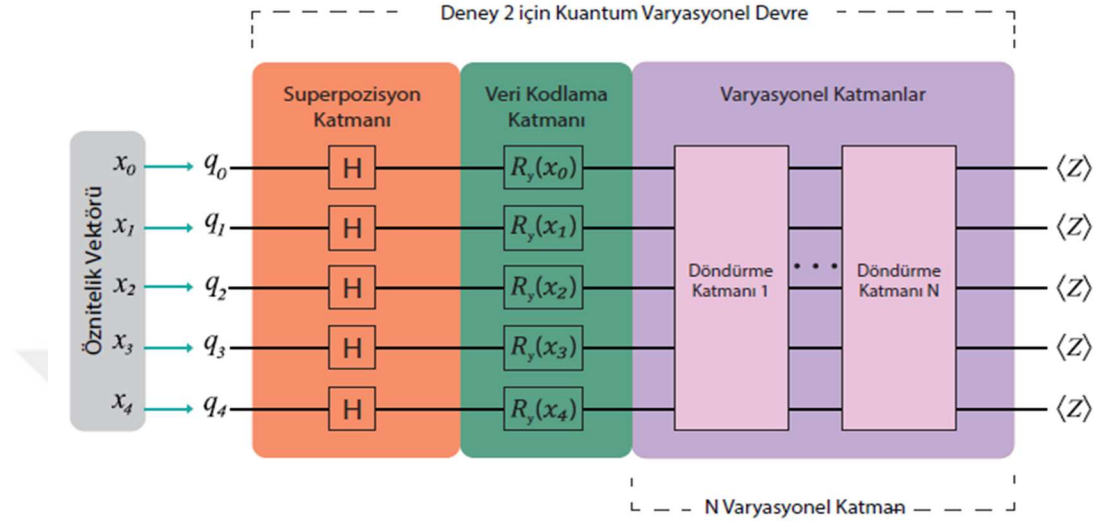
Şekil 13. Varyasyonel katmanda kullanılan farklı dolaşıklık ve rotasyon alt katmanlarının detaylı gösterimi.

Bu çalışmada gerçekleştirilen birinci deneyde, Şekil 14’te verilen kuantum varyasyonel devre mimarisindeki varyasyonel katmanında, Şekil 15’te verilen dolaşıklık katmanı tip 1 devresi ve rotasyon katmanı kullanılmıştır. Dolaşıklık katmanı tip 1 devresi, tüm kubitleri dolaşıklık transferi ile dolaylı olarak dolaşık hale getirmektedir. Rotasyon katmanındaki, rotasyon kapılarına ait parametreler, klasik bir optimizasyon algoritması tarafından eğitilebilir parametrelerdir. Bu deneyde kullanılan devre mimarisi Şekil 15’te verilmiştir.



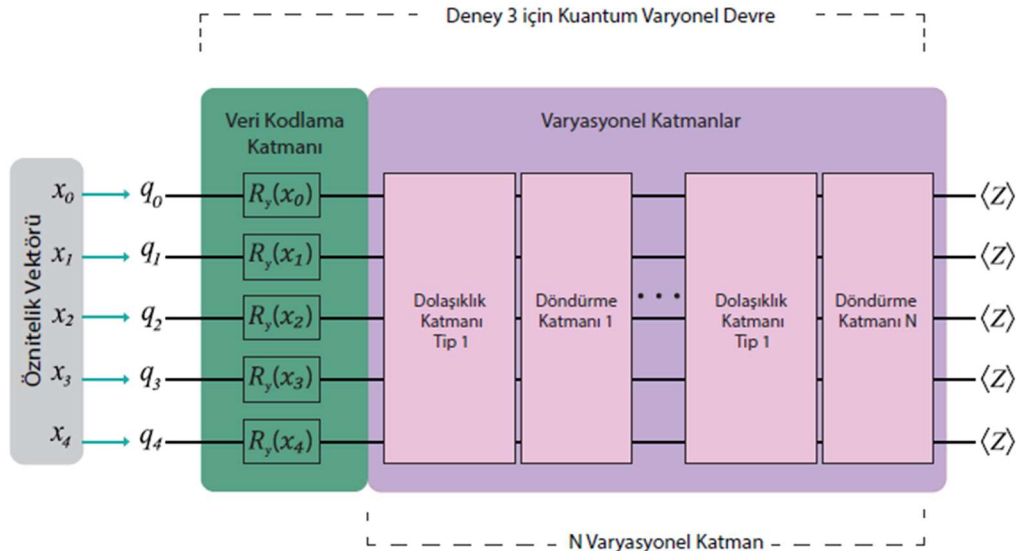
Şekil 14. Deney 1’de kullanılan kuantum varyasyonel devre mimarisi.

İkinci deneyde, birinci deneyde kullanılan varyasyonel katmandaki dolaşıklık alt katmanı kaldırılarak dolaşıklığın etkisi araştırıldı. Bu deneyde kullanılan devre mimarisi Şekil 15’ da verilmiştir.



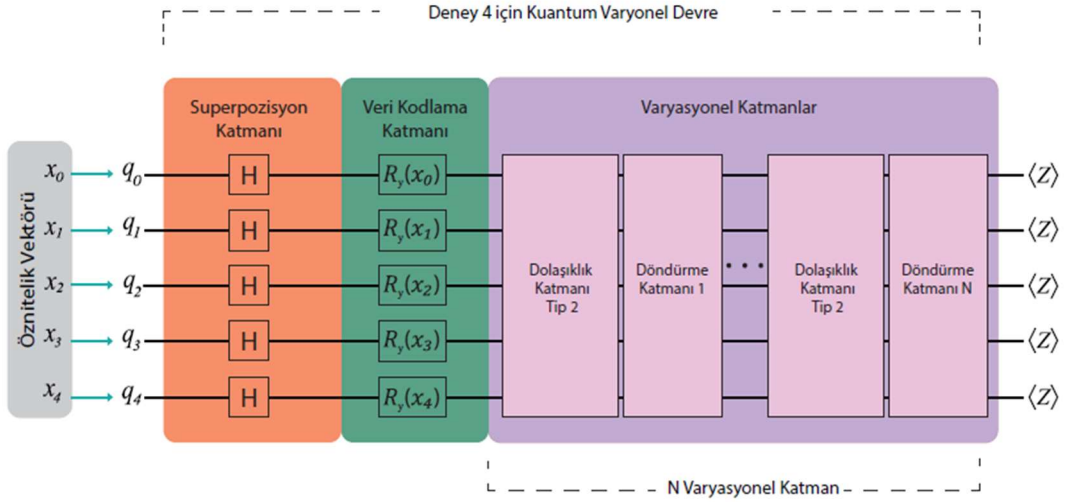
Şekil 15. Deney 2’de kullanılan kuantum varyasyonel devre mimarisi.

Üçüncü deneyde, birinci deneyde kullanılan süperpozisyon katmanını kaldırılarak, Hadamard kapısının veri kodlama katmanından önce kullanılmasının nasıl bir etki yaratacağı araştırıldı. Bu deneyde kullanılan devre mimarisi Şekil 16’ de verilmiştir.



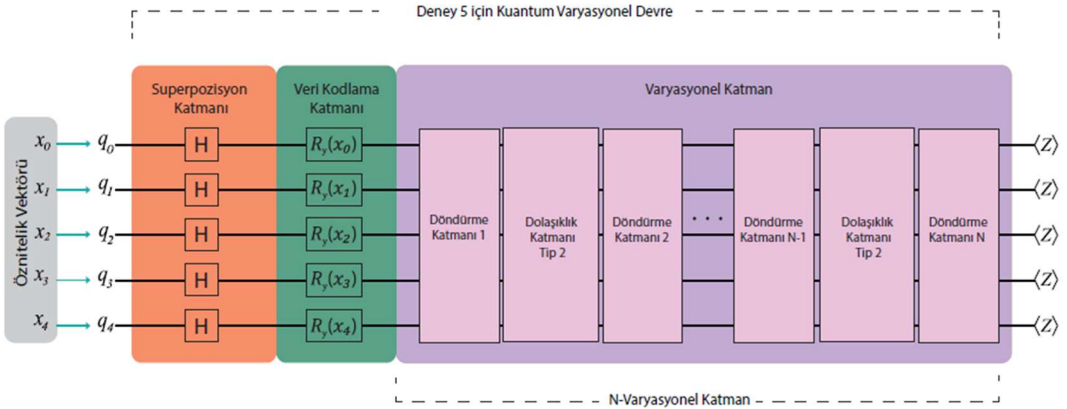
Şekil 16. Deney 3’de kullanılan kuantum varyasyonel devre mimarisi.

Dördüncü deneyde, birinci deneyde kullanılan varyasyonel katmandaki dolaşık katmanı tip 1 yerine, Şekil 13'te verilen dolaşıklık katmanı tip 2 devresini kullanıldı. Dolaşıklık katmanı tip 2 devresi, tüm kubitleri birbiriyle doğrudan dolaşık hale getirir. Bu deneyde iki farklı dolaşıklık alt katmanının birbirine göre etkisi araştırılmıştır. Kullanılan devre mimarisi Şekil 17'de verilmiştir.



Şekil 17. Deney 4’de kullanılan kuantum varyasyonel devre mimarisi.

Son deneyde ise Şekil 13'te verilen varyasyonel devre modelindeki varyasyonel katmanda, “rotasyon katmanı-dolaşıklık katmanı tip 2” katmanından oluşan bir varyasyonel alt katman dizisi tanımlandı. Bu deneyde farklı bir varyasyonel katman yapısının etkisi araştırılmıştır. Kullanılan devre mimarisi Şekil 18’de verilmiştir.



Şekil 18. Deney 5’de kullanılan kuantum varyasyonel devre mimarisi.

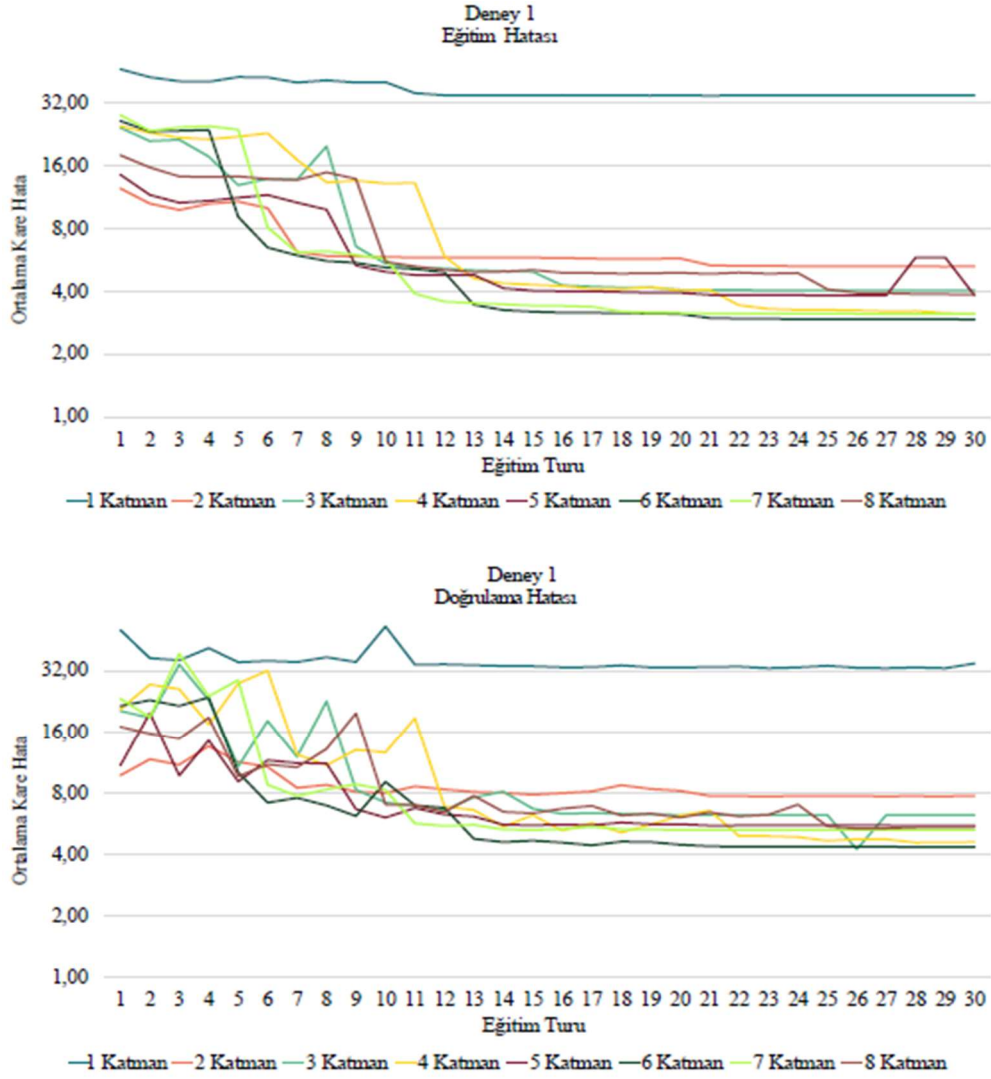
3.5 Deney Prosedürü

Bu çalışmada Pennylane kuantum hesaplama simülatörü (Bergholm, ve diğerleri, 2018) kullanıldı. Gerçekleştirilen tüm deneylerde veri kümesinin %9'u eğitim, %1'i doğrulama ve %90'ı test için ayrılmıştır. Klasik optimizasyon algoritması olarak Adam stokastik gradyent algoritması kullanılmıştır. Başlangıçta 0.1 gibi büyük bir öğrenme oranı ile başlayıp doğrulama hatasında 3 eğitim dönemi boyunca azalma gerçekleşmediğinden 0.1 çarpanıyla dinamik olarak azaltılıp, en küçük öğrenme oranı 0.00001 olarak ayarlanmıştır. Eğitim dönem sayısı (epoch) tüm deneyler için 30 alınmış ve hata fonksiyonu olarak, ortalama kare hata fonksiyonu kullanılmıştır.



4. BULGULAR ve TARTIŞMA

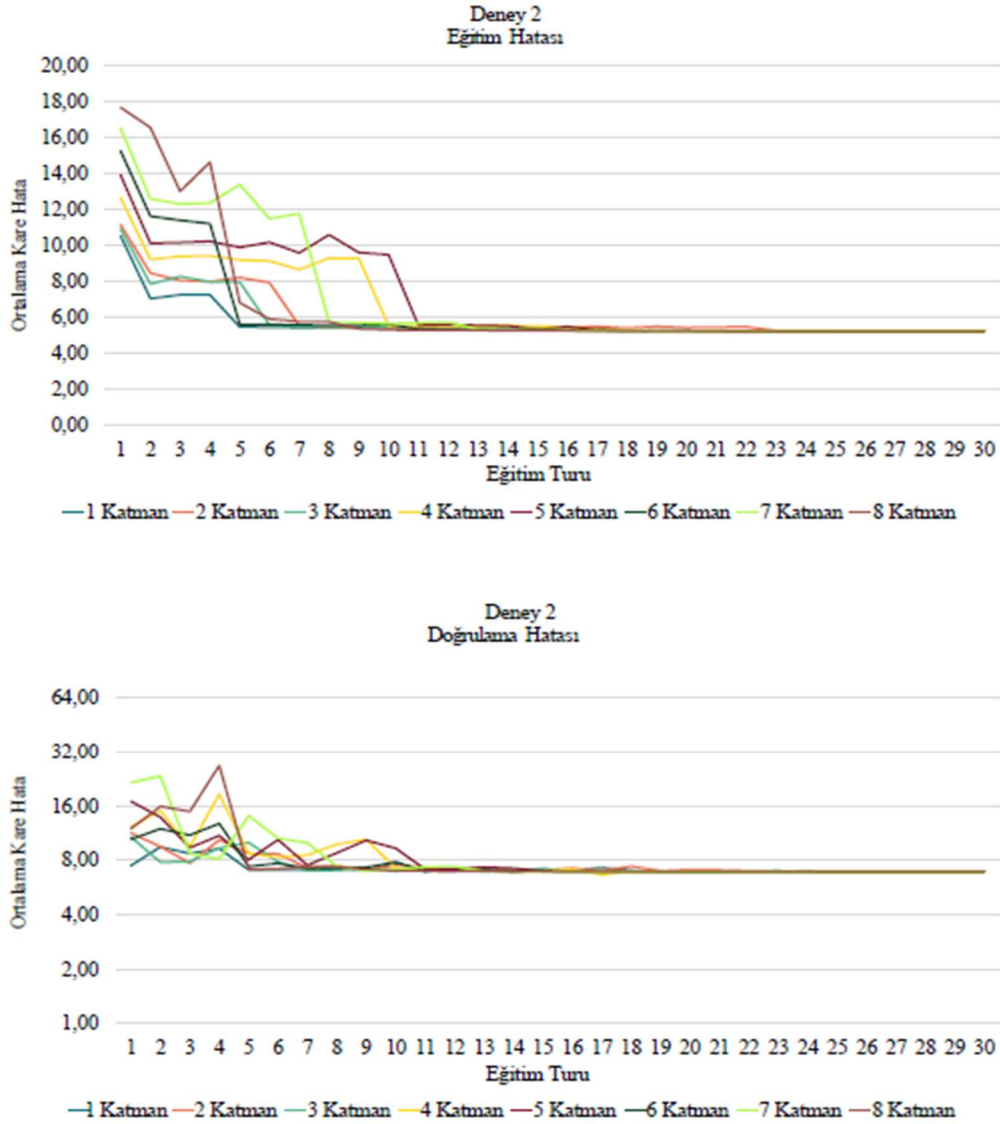
Çalışmada gerçekleştirilen 1, 2, 3, 4 ve 5. deneylerin eğitim ve doğrulama aşamalarının eğitim periyodu sayısına göre değişimi Şekil 19-23'te verilmektedir. Ek olarak, her deneyin test sonuçları Tablo 4' te gösterilmektedir.



Şekil 19. Deney 1 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi.

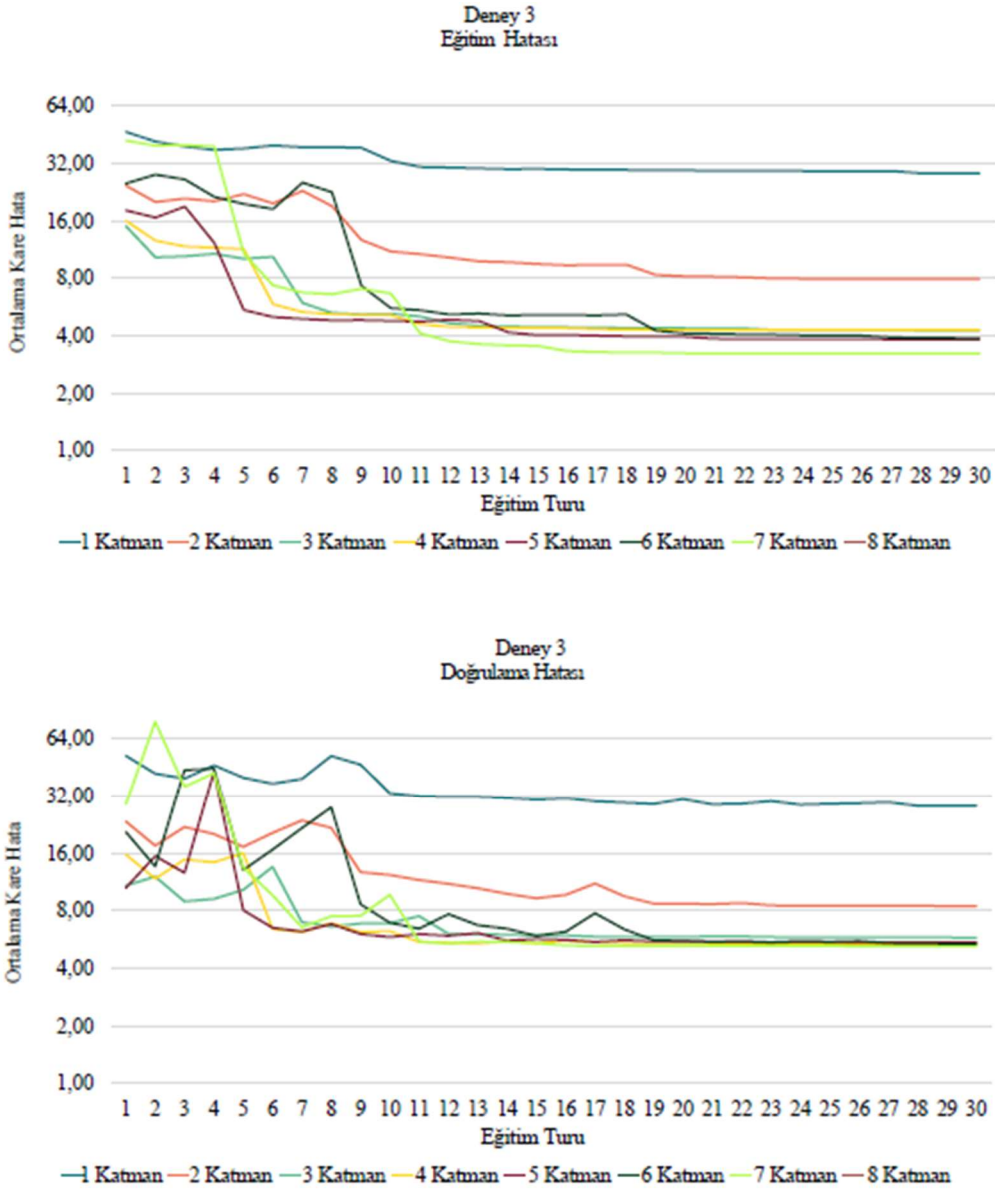
Şekil 20'de görüldüğü gibi 1. deneyde varyasyon katmanı sayısı arttıkça eğitim ve doğrulama hatalarının azaldığı görülmektedir. Deney1 için Tablo 4'e bakıldığında katman sayısı arttıkça test sonuçlarında düzenli bir artış veya azalma olmamaktadır.

Ancak en düşük hata 6 katmanlı modelde elde edilmiştir. Bu hata, yapılan tüm deneyler arasında elde edilen en düşük hata değeridir. 6. katmandan sonra hata değerlerinin artması R^2 skorundan da anlaşılacağı üzere modelin veri noktalarına olan mesafesinin arttığını göstermektedir.



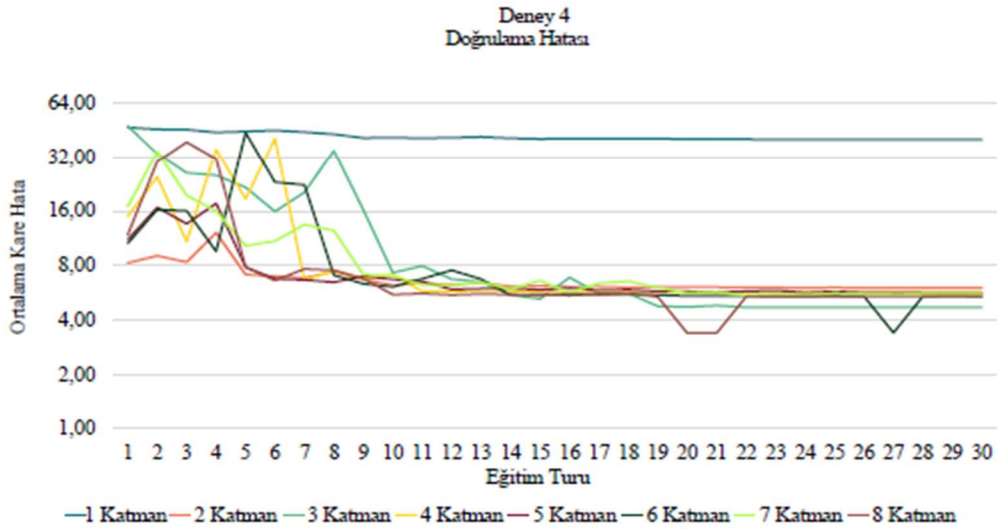
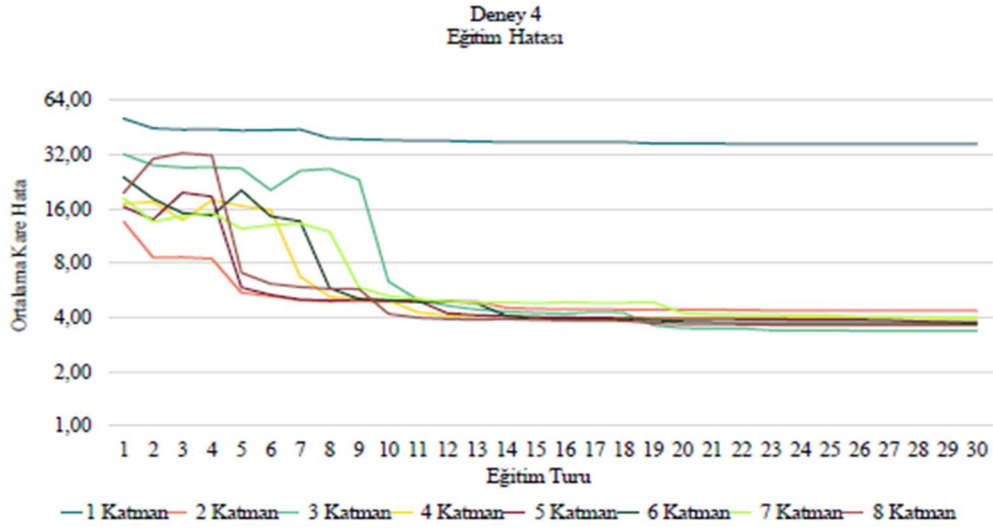
Şekil 20. Deney 2 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi.

Deney 2’de katman sayısı arttıkça eğitim ve doğrulama hatası 11. eğitim dönemine kadar azalmış, fakat daha sonrasında herhangi bir iyileşme görülmemiştir. Tablo 4’te görüleceği gibi katman sayısının artması test hatalarında aşağı yönlü bir etkiye neden olmamıştır. Bu bağlamda Tablo 4’e göre bakıldığında dolaşıklık katmanının kullanılması test sonuçlarına büyük etkiye sahip olduğu gözlenmektedir.



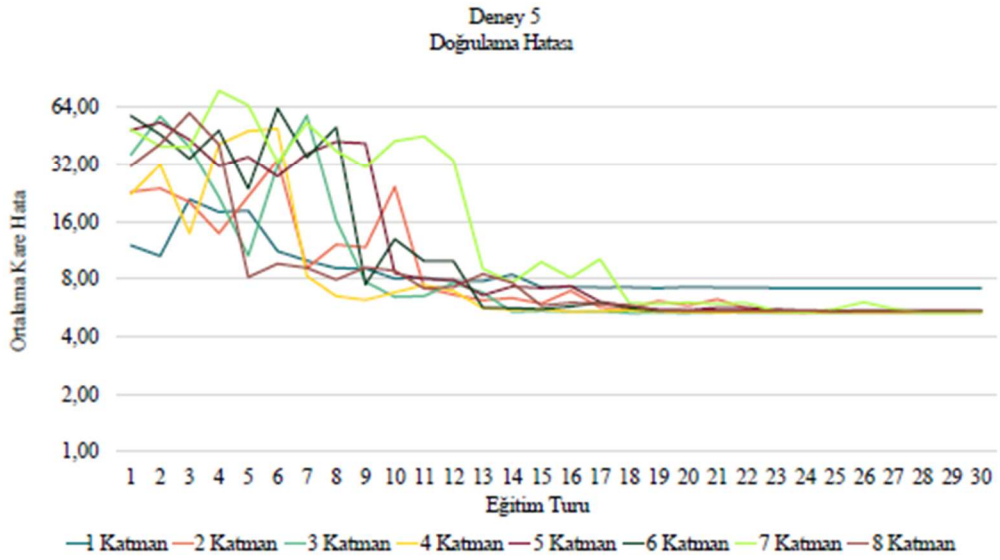
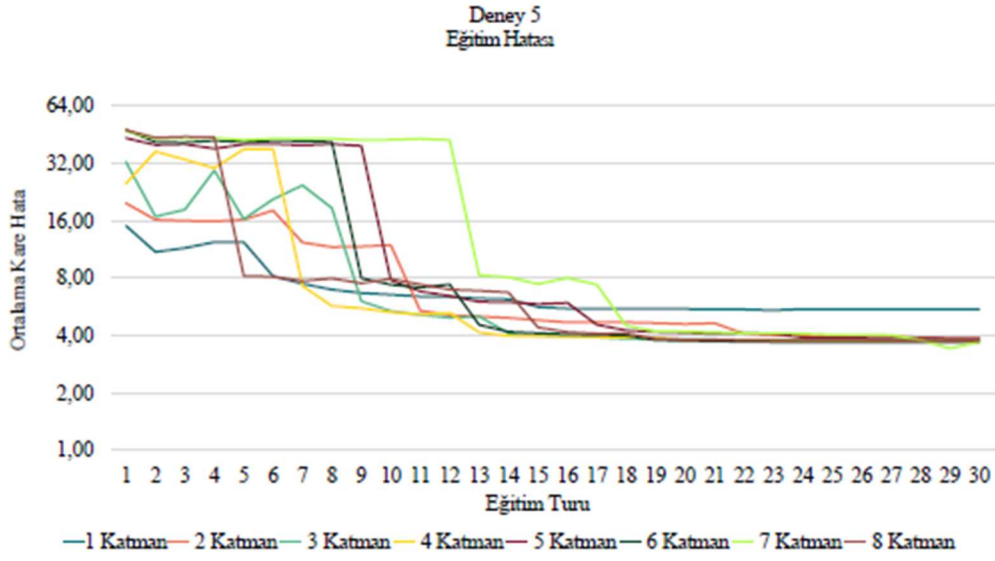
Şekil 21. Deney 3 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi.

Deney 3'te katman sayısı artıkça eğitim ve doğrulama hatasında bir azalma gözlenmiş fakat deney 1 ile karşılaştırıldığında bu azalma miktarının dikkate değer bir azalma olmadığı gözlenmiştir. Ancak Tablo 4'teki test hata sonuçlarına göre karşılaştırılma yapıldığında Deney 1'in Deney 3'e göre sekizinci katmana kadar daha düşük hata oranı ile daha yüksek bir performans gösterdiği söylenebilir. Bu bağlamda veri kodlama katmanından önce süperpozisyon katmanının kullanılması daha az varyasyonel katman kullanmamıza olanak sağladığı tespit edilmiştir.



Şekil 22. Deney 4 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi.

Deney 4'te katman sayısı arttıkça eğitim ve doğrulama hatası deney 1'e benzerdir. Ancak Tablo 4'e bakıldığında deney 1'de elde edilen hata sonuçlarının genel olarak daha düşük olduğu görülmektedir.



Şekil 23. Deney 5 için eğitim hatasının ve doğrulama hatasının eğitim dönemi sayısına göre değişimi.

Deney 5'te ise katman sayısı arttıkça eğitim ve doğrulama hatasında azalma meydana gelmiş fakat Tablo 4'teki test sonuçlarına göre deney 1 ve 2 ile karşılaştırıldığında daha yüksek hata sonucuna sahip olduğu görülür.

5. SONUÇ ve ÖNERİLER

Kuantum bilgisayar teknolojileri ilerledikçe, kuantum makine öğrenmesi de paralel olarak gelişir ve yeni alanlara uygulanır. Hibrit bir modelleme olan varyasyonel kuantum devre algoritmalarında kuantum hesaplamanın hızlı işlem gücü kullanılarak klasik makine öğrenmesine göre daha iyi sonuçlar alındığı bilinmektedir. Bu çalışmada, hava tahmini için küçük boyutlu eğitim verileri kullanılarak, kuantum hesaplamanın iki önemli özelliği olan dolaşıklık ve süperpozisyonun, hibrit kuantum-klasik makine öğrenmesinin performansı üzerindeki etkisi araştırılmıştır. Değişken katmanlar arasında dolaşıklık katmanının kullanılması, devre performansı üzerinde önemli iyileşmeler sağlamıştır (bkz. Tablo 4). Hata toleransı performansı açısından, dolaylı (transfer) dolaşıklık katmanının etkisi, doğrudan dolaşıklığın etkisinden daha iyidir (bkz. Tablo 4). Süperpozisyon katmanının veri kodlama katmanından önce kullanılması, daha az varyasyon katmanının kullanılmasını sağlar (bakınız Tablo 4). Hesaplama karmaşıklığı ve hata toleransı açısından yakın dönem kuantum teknolojileri için uygundur.

Donanımsal olarak geliştirilen kuantum bilgisayarlar ve kullanılmakta olan simülatörler geliştikçe, hesaplama tabanlı yaklaşımlardaki problem çözme becerilerimiz gelişecektir. Günümüzde üstel seviyede artan datalar ve bilgisayarların işlem gücündeki artış makine öğrenme alanını hızla değiştirmektedir. Kuantum hesaplamanın yetenekleri, makine öğrenmesi için sınır problemler olarak kabul edilen otonom sistemlerin geliştirilmesi, makineler arası iletişimin anlamlı seviyede gerçekleştirilmesi (IoT-Nesnelerin İnterneti), doğal dil işleme, açıklanabilirlik ve yorumlanabilirlik gibi problemlerin etkili çözümünde kuantum makine öğrenmesinin yöntem olarak seçilmesini anlamlı kılmaktadır.

Tablo 4. Tüm Deneyler için test verilerinde ortalama kare hatası, R² Skoru ve Standart Sapma Sonuçları

Deney 1				
# Katman	OKH (MSE)	R ² Skoru	Standart Sapma	
1	33.7237	0.4171	46.7110	
2	5.4962	0.9050	7.2614	
3	4.2573	0.9264	7.3165	
4	3.242	0.9440	7.3420	
5	4.0139	0.9295	7.3265	
6	3.0793	0.9468	7.3356	
7	3.3926	0.9414	7.3463	
8	4.1031	0.9291	7.4942	
Deney 2				
# Katman	OKH (MSE)	R ² Skoru	Standart Sapma	
1	5.3956	0.9067	7.2827	
2	5.3896	0.9068	7.2925	
3	5.402	0.9066	7.2985	
4	5.4459	0.9059	7.2886	
5	5.4136	0.9064	7.2869	
6	5.3996	0.9067	7.2958	
7	5.4203	0.9063	7.2854	
8	5.4148	0.9064	7.2896	
Deney 3				
# Katman	OKH (MSE)	R ² Skoru	Standart Sapma	
1	28.3449	0.5101	48.5700	
2	8.4131	0.8546	7.1925	
3	4.4781	0.9226	7.2786	
4	4.4672	0.9228	7.3231	
5	4.084	0.9294	7.3505	
6	4.0592	0.9298	7.3525	
7	3.6647	0.9367	7.3556	
8	3.99	0.9311	7.3448	
Deney 4				
# Katman	OKH (MSE)	R ² Skoru	Standart Sapma	
1	36.3256	0.3721	42.1010	
2	4.5752	0.9209	7.3308	
3	3.6155	0.9375	7.3252	
4	4.1919	0.9275	7.4157	
5	4.0221	0.9305	7.3322	
6	4.0032	0.9308	7.3472	
7	4.238	0.9267	7.3423	
8	3.9611	0.9315	7.3467	
Deney 5				
# Katman	OKH (MSE)	R ² Skoru	Standart Sapma	
1	5.6709	0.9020	4.2380	
2	4.1363	0.9285	7.3147	
3	4.0384	0.9302	7.3366	
4	3.9693	0.9314	7.3562	
5	4.096	0.9292	7.3530	
6	4.0213	0.9305	7.3064	
7	3.9637	0.9315	7.4283	
8	4.0156	0.9306	7.3475	

KAYNAKLAR

- Aaronson, S.; Grier, D. ve Schaeffer, L. (2015). The classification of reversible bit operations. 03.11.2022 tarihinde <https://www.scottaaronson.com/papers/gates.pdf> adresinden alındı.
- Abdi, H. ve Williams, L. J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4). 433-459.
- Abhijith, J., Adedoyin, A., Ambrosiano, J., Anisimov, P., Casper, W., Chennupati, G., ve Lokhov, A. Y. (2018). Quantum algorithm implementations for beginners. DOI:10.1145/3517340
- Aımeur, E., Brassard, G., ve Gambs, S. (2006). Machine learning in a quantum world. In *Advances in Artificial Intelligence: 19th Conference of the Canadian Society for Computational Studies of Intelligence, Canadian AI 2006, Québec City, Québec, Canada, June 7-9, 2006. Proceedings 19* (pp. 431-442). DOI: 10.1007/11766247_37.
- Alpaydin, E. (2020). *Introduction to machine learning*. Londra: MIT press.
- Andreucut, M., ve Ali, M. K. (2002). A quantum neural network model. *International Journal of Modern Physics C*, 13(01), 75-88.
- Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., ve Martinis, J. M. (2019). Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779), 505-510.
- Benedetti, M., Lloyd, E., Sack, S., ve Fiorentini, M. (2019). Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4), 043001.
- Benedetti, M., Realpe-Gómez, J., Biswas, R., ve Perdomo-Ortiz, A. (2016). Estimation of effective temperatures in quantum annealers for sampling applications: A case

study with possible applications in deep learning. *Physical Review A*, 94(2), 022308.

Benioff, P. (1980). The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *Journal of statistical physics*, 22, 563-591.

Bergholm, V., Izaac, J., Schuld, M., Gogolin, C., Ahmed, S., Ajith, V., ve Killoran, N. (2018). PennyLane: Automatic differentiation of hybrid quantum-classical computations. DOI: 10.48550/arXiv.1811.04968

Bernstein, E., ve Vazirani, U. (1993). Quantum complexity theory. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing.*, (s. 11-20). DOI: 10.1137/S0097539796300921

Bertlmann, A. R. (2023). Mathematical Formalism of Quantum Mechanics. 14.03.2023 tarihinde http://homepage.univie.ac.at/reinhold.bertlmann/pdfs/T2_Skript_Chapter_3.pdf adresinden alındı.

Biamonte, J. D., ve Love, P. J. (2008). Realizable Hamiltonians for universal adiabatic quantum computers. *Physical Review A*, 78(1), 012352.

Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., ve Lloyd, S. (2017). Quantum machine learning. *Nature*, 549(7671), 195-202.

Bradley, P. S., ve Fayyad, U. M. (1998). Refining initial points for k-means clustering. In *ICML (Vol. 98)*, 91-99.

Broughton, M., Verdon, G., McCourt, T., Martinez, A. J., Yoo, J. H., Isakov, S. V., ve Mohseni, M. (2020). Tensorflow quantum: A software framework for quantum machine learning. DOI: 10.48550/arXiv.2003.02989

Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K., ve Coles, P. J. (2021). Variational quantum algorithms. *Nature Reviews Physics*, 3(9), 625-644.

- Copeland, B. J. (1997). *The church-turing thesis*. 08.02 .2023 tarihinde <https://plato.stanford.edu/ENTRIES/church-turing/> adresinden alındı.
- Cortes, C., ve Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20, 273-297.
- da Silva, A. J., de Oliveira, W. R., ve Ludermir, T. B. (2015). Comments on “quantum mp neural network”. *International Journal of Theoretical Physics*, 54, 1878-1881.
- de Oliveira, W. R., Silva, A. J., Ludermir, T. B., Leonel, A., Galindo, W. R., ve Pereira, J. C. (2008, Ekim). Quantum logical neural networks. In 2008 10th Brazilian Symposium on Neural Networks. *IEEE*, 147-152.
- Desurvire, E. (2009). *Classical and quantum information theory: an introduction for the telecom scientist*. New York: Cambridge University press.
- Deutsch, D. (1985). Quantum theory, the Church–Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818). s. 97-117.
- Deutsch, D., ve Jozsa, R. (1992). Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439(1907), 553-558.
- DiVincenzo, D. (1997). Topics in quantum computers. *Mesoscopic electron transport*. DOI: 10.1007/978-94-015-8839-3_18.
- Dreyfus, G. (2005). *Neural networks: methodology and applications*. Springer Science ve Business Media. Verlag Berlin Heidelberg: Springer.
- Durr, C., ve Hoyer, P. (1996). A quantum algorithm for finding the minimum. DOI: 10.48550/arXiv.quant-ph/9607014.
- Ezhov, A. A., ve Ventura, D. (2000). Quantum neural networks. *Future Directions for Intelligent Systems and Information Sciences: Studies in Fuzziness and Soft Computing*, 45

- Farhi, E., ve Neven, H. (2018). Classification with quantum neural networks on near term processors. DOI:10.48550/arXiv.1802.06002.
- Fernandes de Mello, R., ve Antonelli Ponti, M. (2018). A brief review on machine learning. *Machine learning: A practical approach on the statistical learning theory*. 1-74.
- Feynman, R. (1960). There is plenty of room at the bottom. *California Institute of Technology. Int. J. Eng. Sci*, 4, 23-36.
- Frolov, A. V. (2017). Can a quantum computer be applied for numerical weather prediction?. *Russian meteorology and hydrology*, 42,, 545-553.
- García, D. P., Cruz-Benito, J., ve García-Peñalvo, F. J. (2022). Systematic literature review: Quantum machine learning and its applications. DOI: 10.48550/arXiv.2201.04093
- Gasiorowicz, S. (2007). *Quantum physics*. Hoboken: John Wiley ve Sons.
- Giovannetti, V., Lloyd, S., ve Maccone, L. (2008). Quantum random access memory. *Physical review letters*, 100(16), 160501.
- Godfrey, M. D. (1993). Introduction to “The First Draft Report on the EDVAC” by John von Neumann. *Annals of the History of Computing*, 15(1), 11-21.
- Gonçalves, C. P. (2016). Quantum neural machine learning-backpropagation and dynamics. DOI: 10.14704/nq.2017.15.1.1008.
- Griffiths, D. J., ve Schroeter, D. F. (2018). *Introduction to quantum mechanics*. Cambridge: Cambridge university press.
- Grover, L. K. (1996, Temmuz). A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 212-219. DOI: 10.1145/237814.237866 .
- Guerreschi, G. G., ve Smelyanskiy, M. (2017). Practical optimization for hybrid quantum-classical algorithms. DOI: 10.48550/arXiv.1701.01450

- Guicciardini, N. (2005). Isaac newton, philosophiae naturalis principia mathematica, (1687). In Landmark Writings in Western Mathematics 1640-1940. *Elsevier Science*, 59-87.
- Hales, L., ve Hallgren, S. (2000, Kasım). An improved quantum Fourier transform algorithm and applications. In Proceedings 41st Annual Symposium on Foundations of Computer Science. *IEEE*, 515-525.
- Harrow, A. W., Hassidim, A., ve Lloyd, S. (2009). Quantum algorithm for linear systems of equations. *Physical review letters*, 103(15), 150502.
- Hartigan, J. A., ve Wong, M. A. (1979). A k-means clustering algorithm. *Applied statistics*, 28(1), 100-108.
- Hinton, G. E., Osindero, S., ve Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7), 1527-1554.
- Holmstrom, M., Liu, D., ve Vo, C. (2016, Aralık). *Machine Learning Applied to Weather Forecasting, Stanford University*.
- Howard, D., Mark, B., ve Martin, H. (2004). *Neural network toolbox user's guide. The Mathwork*. 15.06.2023 tarihinde <https://www.mathworks.com/>: <https://citeseerx.ist.psu.edu/document?repid=rep1&vtype=pdf&doi=59c12012aa5481818f98d3c04a8b8f65f1f1d185> adresinden alındı.
- Hu, Y. H., ve Hwang, J. N. (2002). *Handbook of neural network signal processing*. Boca Raton, Florida: CRC Press LLC.
- Janiesch, C., Zschech, P., ve Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31(3), 685-695.
- Jolliffe, I. T., ve Cadima, J. (2016). Principal component analysis: a review and recent developments. *Philosophical transactions of the royal society A. Mathematical, Physical and Engineering Sciences*, 374(2065), 20150202.

- Kak, S. C. (1995). Quantum neural computing. *Advances in imaging and electron physics*, 94, 259-313. DOI: 10.1016/S1076-5670(08)70147-2.
- Kitaev, A. Y. (1995). Quantum measurements and the Abelian stabilizer problem. DOI: 10.48550/arXiv.quant-ph/9511026
- Kitaev, A. Y., Shen, A., Vyalyi, M. N., ve Vyalyi, M. N. (2002). *Classical and quantum computation (No. 47)*. Rhode Island: American Mathematical Society.
- Kopczyk, D. (2018). Quantum machine learning for data scientists. DOI: 10.48550/arXiv.1804.10068
- Liu, C. Y., Chen, C., Chang, C. T., ve Shih, L. M. (2013). Single-hidden-layer feed-forward quantum neural network based on Grover learning. *Neural Networks*, 45, 144-150.
- Lloyd, S., Mohseni, M., ve Rebentrost, P. (2013). Quantum algorithms for supervised and unsupervised machine learning. DOI: 10.48550/arXiv.1307.0411
- Lloyd, S., Mohseni, M., ve Rebentrost, P. (2014). Quantum principal component analysis. *Nature Physics*, 10(9), 631-633.
- Lomonaco, S. J., ve Brandt, H. E. (2002). *Quantum Computation and Information: AMS Special Session Quantum Computation and Information*, January 19-21, 2000, Washington (Vol. 305). *American Mathematical Soc.*
- MacQuen, J. B. (1967). Some methods for classification and analysis of multivariate observation. In *Proceedings of the 5th Berkley Symposium on Mathematical Statistics and Probability*. (Vol. 1, No. 14, pp. 281-297).
- Maheshwari, A., ve Smid, M. (2012). *Introduction to theory of computation*. Ottawa: Carleton University.
- Manin, Y. (1980). Computable and uncomputable. *Sovetskoye Radio, Moscow*, 128, 28

- McClean, J. R., Romero, J., Babbush, R., ve Aspuru-Guzik, A. (2016). The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 18(2), 023023.
- McMahon, D. (2007). *Quantum computing explained*. Hoboken, New Jersey: John Wiley ve Sons.
- Menneer, T., ve Narayanan, A. (1995). Quantum-inspired neural networks. *Tech. Rep. R329*.
- Mitarai, K., Negoro, M., Kitagawa, M., ve Fujii, K. (2018). Quantum circuit learning. *Physical Review A*, 98(3), 032309.
- Nielsen, M. A., ve Chuang, I. (2002). *Quantum computation and quantum information*. Cambridge: Cambridge University Press.
- Perus, M. (1996). Neuro-quantum parallelism in brain-mind and computers. *Informatica*, 20, 173-183.04.10.2022 tarihinde <https://www.dlib.si/details/URN:NBN:SI:DOC-4EIHJJ0P?&language=eng> adresinden alındı.
- Peruzzo, A., McClean, J., Shadbolt, P., Yung, M. H., Zhou, X. Q., Love, P. J., ve O'brien, J. L. (2014). A variational eigenvalue solver on a photonic quantum processor. *Nature communications*, 5(1), 4213.
- Plenio, M. (2002). *Quantum Mechanics*. Londra: Imperial College.
- Preskill, J. (2018). Quantum computing in the NISQ era and beyond. *Quantum*, 2, 79.
- Rebentrost, P., Mohseni, M., ve Lloyd, S. (2014). Quantum support vector machine for big data classification. *Physical review letters*, 113(13), 130503.
- Ricks, B., ve Ventura, D. (2003). Training a quantum neural network. *Advances in neural information processing systems*, 16.
- Ruder, S. (2016). An overview of gradient descent optimization algorithms. DOI: 10.48550/arXiv.1609.04747.

- Safari, A., ve Ghavifekr, A. A. (2021). Quantum Neural Networks (QNN) Application in Weather Prediction of Smart Grids. In *2021 11th Smart Grid Conference (SGC). IEEE*, 1-6. DOI:10.1109/SGC54087.2021.9664117.
- Sađırođlu, Ő. B. (2003). *Mühendislikte Yapay Zeka Uygulamaları*. Kayseri: Ufuk Yayıncılık.
- Sakurai, J. J., ve Commins, E. D. (1995). *Modern quantum mechanics, revised edition*. San Francisco: Addison-Wesley.
- Scheinerman, E. R. (1996). *Invitation to Dynamical System*. . Baltimore, Maryland: Johns Hopkins University.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61, 85-117.
- Schuld, M., ve Killoran, N. (2019). Quantum machine learning in feature hilbert spaces. *Physical review letters*, 122(4), 040504
- Schuld, M., ve Petruccione, F. (2018). *Supervised learning with quantum computers*. Berlin: Springer.
- Schuld, M., Bergholm, V., Gogolin, C., Izaac, J., ve Killoran, N. (2019). Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3), 032331.
- Schuld, M., Sinayskiy, I., ve Petruccione, F. (2014). The quest for a quantum neural network. *Quantum Information Processing*, 13, s. 2567-2586.
- Schuld, M., Sinayskiy, I., ve Petruccione, F. (2015). An introduction to quantum machine learning. *Contemporary Physics*, 56(2), 172-185.
- Schuld, M., Sinayskiy, I., ve Petruccione, F. (2015). Simulating a perceptron on a quantum computer. *Physics Letters A*, 379(7), 660-663.
- Seow, K. L., Behrman, E., ve Steck, J. (2015). Efficient learning algorithm for quantum perceptron unitary weights. DOI: 10.48550/arXiv.1512.00522.

- Shor, P. W. (1994, Kasım). Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science (pp. 124-134)*. IEEE. 124-134. DOI:10.1109/sfcs.1994.365700.
- Sinaga, K. P., ve Yang, M. S. (2020). Unsupervised K-means clustering algorithm. *IEEE access*, 8, 80716-80727.
- Siomau, M. (2014). A quantum model for autonomous learning automata. *Quantum information processing*, 13(5), 1211-1221.
- Sipser, M. (2006). *Introduction to the Theory of Computation, Second Edition*. Boston: Thomson Course Tecnology.
- Smola, A. J., ve Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, 14, 199-222.
- Suykens, J. A., ve Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural processing letters*, 9, 293-300.
- Tacchino, F., Macchiavello, C., Gerace, D., ve Bajoni, D. (2019). An artificial neuron implemented on an actual quantum processor. npj. *Quantum Information*, 5(1), 26.
- Tusarova, T. (2004). Quantum complexity classes. Cornell University, DOI:10.48550/arXiv.cs/0409051
- Ullah Khan, S. (2019). *Quantum K means Algorithm*. 11.04.2023 tarihinde <https://www.divaportal.org/smash/record.jsf?pid=diva2%3A1381305&dswid=7877> adresinden alındı.
- Wiggins, S. (2003). *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. . Berlin: Springer-Verlag.
- Williams, C. P., ve Clearwater, S. H. (1998). *Explorations in quantum computing*. New York: Springer-Verlag.

- Williams, C. P., ve Clearwater, S. H. (1999). *Ultimate zero and one: computing at the quantum frontier*. New York: Springer-Verlag.
- Zak, M., ve Williams, C. P. (1998). Quantum neural nets. *International journal of theoretical physics*, 37(2), 651-684.
- Zhang, Y., ve Ni, Q. (2020). Recent advances in quantum machine learning. *Quantum Engineering*, 2(1), e34.
- Zhou, R. (2010). Quantum competitive neural network. *International Journal of Theoretical Physics*, 49, 110-119.
- Zubov, D., Volponi, F., ve Khosravy, M. (2015, Ekim). D-wave quantum computing ising model
- Zubov, D., Volponi, F., ve Khosravy, M. (2015, Ekim). D-wave quantum computing ising model: A case study for the forecasting of heat waves. In *2015 International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, s. 149-152. DOI: 10.1109/ICCAIS.2015.7338651

ÖZGEÇMİŞ

Beşir OĞUR

Eğitim

Yüksek Lisans	2007	Beykent Üniversitesi, Bilgisayar Mühendisliği
Lisans	1998	Van Yüzüncü Yıl Üniversitesi, Fizik Öğretmenliği
Lisans	2007	Ahmet Yesevi Üniversitesi, Bilgisayar Mühendisliği

İş/İstihdam (Varsa)

1998 Milli Eğitim Bakanlığında Fizik Öğretmeni

Yayınlar ve Diğer Bilimsel/Sanatsal Faaliyetler

1. Beşir Oğur ve İhsan Yılmaz, The Effect of Superposition and Entanglement on Hybrid Quantum Machine Learning for Weather Forecasting' Quantum Information and Computation, Vol. 23, No. 3&4 (2023) 0181–0194, ©Rinton Press. doi: <https://doi.org/10.26421/QIC23.3-4-1>

